# Can VAEs Generate Novel Examples?

**Alican Bozkurt**
Northeastern University
Boston, MA 02115
alican@ece.neu.edu

**Babak Esmaeili**
Northeastern University
Boston, MA 02115
esmaeili.b@husky.neu.edu

**Dana H. Brooks**
Northeastern University
Boston, MA 02115
brooks@ece.neu.edu

**Jennifer G. Dy**
Northeastern University
Boston, MA 02115
jdy@ece.neu.edu

**Jan-Willem van de Meent**
Northeastern University
Boston, MA 02115
j.vandemeent@northeastern.edu

## Abstract

An implicit goal in works on deep generative models is that such models should be able to generate novel examples that were not previously seen in the training data. In this paper, we investigate to what extent this property holds for widely employed variational autoencoder (VAE) architectures. VAEs maximize a lower bound on the log marginal likelihood, which implies that they will in principle overfit the training data when provided with a sufficiently expressive decoder. In the limit of an infinite capacity decoder, the optimal generative model is a uniform mixture over the training data. More generally, an optimal decoder should output a weighted average over the examples in the training data, where the magnitude of the weights is determined by the proximity in the latent space. This leads to the hypothesis that, for a sufficiently high capacity encoder and decoder, the VAE decoder will perform nearest-neighbor matching according to the coordinates in the latent space. To test this hypothesis, we investigate generalization on the MNIST dataset. We consider both generalization to new examples of previously seen classes, and generalization to the classes that were withheld from the training set. In both cases, we find that reconstructions are closely approximated by nearest neighbors for higher-dimensional parameterizations. When generalizing to unseen classes however, lower-dimensional parameterizations offer a clear advantage.

## 1 Introduction

Variational autoencoders [Kingma and Welling, 2013, Rezende et al., 2014] jointly train a generative model $p_\theta(\boldsymbol{x}, \boldsymbol{z})$ and an inference model $q_\phi(\boldsymbol{z}, \boldsymbol{x})$. The generative model is defined in terms of a likelihood $p_\theta(\boldsymbol{x} \mid \boldsymbol{z})$ and a prior $p(\boldsymbol{z})$. The likelihood is parameterized using a neural network, which we refer to as the *decoder*, where the prior is most commonly a spherical Gaussian. The inference model is defined in terms of a conditional $q_\phi(\boldsymbol{z} \mid \boldsymbol{x})$, parameterized by an *encoder* network and the empirical distribution $\hat{p}(\boldsymbol{x})$,

$$\mathcal{L}(\theta, \phi) = \mathbb{E}_{\hat{p}(\boldsymbol{x})}\left[\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p_\theta(\boldsymbol{x}, \boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\right]\right] \leq \mathbb{E}_{\hat{p}(\boldsymbol{x})}\left[\log p_\theta(\boldsymbol{x})\right], \qquad \hat{p}(\boldsymbol{x}) = \frac{1}{N}\sum_{n=1}^{N} \delta_{\boldsymbol{x}_n}(\boldsymbol{x}).$$

One of the basic premises for works on deep generative models (including VAEs) is that we expect such models to not only reproduce the data that they are trained on faithfully, but also to generate entirely novel but plausible samples. The ambiguity in the terms “novel” and “plausible” in this premise induces a vague idea about generalization.

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

Even though discriminative models have been enjoying a clear definition of "generalization performance" and guaranteed generalization bounds due to statistical learning theory [Wang et al., 2018], works on generalization in generative models started to emerge only very recently, without a widely agreed metric, and honestly, without even a clear definition of generalization.

While generalization has been simply defined as the performance on a test dataset, here we concentrate on a more challenging task. Under the *manifold hypothesis*, real-world data presented in high dimensional spaces are expected to lie in a manifold-space of much lower dimensionality [Bengio et al., 2013]. The data in the train and test sets can be different in the high-dimensional space, but identical in the feature space. We define generalization as the ability to reason about unseen data that is significantly different than the training data in the feature space. For example, we can imagine a dataset containing different shapes and colours such that all shapes and colours exist, but not all combinations exist. A model that is able to generalize well must perform well on combinations of shapes and colours not seen during training.

An interesting observation made by several researchers [Bousquet et al., 2017, Rezende and Viola, 2018, Alemi et al., 2018], but given in its most general from in Shu et al. [2018], is that the reconstruction obtained from an optimal decoder of a VAE is a convex combination of examples in the training data.

**Theorem 1 (Shu et al. [2018])** *Let $\mathcal{P}$ be an exponential family with corresponding mean parameter space $\mathcal{M}$ and sufficient statistic function $T(\cdot)$. Consider $\boldsymbol{z} \in \mathcal{Z}$, $g \in \mathcal{G} : \mathcal{Z} \to \mathcal{M}$, and a fixed $q_\phi(\boldsymbol{z}|\boldsymbol{x})$. Supposing $\mathcal{G}$ has infinite capacity, then the optimal generative model $g^*$ returns:*

$$\mu = g^*(\boldsymbol{z}) = \sum_{i=1}^{n} q_\phi(\boldsymbol{x}^{(i)}|\boldsymbol{z}) T(\boldsymbol{x}^{(i)}) = \sum_{i=1}^{n} \frac{q_\phi(\boldsymbol{z}|\boldsymbol{x}^{(i)})}{\sum_j q_\phi(\boldsymbol{z}|\boldsymbol{x}^{(j)})} T(\boldsymbol{x}^{(i)}) = \sum_{i=1}^{n} w_i(\boldsymbol{z}) T(\boldsymbol{x}^{(i)}). \tag{1}$$

In the specific case of a Bernoulli likelihood, Theorem 1 states that the reconstructed image is simply a weighted average over images observed during training, where the magnitude of the weights is governed by the proximity to training examples in the latent space. For the rest of this paper, when we talk about "weighted average" or "weights", we refer to the terms $\mu$ and $w_i$ in Theorem 1 respectively, and we use $\hat{\boldsymbol{x}}$ to refer to the output of the decoder.

Theorem 1 raises a number of interesting questions. How closely do commonly trained VAEs mirror this behavior in practice? How strong is the infinite capacity assumption and what is the relationship between the model capacity and generalization? On average, what does the distribution of weights look like for a given test instance?

To answer these questions, we perform a series of experiments, explained in the following section. In the heart of all our experiments lies this hypothesis: if Theorem 1 holds, then VAEs should not able to reconstruct an out-of-training-distribution sample.

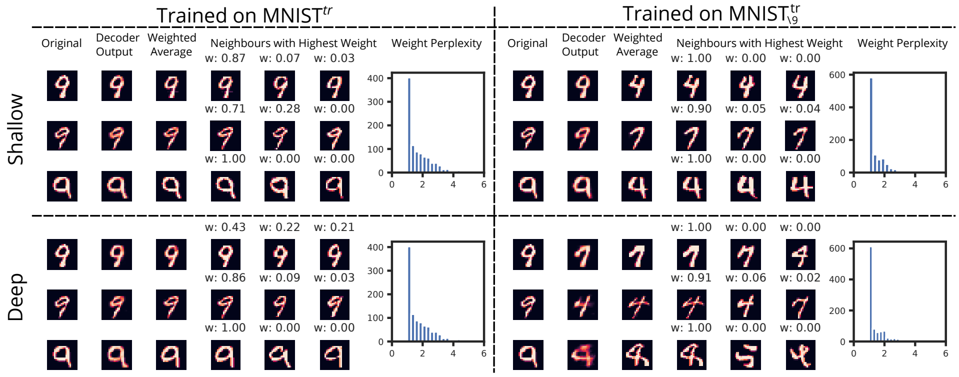

Figure 1: Reconstruction of seen vs unseen classes for shallow and deep VAE.

## 2 Experiments

We will be using different partitions of the MNIST dataset in our experiments, so we find it useful to introduce a notation for brevity. Let MNIST$^{te}_{d}$ be the dataset containing samples of digit $d$ from test set of MNIST. Similarly, MNIST$^{tr}_{\backslash d}$ denotes the dataset containing samples of all digits except $d$ from training set of MNIST. When used without any subscripts, e.g. MNIST$^{tr}$, it indicates samples from all digits are present in the dataset.

In order to evaluate in-distribution and out-of-distribution generalization performance of VAEs, we perform the following experiment. We train a shallow and a deep VAE with dim$(\boldsymbol{z}) = 50$ on two datasets based on MNIST: MNIST$^{tr}$ and MNIST$^{tr}_{\backslash 9}$. All networks are evaluated on the same test dataset MNIST$^{te}_{9}$. In the shallow encoder and decoder, we have used a single hidden layer (400 neurons), and their deep counterparts have three hidden layers (400, 200, and 100 neurons). After training, we evaluate all four models by feeding them samples from MNIST$^{te}_{9}$ and observing the decoder output, the weighted average calculated according to (1), and the three training examples with the highest weight (see Figure 1).

We make three observations here. First, for both VAEs trained on MNIST$^{tr}$, reconstruction and weighted average images look quite similar (see "Trained on MNIST$^{tr}$" column in Figure 1). Second, for most test samples presented, it is only a single training example that contributes to the weighted average. We confirm this quantitatively by computing the perplexity histograms of the weights for all test data, which we show in Figure 1. For all cases, the histograms have a high peak at 1, indicating that for most test examples, a single training sample has weight 1 and the rest have zero weight.[1] Note that this is equivalent to $k$-nearest neighbour matching with $k = 1$.

Perhaps the most fascinating finding here is the shallow VAE's ability to generalize to out-of-training-distribution samples: It can reconstruct perfectly passable nines even though it has not seen one during training. Furthermore, the assumption of infinite capacity in Theorem 1 clearly matters as the decoder outputs are not similar to the weighted average for the shallow VAE trained on MNIST$^{tr}_{\backslash 9}$. As shown in Figure 1, the reconstructions are more similar to the input images in the shallow VAE, while being closer to the weighted average in the deep VAE. We confirm this by comparing the binary cross entropy loss of reconstructions to the samples from withheld class and the weighted average (see Figure 2). The reconstructions are closer to the input than the weighted average for the shallow VAE (most of the mass of blue distribution lies below the line). On the other hand, the reconstructions are closer closer to the weighted average than the input for the deep VAE (most of the mass of green distribution lies above the line).

One final counter-intuitive observation is that increasing the complexity affects encoder and decoder differently. For decoder, we see some change in characteristics of the reconstructions, whereas the weight perplexity histograms are similar for both deep and shallow cases, which can be considered as a characteristic of the encoder.

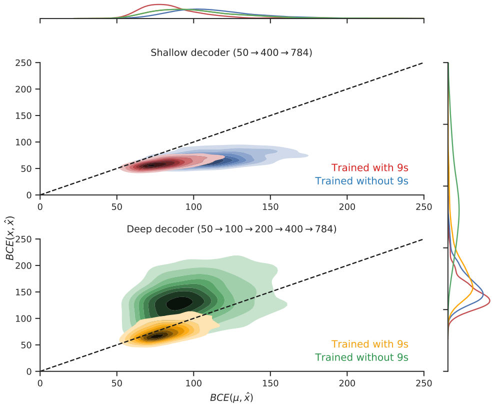

Figure 2: Distribution of binary cross entropy loss between input image $\boldsymbol{x}$ and the decoder output $\hat{x}$, vs the loss between weighted average image $\mu$ and $\hat{x}$; calculated over MNIST$^{te}_{9}$.

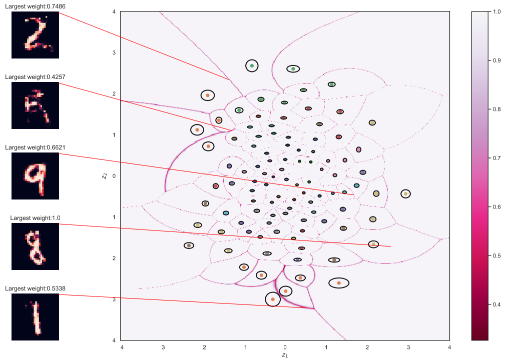

Figure 3: mini-MNIST latent space: points and ellipses show posterior distribution, and colormap shows the magnitude of largest weight. The images on the left are decoder outputs of the $\boldsymbol{z}$s chosen via stratified sampling, along with their weights.

[1] Note that the weights are normalized Gaussian likelihoods, so $w_i \in (0, 1)$, but for weights close to zero, effect of the corresponding training sample is negligible.

To visualize the latent space, we trained a VAE with $\dim(\boldsymbol{z}) = 2$. Since our infinite capacity assumption simply would not hold for MNIST$^{tr}$, we uniformly sample 10 samples per digit from MNIST$^{tr}$, and train our VAE on this "mini-MNIST" dataset. Figure 3 shows the latent space, where the points and ellipses show the posterior distribution, and the overlaid colormap shows the largest weight for a given $\boldsymbol{z}$ ($\max_i w_i(\boldsymbol{z})$). First observation is posterior distributions are converging to delta distributions which is an indicator of optimality [Rezende and Viola, 2018]. Second, for most of the latent space, maximum weight is 1, i.e. the weighted average is the nearest neighbour in latent space.

### 2.1 Role of Encoder/Decoder Capacity in Generalization

As with "generalization", "capacity" of a neural network is a hard aspect to characterize. Recent work has attempted to characterize capacity of discriminative models and datasets using algebraic topology [Guss and Salakhutdinov, 2018], but there are no equivalent results for their generative counterparts. Here, we will use number of parameters and layers as simple proxies for capacity.

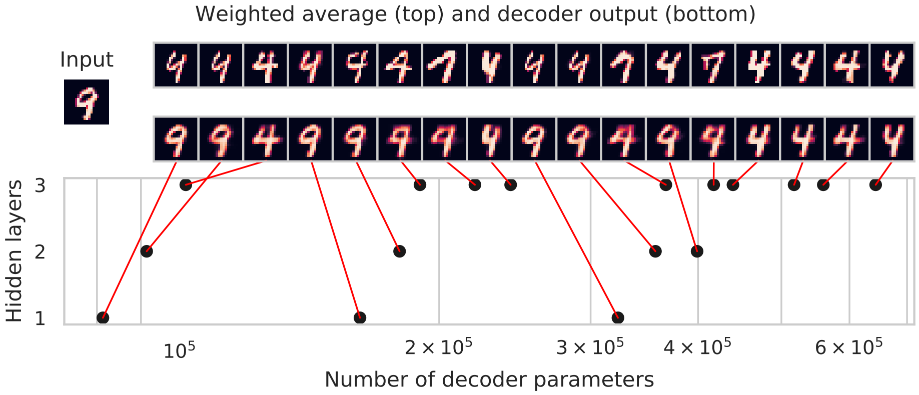

Figure 4: Decoder outputs and weighted average images for VAEs with different architectures.

In the previous experiments, we provided a comparison between a shallow and a deep VAE. Here, we do a finer grained complexity analysis, and identify regions where Theorem 1 holds. In Figure 4, we show reconstructed and weighted average images for 17 VAEs with different network architectures given an input sample from the withheld class. As VAEs get more complex, they overfit the training data therefore fail to reconstruct the unseen digit. Moreover, we observe that the reconstruction is closer to the weighted average for higher capacity networks. Another -intuitive- observation here the number of layers plays a more crucial role in complexity than number of parameters, since for VAEs with 3 hidden layers, reconstructions are more similar to weighted average, while in single hidden layer VAEs, the reconstructions match the input sample regardless of the number of parameters.

## 3 Conclusion

The theorem by Shu et al. provides an interesting viewpoint on how VAEs reason about unseen data. By investigating how much this theorem applies in practice, we uncovered some interesting properties about VAEs. In particular, we studied the connection between network capacity and generalization. Our findings show that networks with restricted capacity generalize better to out-of-training-distribution samples. For networks with sufficiently high capacity, we found that the number of training samples accountable for reconstructing a sample is often quite low, which indicates that generative capability of such VAEs are similar to a generator that employs nearest neighbour matching. We also found that VAEs with larger number of layers behave more consistently with Theorem 1 in comparison to number of parameters. For future work, it may be helpful to investigate objectives where the optimal generative model combines different *features* of training samples rather than averaging them in the high-dimensional space.

## References

Alexander Alemi, Ben Poole, Ian Fischer, Joshua Dillon, Rif A Saurous, and Kevin Murphy. Fixing a broken elbo. In *International Conference on Machine Learning*, pages 159–168, 2018.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

Olivier Bousquet, Sylvain Gelly, Ilya Tolstikhin, Carl-Johann Simon-Gabriel, and Bernhard Schoelkopf. From optimal transport to generative modeling: the VEGAN cookbook. *arXiv:1705.07642 [stat]*, May 2017. URL http://arxiv.org/abs/1705.07642. arXiv: 1705.07642.

William H Guss and Ruslan Salakhutdinov. On characterizing the capacity of neural networks using algebraic topology. *arXiv preprint arXiv:1802.04443*, 2018.

Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. *arXiv:1312.6114 [cs, stat]*, December 2013. URL http://arxiv.org/abs/1312.6114. arXiv: 1312.6114.

Danilo Jimenez Rezende and Fabio Viola. Taming VAEs. *arXiv:1810.00597 [cs, stat]*, October 2018. URL http://arxiv.org/abs/1810.00597. arXiv: 1810.00597.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of The 31st International Conference on Machine Learning*, pages 1278–1286, 2014.

Rui Shu, Hung H. Bui, Shengjia Zhao, Mykel J. Kochenderfer, and Stefano Ermon. Amortized Inference Regularization. May 2018. URL https://arxiv.org/abs/1805.08913.

Huan Wang, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. Identifying Generalization Properties in Neural Networks. *arXiv:1809.07402 [cs, stat]*, September 2018. URL http://arxiv.org/abs/1809.07402. arXiv: 1809.07402.

## A Complexity Diagrams for Different Held-out Digits

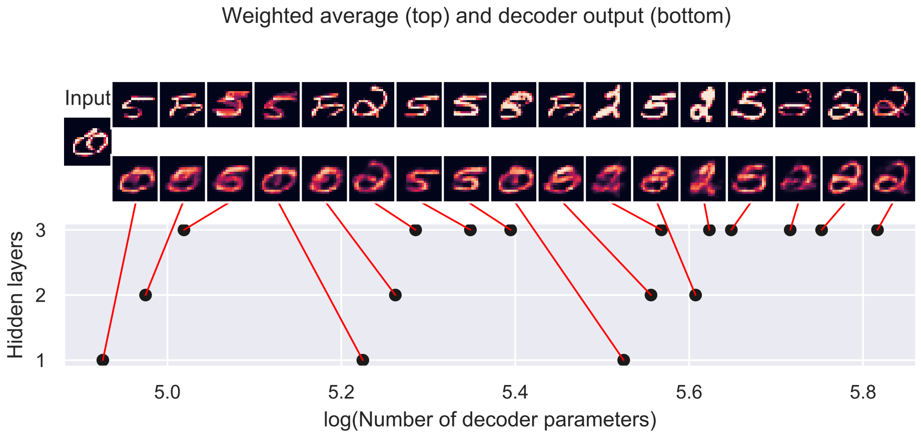

Figure 5: Decoder outputs and weighted average images for VAEs with different architectures trained on MNIST$_0^{tr}$.

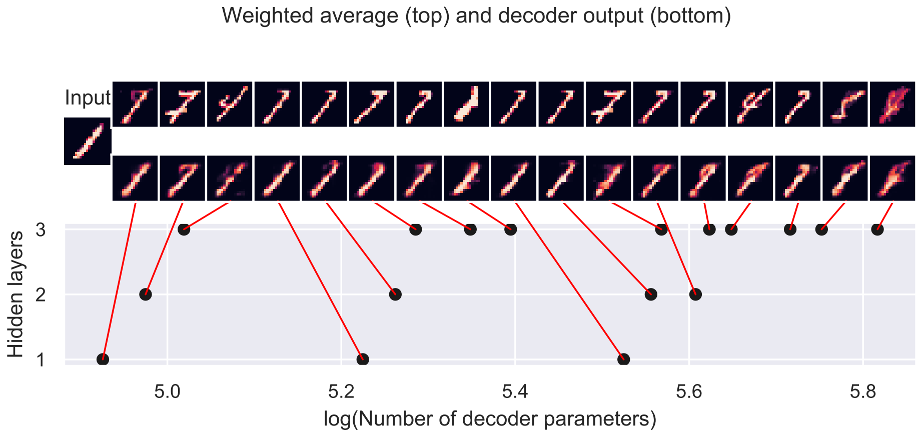

Figure 6: Decoder outputs and weighted average images for VAEs with different architectures trained on MNIST$_1^{tr}$.

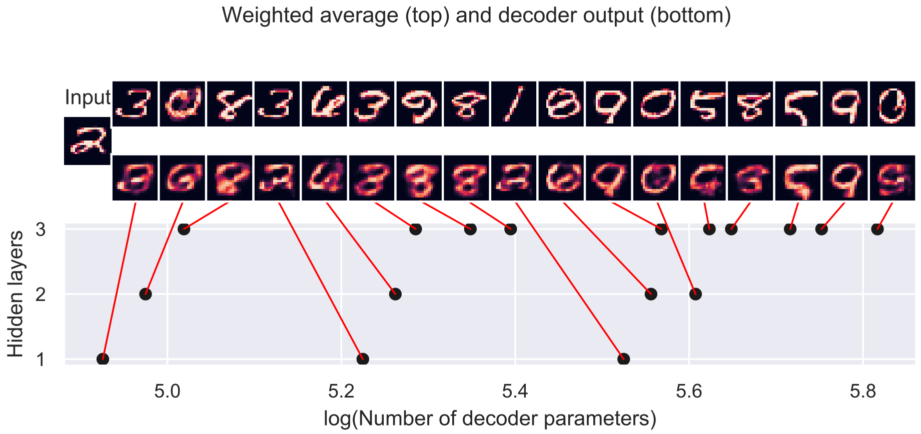

Figure 7: Decoder outputs and weighted average images for VAEs with different architectures trained on MNIST$_2^{tr}$.

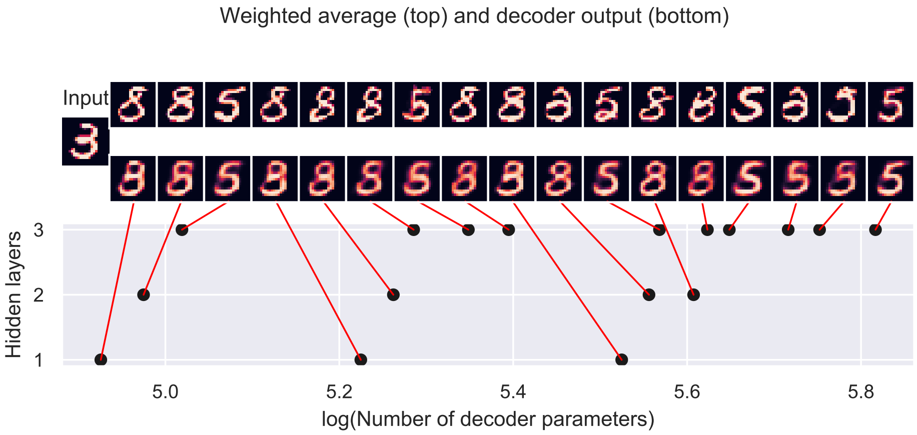

Figure 8: Decoder outputs and weighted average images for VAEs with different architectures trained on MNIST$_3^{tr}$.

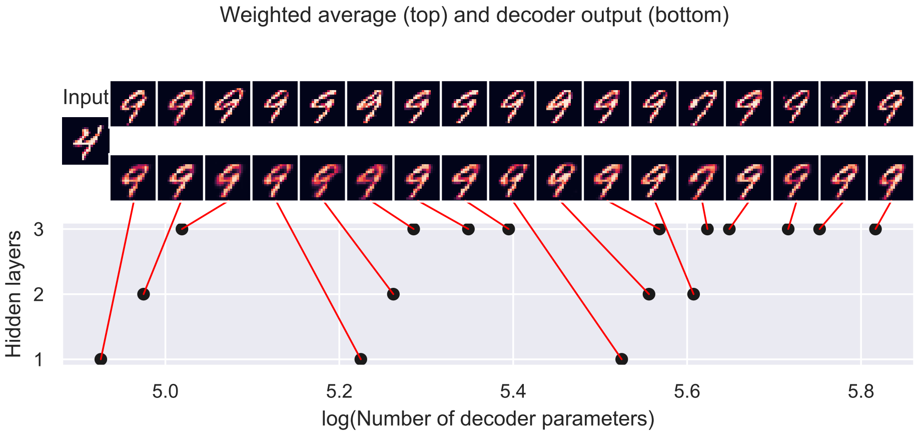

Figure 9: Decoder outputs and weighted average images for VAEs with different architectures trained on MNIST$_4^{tr}$.

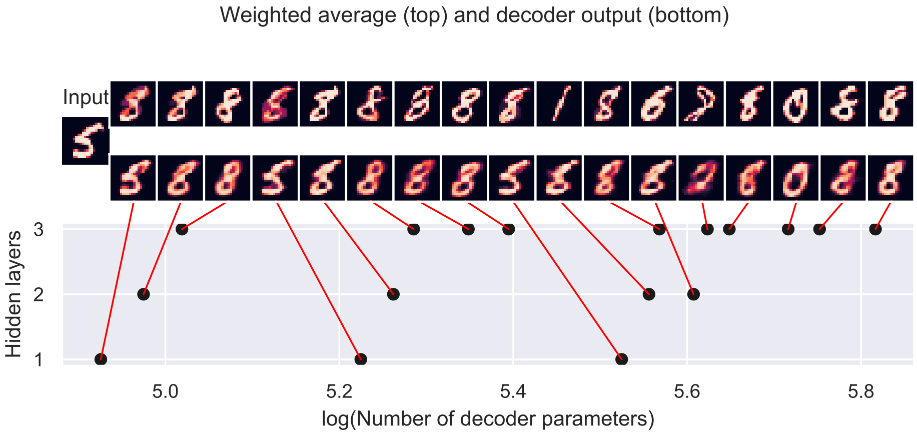

Figure 10: Decoder outputs and weighted average images for VAEs with different architectures trained on MNIST$_5^{tr}$.

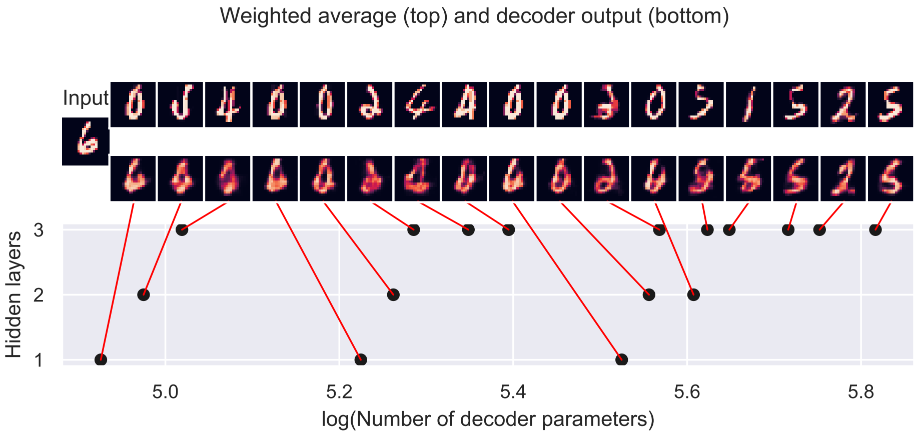

Figure 11: Decoder outputs and weighted average images for VAEs with different architectures trained on $\mathrm{MNIST}_6^{tr}$.

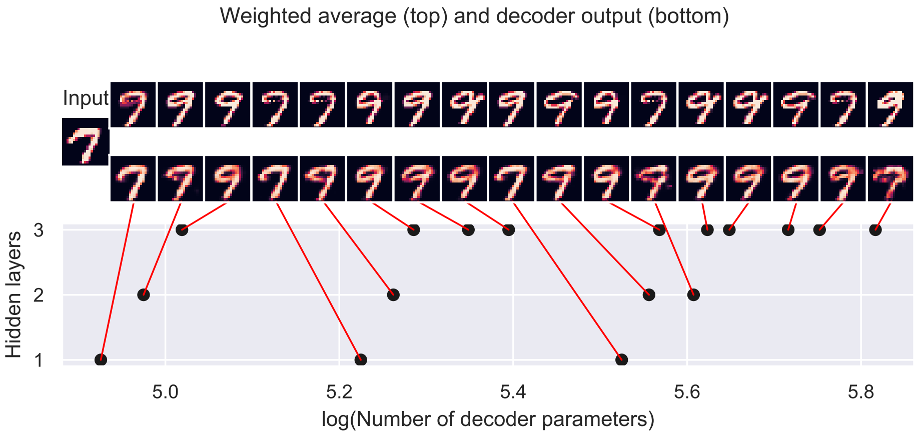

Figure 12: Decoder outputs and weighted average images for VAEs with different architectures trained on $\mathrm{MNIST}_7^{tr}$.

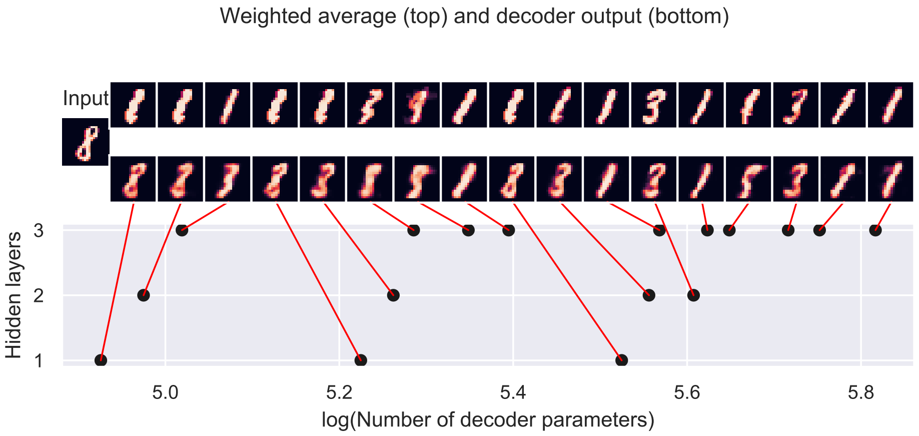

Figure 13: Decoder outputs and weighted average images for VAEs with different architectures trained on $\mathrm{MNIST}_8^{tr}$.