A Popular Generative Model: VAE

Ruixuan Xu
Department of Computer Science and Engineering
The Chinese University of Hong Kong
1155211213@link.cuhk.edu.hk

Xiangxiang Weng
Department of Computer Science and Engineering
The Chinese University of Hong Kong
1155211173@link.cuhk.edu.hk

Abstract

Variational Autoencoders (VAEs) offer a powerful approach to generative modeling and data compression by combining deep learning with probabilistic latent variable models. This report provides an overview of VAEs, detailing their architecture, training process, and theoretical underpinnings, including the derivation of the Evidence Lower Bound (ELBO) and the reparameterization trick. We explore the impact of training dataset size on VAE performance, observing that larger datasets lead to more diverse and realistic generated samples and smoother transitions within the latent space. A comparison with traditional autoencoders (AEs) highlights the advantages of VAEs in terms of generating high-quality, diverse samples and learning a more structured latent space. Empirical results demonstrate the superior performance of VAEs in image generation tasks. Finally, we discuss potential future research directions, including exploring alternative prior distributions, more complex architectures, and applications to diverse data modalities.

1 Introduction to Variational Autoencoders

Data compression and generative modeling are fundamental challenges in machine learning. Variational Autoencoders (VAEs) [1] address both by combining probabilistic generation with representation learning. Since their introduction, VAEs have inspired a wide array of generative models, including both flow-based approaches [5] and adversarial approaches such as Generative Adversarial Networks (GANs) [4].

1.1 Motivation and Background

Traditional autoencoders (AEs) compress data through an encoder-decoder architecture, mapping inputs to a lower-dimensional latent space and then reconstructing them. However, AEs can memorize training data without learning meaningful representations, leading to poor generalization and generation [6]. This occurs because the latent space is not regularized, allowing arbitrary mappings. VAEs address this by incorporating a probabilistic perspective. They encode inputs as probability distributions in the latent space, typically Gaussian, and impose regularization through a prior distribution [3, 7]. This enables effective compression and high-quality generation, capturing the inherent randomness in real-world data for diverse, realistic samples. This probabilistic foundation also provides theoretical guarantees and interpretable latent representations.

1.2 Architecture and Training

A VAE consists of two main components: an encoder and a decoder. The encoder maps the input data to a distribution in the latent space, typically parameterized by a mean and variance vector. The decoder then takes a sample from this latent distribution and reconstructs the input data.

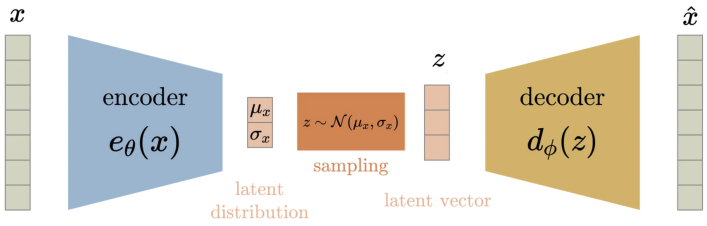


Figure 1: Architecture of VAE.

The training objective of a VAE is to maximize the evidence lower bound (ELBO), which serves as a tractable lower bound on the log-likelihood of the data. The ELBO consists of two terms: a reconstruction loss and a KL-divergence term. The reconstruction loss encourages the decoder to accurately reconstruct the input data, while the KL-divergence term regularizes the latent distribution to be close to a prior distribution, typically a standard normal distribution.

Table 1: Loss Function Details

Term	**Description**
Reconstruction Loss	$\|x - \hat{x}\|_2 = \|x - d_\phi(z)\|_2 = \|x - d_\phi(\mu_x + \sigma_x \epsilon)\|_2$
Similarity Loss	KL Divergence $= D_{KL}(\mathcal{N}(\mu_x, \sigma_x) \parallel \mathcal{N}(0, \mathbf{I}))$
Total Loss	$loss =$ reconstruction loss + similarity loss

2 Problem Setting

2.1 Probabilistic Framework

Let $x \in \mathcal{X}$ be an observed data sample and $z \in \mathcal{Z}$ be the corresponding latent variable. Variational Autoencoders (VAEs) learn two main functions:

- An encoder $q_\phi(z \mid x)$, which serves as a variational approximation to the true posterior $p(z \mid x)$.
- A decoder $p_\theta(x \mid z)$, which models the conditional likelihood of x given z.

The encoder produces parameters $\{\mu_i, \sigma_i\}$ of a Gaussian distribution for each input x_i. The overall objective is to maximize the Evidence Lower Bound (ELBO), defined as

$$\mathcal{L}_{\text{ELBO}} = \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x \mid z)\right] \; - \; D_{\text{KL}}\left(q_\phi(z \mid x) \,\middle\|\, p(z)\right), \tag{1}$$

where $p(z) = \mathcal{N}(0, I)$ denotes the isotropic Gaussian prior. Maximizing the ELBO is equivalent to minimizing its negative, thereby converting the maximization problem into a minimization of the loss function. Specifically, the ELBO comprises two terms: (1) an expected reconstruction term, $\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x \mid z)]$, and (2) the Kullback-Leibler divergence term, $D_{\text{KL}}\left(q_\phi(z \mid x) \,\middle\|\, p(z)\right)$, which regularizes $q_\phi(z \mid x)$ to be close to the prior $p(z)$ [2].

2.2 KL Divergence Derivation

Because we want the distribution of the latent space to approach the standard normal distribution that we have specified, we use KL Divergence to measure how different the two distributions are from each other [3].

For d-dimensional Gaussian distributions, the KL divergence term can be derived analytically:

$$
\begin{aligned}
D_{\mathrm{KL}}(\mathcal{N}(\mu,\sigma^2)\parallel\mathcal{N}(0,1)) &= \int_z \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(z-\mu)^2}{2\sigma^2}\right)\log\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{z^2}{2}\right)dz \\
&= \int_z \left(-\frac{(z-\mu)^2}{2\sigma^2}+\frac{z^2}{2}-\log\sigma\right)\mathcal{N}(\mu,\sigma^2)dz \\
&= -\int_z \frac{(z-\mu)^2}{2\sigma^2}\mathcal{N}(\mu,\sigma^2)dz+\int_z \frac{z^2}{2}\mathcal{N}(\mu,\sigma^2)dz-\int_z \log\sigma\mathcal{N}(\mu,\sigma^2)dz \\
&= \frac{\mathbb{E}[(z-\mu)^2]}{2\sigma^2}+\frac{\mathbb{E}[z^2]}{2}-\log\sigma \\
&= \frac{1}{2}\left(-1+\sigma^2+\mu^2-\log\sigma^2\right). \qquad (2)
\end{aligned}
$$

$$
D_{\mathrm{KL}}\left(q_\phi(z\mid x)\parallel p(z)\right) = \sum_{j=1}^{d}\frac{1}{2}\left(-1+\sigma^{(j)2}+\mu^{(j)2}-\log\sigma^{(j)2}\right). \qquad (3)
$$

For all the sample data, the total loss function is:

$$
\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{d}\frac{1}{2}\left(-1+\sigma_i^{(j)2}+\mu_i^{(j)2}-\log\sigma_i^{(j)2}\right)+\frac{1}{n}\sum_{i=1}^{n}\|x_i-\mu_i'\|^2. \qquad (4)
$$

Here, our project aims to generate colored images, so MSE is used as the reconstruction error.

2.3 Reparameterization Trick

A key innovation in Variational Autoencoders (VAEs) is the reparameterization trick, which enables gradient-based optimization through the sampling stage [1, 2]. Instead of drawing samples directly from $q_\phi(z \mid x)$, the latent variable z_i is reparameterized as

$$
z_i = \mu_i + \sigma_i \odot \epsilon_i, \quad \text{where } \epsilon_i \sim \mathcal{N}(0, I). \qquad (5)
$$

This transformation preserves the distribution of z_i while allowing gradients to pass through the network. Conceptually, the procedure replaces nondifferentiable sampling with a deterministic function of the variational parameters and an auxiliary noise variable, thereby enhancing the tractability of stochastic gradient estimators.

3 Applications and Observations in Variational Autoencoders

Variational Autoencoders (VAEs) have demonstrated a wide range of applications, including anomaly detection, image generation and restoration, and natural language processing [8, 9]. In this study, we focus on visualizing the effects of random sampling in the latent probability space, with the aim of presenting key results and insights.

The experiments were conducted using the PyTorch framework, where three separate training sessions were performed with progressively larger training datasets. In the first experiment, the VAE was trained on a single image of Pikachu. The generated samples in this case exhibited poor quality, with many outputs being unrecognizable in terms of their basic shapes and structures. This phenomenon can be attributed to an overly concentrated distribution in the latent space, as the network had access to only a single data point during training. Consequently, the model lacked sufficient variability to represent meaningful features across the latent space.

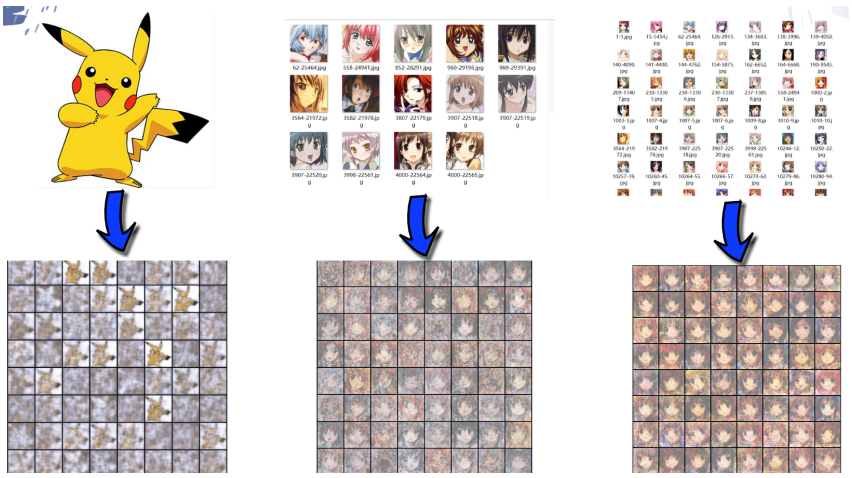


Figure 2: Generated images from the VAE.

As the training dataset grew larger (from 1 image to 14 images, then to 70 images), the latent space captured increasingly diverse information, yielding more distinct and recognizable samples. Notably, sufficiently large datasets produced smooth transitions between different faces, underscoring a central advantage of VAEs: their capacity to learn continuous and smooth latent representations conducive to generating high-quality, diverse outputs.

For our VAE implementation, we set the latent dimension to 50. The encoder uses three convolutional layers ($3 \to 64 \to 128 \to 256$ channels) with a 4×4 kernel, stride 2, and padding 1, followed by batch normalization and LeakyReLU activations. The resulting $256 \times 8 \times 8$ feature map is flattened into two fully connected layers to produce μ and $\log \sigma^2$. The decoder maps the latent code back to a $256 \times 8 \times 8$ feature map, then applies three transposed convolutional layers (kernel 4×4, stride 2, padding 1) to reduce channels from 256 to 128, 64, and finally 3. Batch normalization and ReLU activations precede a final Sigmoid to generate RGB images. Training uses MSE as the reconstruction loss and a Kullback–Leibler divergence term to regularize the latent space, with Adam ($\alpha = 10^{-3}$, batch size 128) converging in about 300 epochs. Our findings highlight how larger datasets enhance representation quality, promoting smooth interpolations and demonstrating the versatility of VAEs in image synthesis.

4 Comparison with Traditional Generative Models

Generative modeling has often leveraged standard Autoencoders (AE) for dimensionality reduction and data synthesis [10]. VAEs and AEs share foundational similarities—both employ neural network architectures for encoding and decoding—yet differ significantly in their theoretical underpinnings and empirical performance. Specifically, VAEs integrate probabilistic elements into the generative process, thereby offering a more robust framework for capturing complex data distributions.

4.1 Theoretical Framework Comparison

Table 2: Theoretical Comparison of AE and VAE

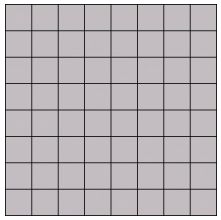
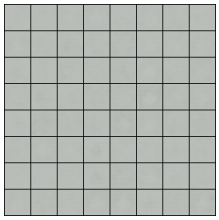
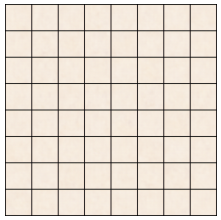
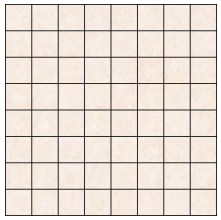
Aspect	**AE**	**VAE**
Mathematical Foundation	Neural Networks	Probability Theory
Optimization Objective	Reconstruction Error	ELBO
Latent Space Structure	Non-linear	Probabilistic
Regularization	Weight Penalties	KL Divergence

AEs learn higher-order, non-linear transformations between the input and latent representations. VAEs extend this framework by introducing a latent variable model governed by probability theory, enabling explicit control over the probabilistic structure of the latent space via the ELBO objective.

4.2 Comparison with AE Results

We additionally constructed an Autoencoder (AE) model, maintaining identical parameters to those of the Variational Autoencoder (VAE) model, but without imposing the Gaussian distribution constraint on the latent space. In this configuration, the generated data is obtained directly through reconstruction following compression. Consequently, the loss function for the AE is simplified to include only the mean squared error (MSE) term. Due to the differing definitions of the loss functions between the two models, a direct comparison of their performance based purely on the magnitude of the loss is not appropriate. Instead, we evaluate their performance qualitatively through a visual comparison of the generated outputs after training both models for an equivalent number of epochs. The results reveal that the VAE demonstrates a markedly superior performance in image generation compared to the AE:

Table 3: Comparison of AE and VAE results at different epochs.

Epoch	AE Result	Epoch	VAE Result
10		10	
130		130	
300		300	

In summary, AEs remain valuable in certain contexts, but VAEs further enhance these capabilities by introducing a principled probabilistic framework that excels at capturing the intricacies of high-dimensional data distributions. This approach not only delivers high-fidelity reconstructions but also yields latent representations with well-defined statistical properties, rendering it particularly suited for modern applications in image synthesis and representation learning.

5 Conclusion and Future Work

In this project, we have examined the capacity of Variational Autoencoders (VAEs) to integrate probabilistic modeling with deep learning for generative and reconstructive tasks. Empirical evidence indicates that VAEs outperform conventional methods such as autoencoders and principal component analysis (PCA) in both the fidelity of generated outputs and the interpretability of learned latent spaces.

Our findings further underscore the importance of dataset size: larger, more diverse training sets enable richer latent representations, producing smoother interpolations and higher-quality samples.

Future investigations could pursue several promising directions. First, exploring alternative prior distributions beyond the standard normal [11] may enrich the expressive power of VAEs and expand their generative capabilities. Second, adopting more advanced model architectures has the potential to improve performance on higher-resolution images [12]. Lastly, extending VAE-based frameworks to other data modalities, such as text and audio, would further illustrate VAEs' versatility and utility across diverse application domains.

References

[1] Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

[2] Rezende, D. J., Mohamed, S., & Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning* (pp. 1278–1286).

[3] Doersch, C. (2016). Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*.

[4] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. *Advances in Neural Information Processing Systems*, *27*.

[5] Papamakarios, G., Nalisnick, E., Rezende, D. J., Mohamed, S., & Lakshminarayanan, B. (2021). Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57), 1–64.

[6] Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, 313(5786), 504–507.

[7] Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X. M., Botvinick, M., Mohamed, S., & Lerchner, A. (2017). beta-vae: Learning basic visual concepts with a constrained variational framework. In *International conference on learning representations*.

[8] Bachman, P., & Precup, D. (2017). Variational generative stochastic networks with collaborative shaping. *arXiv preprint arXiv:1708.00805*.

[9] Sønderby, C. K., Raiko, T., Maaløe, L., Sønderby, S. K., & Winther, O. (2016). How to train deep variational autoencoders and probabilistic ladder networks. In *33rd International Conference on Machine Learning*.

[10] Kramer, M. A. (1991). Nonlinear principal component analysis using autoassociative neural networks. *AIChE Journal*, 37(2), 233–243.

[11] Tomczak, J. M., & Welling, M. (2018). VAE with a VampPrior. In *International Conference on Artificial Intelligence and Statistics* (pp. 1214–1223). PMLR.

[12] Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.