

Variational Autoencoders

Vinay Setty
vinay.j.setty@uis.no



Department of Electrical Engineering and Computer Science
University of Stavanger

January 14, 2026



Introduction

Denoising Autoencoder

Variational Autoencoder

ELBO

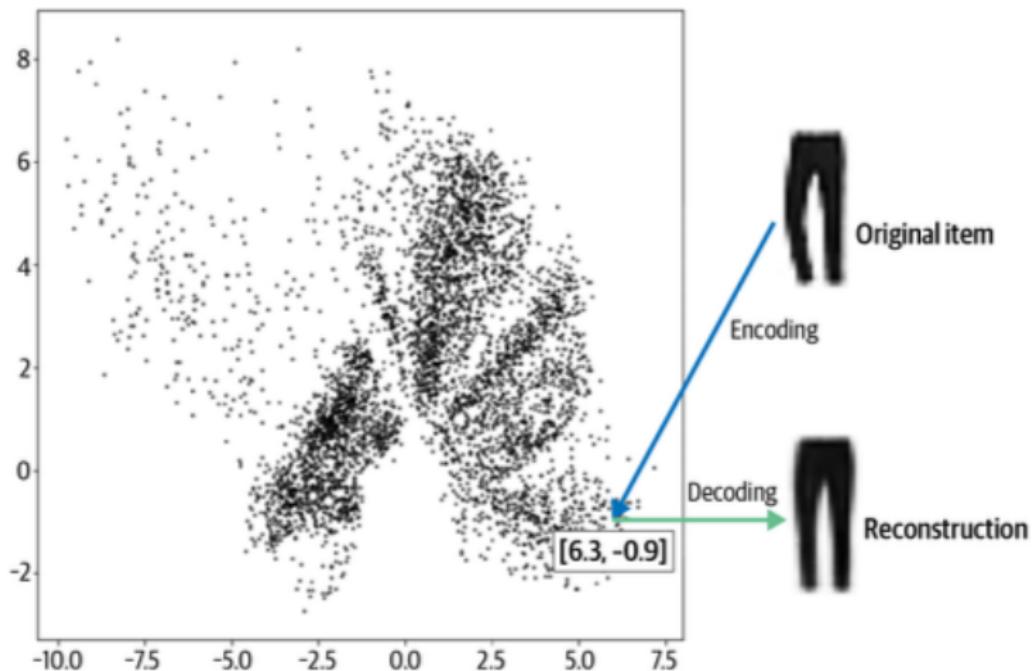
Reparameterization Trick

β -VAE

Autoencoders



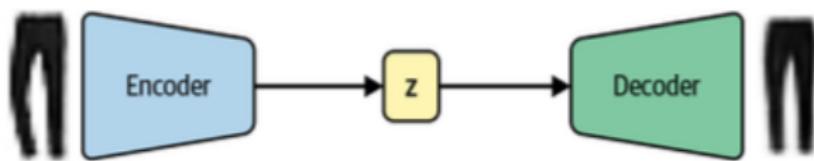
Autoencoders (cont.)



Autoencoder Definition



- ▶ Learn an identity function with a bottleneck
- ▶ Compress data into a low-dimensional latent space
- ▶ Reconstruct input from the compressed code



Source: [1]

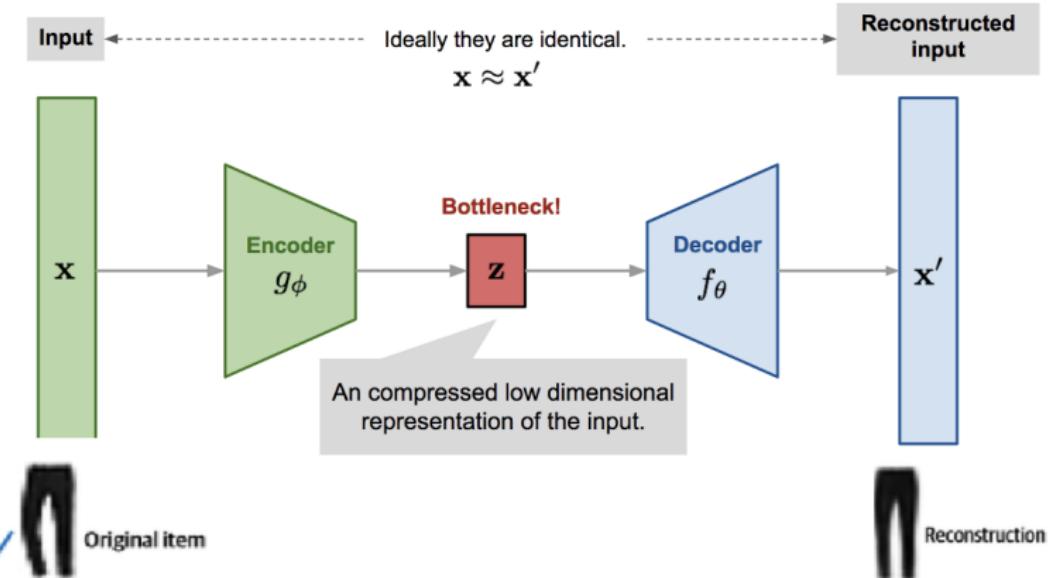
Autoencoder Definition (cont.)



Notations:

- ▶ Dataset: $\mathcal{D} = \{x^{(1)}, x^{(2)}, \dots, x^{(n)}\}$, $|\mathcal{D}| = n$
- ▶ Each sample: $x^{(i)} \in \mathbb{R}^d$, e.g. $x^{(i)} = [x_1^{(i)}, \dots, x_d^{(i)}]$
- ▶ One data point: $x \in \mathcal{D}$
- ▶ Reconstruction: x'

Autoencoder Definition (cont.)



Autoencoder Definition (cont.)



$$z = g_\phi(x)$$
$$x' = f_\theta(g_\phi(x))$$

- ▶ Bottleneck z is the learned representation
- ▶ Objective is reconstruction quality, not likelihood modeling
- ▶ Good z should capture factors useful for reconstructing x

A common choice (MSE):

$$L_{AE}(\theta, \phi) = \frac{1}{n} \sum_{i=1}^n \left(x^{(i)} - f_\theta(g_\phi(x^{(i)})) \right)^2$$

- ▶ Cross-entropy often used for binary inputs with sigmoid output
- ▶ MSE is typical when outputs are real-valued

Limitations of Autoencoders



- ▶ Autoencoders do not regularize the latent space risking **overfitting**
- ▶ Encoded data occupy **disconnected** regions in latent space
- ▶ Large empty regions ("holes") appear between encoded samples
- ▶ Latent points inside these holes do not decode to valid data
- ▶ Sampling often passes through holes and fails

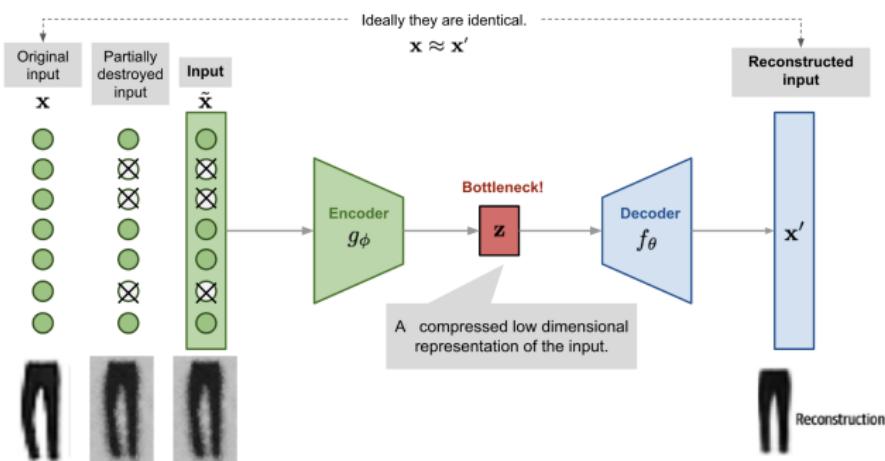
Denoising Autoencoder (DAE): Motivation



Plain AE risks learning a trivial identity map if capacity is too high.

DAE prevents this by:

- ▶ Corrupting inputs stochastically
- ▶ Training the model to reconstruct the clean x





Corruption:

$$\tilde{x} \sim M_D(\tilde{x}|x)$$

Training objective:

$$L_{DAE}(\theta, \phi) = \frac{1}{n} \sum_{i=1}^n \left(x^{(i)} - f_\theta(g_\phi(\tilde{x}^{(i)})) \right)^2$$

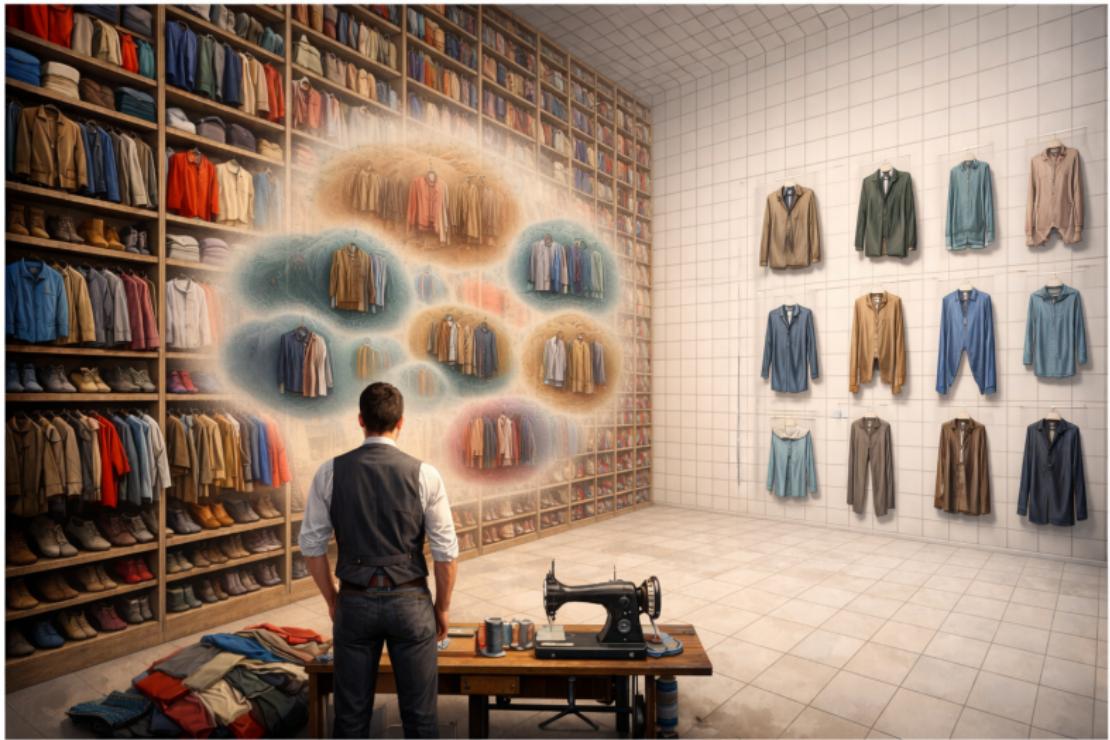
- ▶ M_D can be masking noise, Gaussian noise, etc.
- ▶ Forces the encoder to learn dependencies among input dimensions



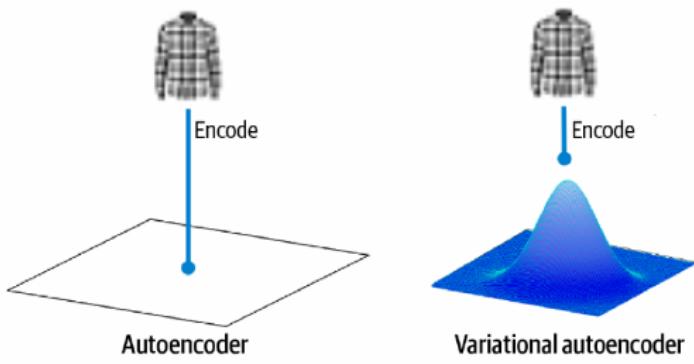
VAE is rooted in variational Bayesian inference:

- ▶ Instead of mapping $x \rightarrow z$ deterministically, map to a distribution
- ▶ Latent variables have a prior and generate data via a likelihood

VAE: Intuition



VAE: Intuition (cont.)



Source: [2]

VAE: Prior, likelihood, posterior



- ▶ Prior: $p_\theta(z)$
- ▶ Likelihood: $p_\theta(x|z)$
- ▶ Posterior: $p_\theta(z|x)$ (intractable in general)

Generation story:

1. Sample $z \sim p_{\theta^*}(z)$
2. Generate $x \sim p_{\theta^*}(x|z)$

VAE: Maximum likelihood objective



Optimize parameters to maximize probability of real data:

$$\theta^* = \arg \max_{\theta} \prod_{i=1}^n p_{\theta}(x^{(i)})$$

Often in log space:

$$\theta^* = \arg \max_{\theta} \sum_{i=1}^n \log p_{\theta}(x^{(i)})$$

Marginal likelihood:

$$p_{\theta}(x^{(i)}) = \int p_{\theta}(x^{(i)}|z)p_{\theta}(z) dz$$

- ▶ Integral over all z is expensive
- ▶ Introduce an approximation $q_{\phi}(z|x)$

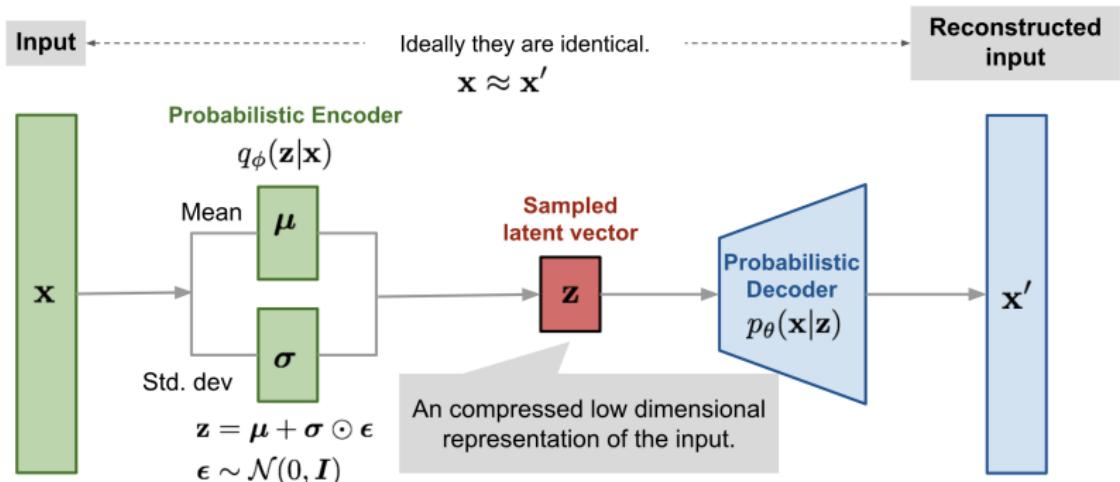


Introduce:

$$q_\phi(z|x)$$

- ▶ $q_\phi(z|x)$ approximates the intractable $p_\theta(z|x)$
- ▶ Makes inference amortized: one network predicts distribution params from x
- ▶ Structure resembles AE:
 - Decoder: $p_\theta(x|z)$
 - Encoder: $q_\phi(z|x)$

VAE Architecture





What is KL divergence?

$$\text{KL}(q \parallel p) = \mathbb{E}_q \left[\log \frac{q(z)}{p(z)} \right]$$

- ▶ Measures how much a distribution q deviates from a reference distribution p
- ▶ Always non-negative
- ▶ Equal to zero only when $q = p$
- ▶ Not symmetric and not a distance

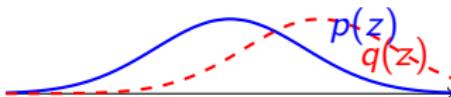


Intuition

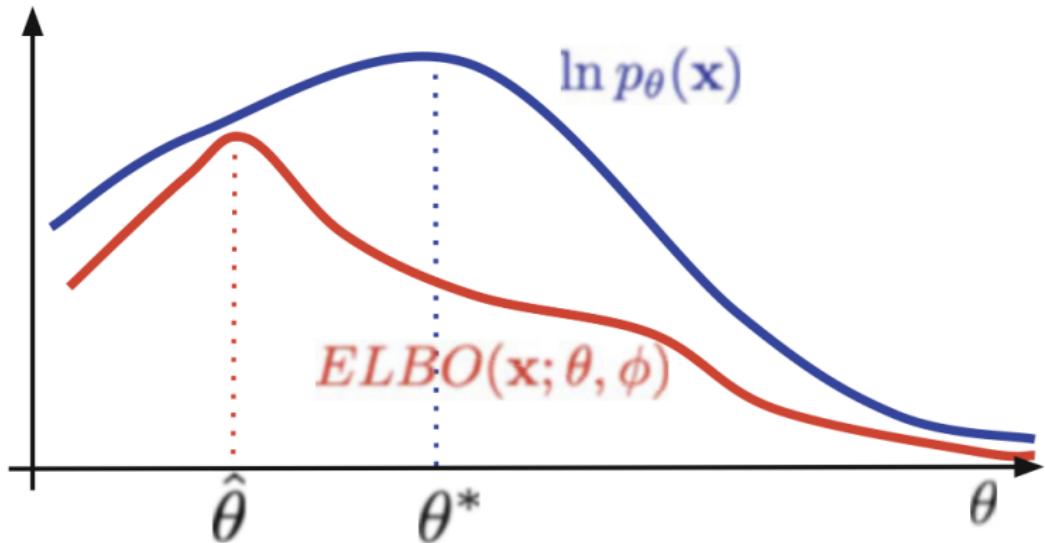
- ▶ Think of p as the rule you agreed to follow
- ▶ q is what you actually do
- ▶ KL divergence is the *penalty for breaking the rule*
- ▶ Larger mismatch means higher cost

Example (VAE context)

- ▶ $p(z) = \mathcal{N}(0, I)$ is the prior
- ▶ $q(z|x)$ is the encoder distribution
- ▶ KL penalizes latent codes that drift too far from zero or become too narrow



Evidence Lower Bound (ELBO)



Source: [3]

ELBO: Step-by-step derivation (1)

Start:

$$D_{KL}(q_\phi(z|x) \| p_\theta(z|x)) = \int q_\phi(z|x) \log \frac{q_\phi(z|x)}{p_\theta(z|x)} dz$$

Use $p(z|x) = \frac{p(z,x)}{p(x)}$:

$$= \int q_\phi(z|x) \log \frac{q_\phi(z|x)p_\theta(x)}{p_\theta(z,x)} dz$$

ELBO: Step-by-step derivation (2)

$$= \int q_\phi(z|x) \left(\log p_\theta(x) + \log \frac{q_\phi(z|x)}{p_\theta(z,x)} \right) dz$$

Because $\int q(z|x)dz = 1$:

$$= \log p_\theta(x) + \int q_\phi(z|x) \log \frac{q_\phi(z|x)}{p_\theta(z,x)} dz$$

Because $p(z, x) = p(x|z)p(z)$:

$$= \log p_\theta(x) + \int q_\phi(z|x) \log \frac{q_\phi(z|x)}{p_\theta(x|z)p_\theta(z)} dz$$

ELBO: Step-by-step derivation (3)

Convert integral to expectation:

$$= \log p_\theta(x) + \mathbb{E}_{z \sim q_\phi(z|x)} \left[\log \frac{q_\phi(z|x)}{p_\theta(z)} - \log p_\theta(x|z) \right]$$

So:

$$D_{KL}(q_\phi(z|x) \| p_\theta(z|x)) = \log p_\theta(x) + D_{KL}(q_\phi(z|x) \| p_\theta(z)) - \mathbb{E}_{z \sim q_\phi(z|x)} \log p_\theta(x|z)$$

ELBO: Rearranging to the bound

Rearrange:

$$\log p_\theta(x) - D_{KL}(q_\phi(z|x) \| p_\theta(z|x)) = \mathbb{E}_{z \sim q_\phi(z|x)} \log p_\theta(x|z) - D_{KL}(q_\phi(z|x) \| p_\theta(z))$$

Define VAE loss (as given):

$$\begin{aligned} L_{VAE}(\theta, \phi) &= -\log p_\theta(x) + D_{KL}(q_\phi(z|x) \| p_\theta(z|x)) \\ &= -\mathbb{E}_{z \sim q_\phi(z|x)} \log p_\theta(x|z) + D_{KL}(q_\phi(z|x) \| p_\theta(z)) \end{aligned}$$

- ▶ First term: reconstruction (negative log-likelihood)
- ▶ Second term: regularization toward prior

ELBO: Why “lower bound”



$$-L_{VAE} = \log p_\theta(x) - D_{KL}(q_\phi(z|x) \| p_\theta(z|x))$$

log

$p_\theta(x)$ because KL divergence is non-negative.

- ▶ Maximizing ELBO is maximizing a lower bound on $\log p_\theta(x)$
- ▶ Also minimizes the gap between approximate and true posterior

Reparameterization trick: The problem



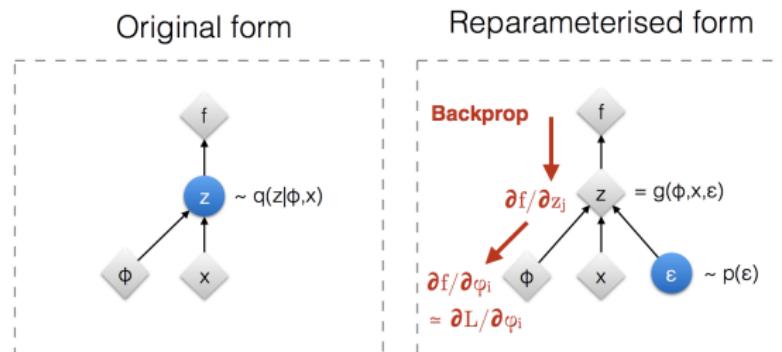
The reconstruction term uses sampling:

$$z \sim q_\phi(z|x)$$

Sampling is stochastic, so naive backpropagation does not work.

Goal:

- ▶ Express sampling as a deterministic transform of noise
- ▶ Push randomness into an auxiliary variable independent of ϕ



: Deterministic node



: Random node

[Kingma, 2013]

[Bengio, 2013]

[Kingma and Welling 2014]

Reparameterization trick: General form



Write the random variable as:

$$z = T_\phi(x, \epsilon)$$

where:

- ▶ ϵ is auxiliary independent noise
- ▶ T_ϕ is a differentiable transformation parameterized by ϕ

Reparameterization trick: Gaussian case (as given)

Assume diagonal Gaussian:

$$z \sim q_{\phi}(z|x^{(i)}) = \mathcal{N}(z; \mu^{(i)}, \sigma^{2(i)}I)$$

Then:

$$z = \mu + \sigma \odot \epsilon, \quad \text{where } \epsilon \sim \mathcal{N}(0, I)$$

⊙ denotes element-wise product.

- ▶ Learn μ and σ via encoder network outputs
- ▶ Gradients pass through μ, σ while randomness stays in ϵ



A representation is disentangled if:

- ▶ Each latent variable is sensitive to one generative factor
- ▶ Relatively invariant to other factors

Examples of factors in faces:

- ▶ skin color, hair color, hair length, emotion, glasses, etc.

β -VAE increases pressure toward factorized, efficient latent codes.

β -VAE: Constrained objective (as given)



Optimization problem:

$$\max_{\phi, \theta} \mathbb{E}_{x \sim \mathcal{D}} \left[\mathbb{E}_{z \sim q_\phi(z|x)} \log p_\theta(x|z) \right]$$

subject to:

$$D_{KL}(q_\phi(z|x) \| p_\theta(z)) < \delta$$

β -VAE: Lagrangian form (as given)



Define Lagrangian:

$$F(\theta, \phi, \beta) = \mathbb{E}_{z \sim q_\phi(z|x)} \log p_\theta(x|z) - \beta (D_{KL}(q_\phi(z|x) \| p_\theta(z)) - \delta)$$

Expand:

$$= \mathbb{E}_{z \sim q_\phi(z|x)} \log p_\theta(x|z) - \beta D_{KL}(q_\phi(z|x) \| p_\theta(z)) + \beta \delta$$

Lower bound:

$$\geq \mathbb{E}_{z \sim q_\phi(z|x)} \log p_\theta(x|z) - \beta D_{KL}(q_\phi(z|x) \| p_\theta(z))$$

β -VAE: Loss function (as given)

$$L_{\beta VAE}(\phi, \beta) = -\mathbb{E}_{z \sim q_\phi(z|x)} \log p_\theta(x|z) + \beta D_{KL}(q_\phi(z|x) \| p_\theta(z))$$

- ▶ $\beta = 1$ recovers VAE
- ▶ $\beta > 1$ strengthens the bottleneck constraint
- ▶ Tradeoff: reconstruction quality vs disentanglement pressure

Bibliography



- [1] L. Weng, "From autoencoder to beta-vae," *lilianweng.github.io*, 2018. [Online]. Available: <https://lilianweng.github.io/posts/2018-08-12-vae/>.
- [2] D. Foster, *Generative deep learning*. " O'Reilly Media, Inc.", 2022.
- [3] J. M. Tomczak, *Deep Generative Modeling*, 2nd ed. Springer Cham, 2024, ISBN: 978-3-031-64086-5. DOI: [10.1007/978-3-031-64087-2](https://doi.org/10.1007/978-3-031-64087-2). [Online]. Available: <https://link.springer.com/book/10.1007/978-3-031-64087-2>.