Probability Language for Variational Autoencoders

N-20251008

§1.1 Why probability enters generative modeling

A deterministic autoencoder learns a pair of functions

$$x \longmapsto h \longmapsto \widehat{x}.$$

This is useful for compression and representation learning, but it is not yet a probabilistic generative model. A generative model should tell us how likely different data points are and how to sample new ones. Therefore the mathematical object is not merely a map, but a probability distribution.

The VAE begins from the following change of viewpoint:

a data point is not only reconstructed;
it is regarded as a sample from an unknown distribution.

If X denotes the random data point, then the ideal object is the data distribution $p_{\text{data}}(x)$. A model tries to build a tractable family $p_\theta(x)$ and choose θ so that p_θ resembles the data distribution.

The slogan for this chapter is:

generation means sampling from a learned distribution.

Before discussing encoders, decoders, or neural networks, we first need the probability language in which the VAE objective is written.

§1.2 Random variables, distributions, and densities

A random variable X is a measurable quantity whose value is uncertain. Its distribution tells us the probability of events such as $X \in A$. When X has a density $p(x)$ with respect to Lebesgue measure, probabilities are computed by integration:

$$\mathbb{P}(X \in A) = \int_A p(x)\,dx.$$

A density is not itself a probability. The value $p(x)$ may even be larger than 1. What matters is the integral over a region.

The expectation of a function $f(X)$ is

$$\mathbb{E}[f(X)] = \int f(x)p(x)\,dx.$$

For a discrete variable, the corresponding formula is a sum:

$$\mathbb{E}[f(X)] = \sum_x f(x)p(x).$$

The formulas look similar, but the measure matters. In VAEs, latent variables are usually continuous, so integrals over z appear everywhere.

§1.3 Joint, marginal, and conditional distributions

VAEs use latent variables, so we need distributions involving both visible variables and hidden variables. Let X be observed data and Z be a latent variable. Their joint density is

$$p(x, z).$$

The joint density can be factored in two ways:

$$p(x, z) = p(x \mid z)p(z) = p(z \mid x)p(x).$$

The marginal density of X is obtained by integrating out Z:

$$p(x) = \int p(x, z)\, dz.$$

This operation is called marginalization. It means that all possible hidden explanations z are summed or integrated away.

The conditional density is given by Bayes' rule:

$$p(z \mid x) = \frac{p(x, z)}{p(x)} = \frac{p(x \mid z)p(z)}{\int p(x \mid z')p(z')\, dz'}.$$

The three words should be kept distinct:

joint	describe x and z together
marginal	hide one variable by integration
conditional	update one variable after observing the other

§1.4 Bayes' rule as inversion of a generative story

A latent variable model usually starts with a generative story:

$$z \sim p(z), \qquad x \sim p_\theta(x \mid z).$$

This is the easy direction: first choose a hidden cause z, then generate an observation x.

Inference goes in the opposite direction. Given an observation x, we ask which latent values z could have produced it. The posterior is

$$p_\theta(z \mid x) = \frac{p_\theta(x \mid z)p(z)}{p_\theta(x)}.$$

The denominator

$$p_\theta(x) = \int p_\theta(x \mid z)p(z)\, dz$$

is called the evidence or marginal likelihood.

This is the first obstacle behind VAEs. The generative direction is easy to write, but the inverse direction requires an integral that is usually not tractable.

§1.5 Gaussian distributions

The standard VAE uses Gaussian latent variables. The one-dimensional Gaussian density is

$$\mathcal{N}(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right).$$

Here μ is the mean and σ^2 is the variance.

The multivariate Gaussian density is

$$\mathcal{N}(z;\mu,\Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(z-\mu)^T\Sigma^{-1}(z-\mu)\right).$$

The matrix Σ is the covariance matrix. It measures not only the variance of each coordinate, but also how coordinates co-vary.

The diagonal case is especially important:

$$\Sigma = \operatorname{diag}(\sigma_1^2,\dots,\sigma_d^2).$$

Then the coordinates are independent under the Gaussian, and the density factorizes:

$$\mathcal{N}(z;\mu,\operatorname{diag}(\sigma^2)) = \prod_{j=1}^{d} \mathcal{N}(z_j;\mu_j,\sigma_j^2).$$

This is why VAE encoders often output a vector of means and a vector of variances rather than a full covariance matrix.

§1.6 Expectation and Monte Carlo approximation

Many VAE formulas contain expectations that cannot be computed exactly. If

$$z \sim q(z),$$

then

$$\mathbb{E}_{z\sim q}[f(z)] = \int f(z)q(z)\,dz.$$

A Monte Carlo approximation uses samples $z^{(1)},\dots,z^{(L)} \sim q$:

$$\mathbb{E}_{z\sim q}[f(z)] \approx \frac{1}{L}\sum_{\ell=1}^{L} f(z^{(\ell)}).$$

This is not a deterministic quadrature rule. It is a random approximation whose error decreases statistically. The key benefit is that the dimension of z can be large: the same formula still makes sense, while grid integration becomes impossible.

§1.7 KL divergence

The Kullback–Leibler divergence compares two distributions q and p:

$$D_{\mathrm{KL}}(q\|p) = \mathbb{E}_q\left[\log\frac{q(Z)}{p(Z)}\right].$$

For continuous densities, this means

$$D_{\mathrm{KL}}(q\|p) = \int q(z)\log\frac{q(z)}{p(z)}\,dz.$$

For discrete distributions, replace the integral by a sum.

The order matters. Usually

$$D_{\mathrm{KL}}(q\|p) \neq D_{\mathrm{KL}}(p\|q).$$

So KL divergence is not a metric distance.

A useful interpretation is coding loss. If data are really drawn from q, but one uses a code optimized for p, then KL measures the expected extra log-loss. In variational inference, we use it to measure how expensive it is to replace the true posterior by an approximate posterior.

The fundamental inequality is

$$D_{\mathrm{KL}}(q\|p) \geq 0,$$

with equality exactly when $q = p$ almost everywhere. This nonnegativity is what turns the ELBO into a lower bound.

§1.8 Jensen's inequality

The other basic ingredient is Jensen's inequality. Since log is concave,

$$\log \mathbb{E}[Y] \geq \mathbb{E}[\log Y]$$

for positive random variables Y.

This inequality is the analytic reason variational lower bounds exist. In the VAE derivation, we will write a difficult likelihood as

$$\log \int q(z) \frac{p(x,z)}{q(z)}\, dz = \log \mathbb{E}_q \left[\frac{p(x,Z)}{q(Z)} \right]$$

and then move the logarithm inside the expectation:

$$\log \mathbb{E}_q \left[\frac{p(x,Z)}{q(Z)} \right] \geq \mathbb{E}_q \left[\log \frac{p(x,Z)}{q(Z)} \right].$$

This is exactly the shape of the ELBO.

§1.9 The next abstraction

The VAE is built from probability, not merely from neural network architecture. The important vocabulary is:

term	role in VAE
$p(z)$	prior over latent variables
$p_\theta(x \mid z)$	decoder likelihood
$p_\theta(x) = \int p_\theta(x \mid z) p(z)\, dz$	marginal likelihood
$p_\theta(z \mid x)$	true posterior
$q_\phi(z \mid x)$	approximate posterior / encoder
$D_{\mathrm{KL}}(q\|p)$	penalty for replacing p by q
$\mathbb{E}_q[f]$	average under the approximate posterior

The next note will use this language to describe a latent variable model. The main tension will be simple: the model is easy to sample from, but hard to invert.