Understanding Deep Learning

Chapter 17: Variational autoencoders

This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.

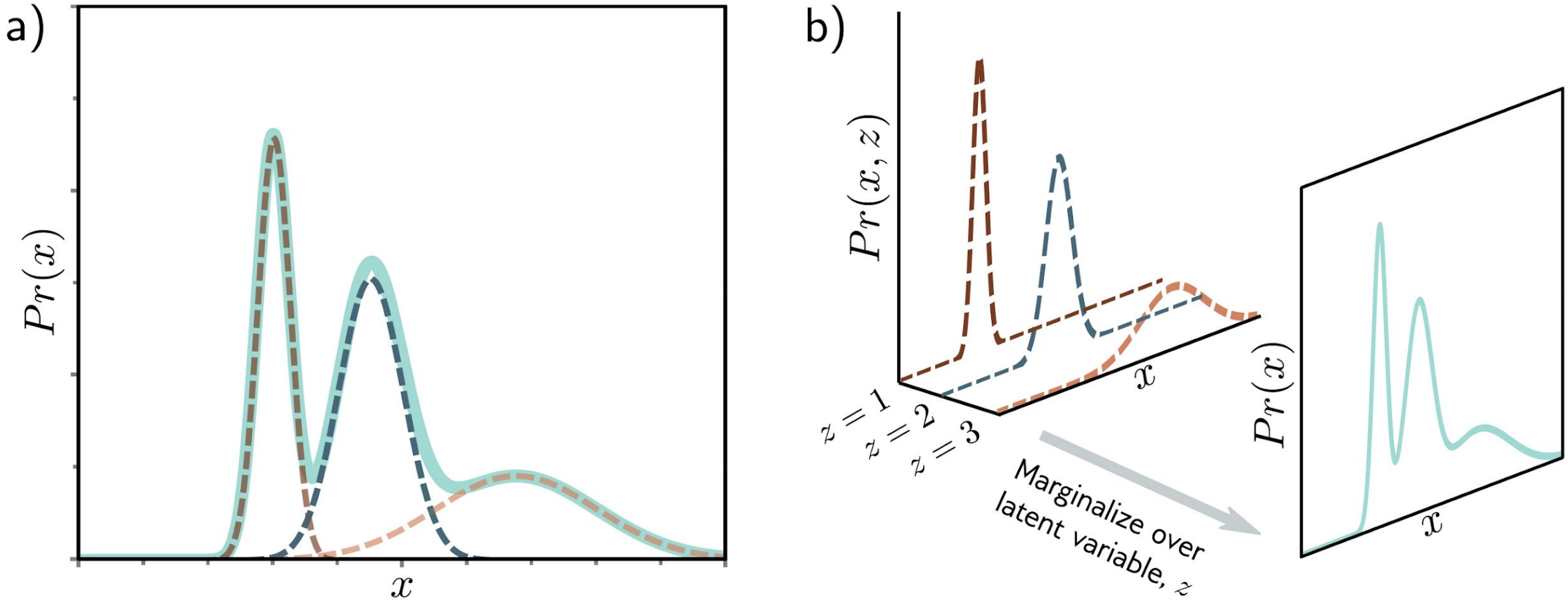


Figure 17.1 Mixture of Gaussians (MoG). a) The MoG describes a complex probability distribution (cyan curve) as a weighted sum of Gaussian components (dashed curves). b) This sum is the marginalization of the joint density $Pr(x, z)$ between the continuous observed data x and a discrete latent variable z.

This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.

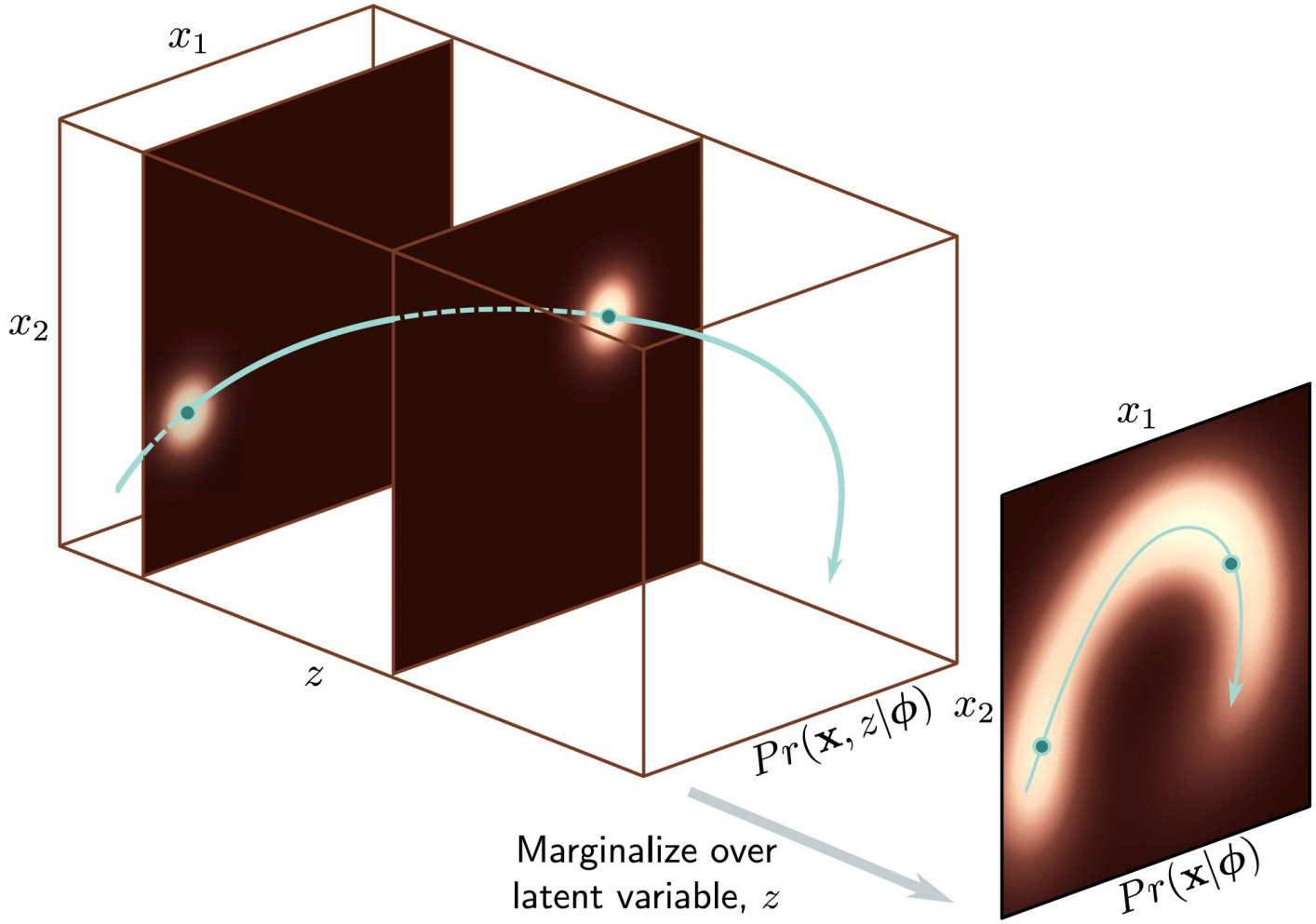


Figure 17.2 Nonlinear latent variable model. A complex 2D density $Pr(\mathbf{x})$ (right) is created as the marginalization of the joint distribution $Pr(\mathbf{x}, z)$ (left) over the latent variable z; to create $Pr(\mathbf{x})$, we integrate the 3D volume over the dimension z. For each z, the distribution over $\mathbf{x}$ is a spherical Gaussian (two slices shown) with a mean $\mathbf{f}[z, \boldsymbol{\phi}]$ that is a nonlinear function of z and depends on parameters $\boldsymbol{\phi}$. The distribution $Pr(\mathbf{x})$ is a weighted sum of these Gaussians.

This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.

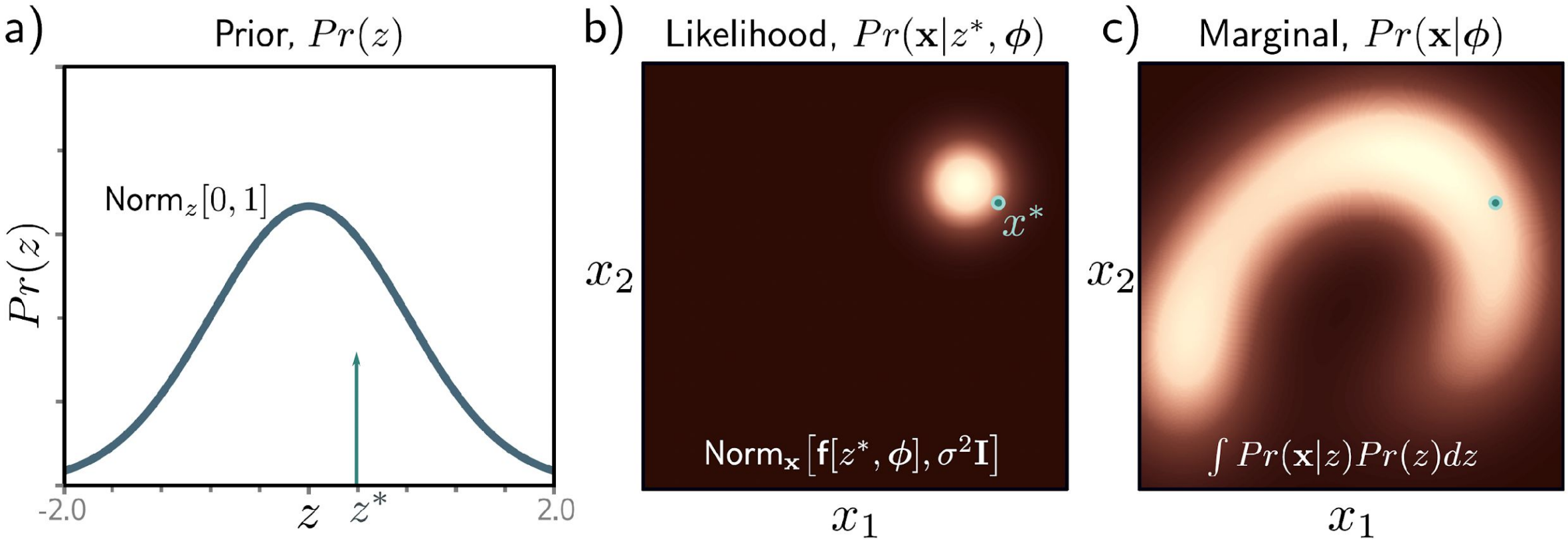
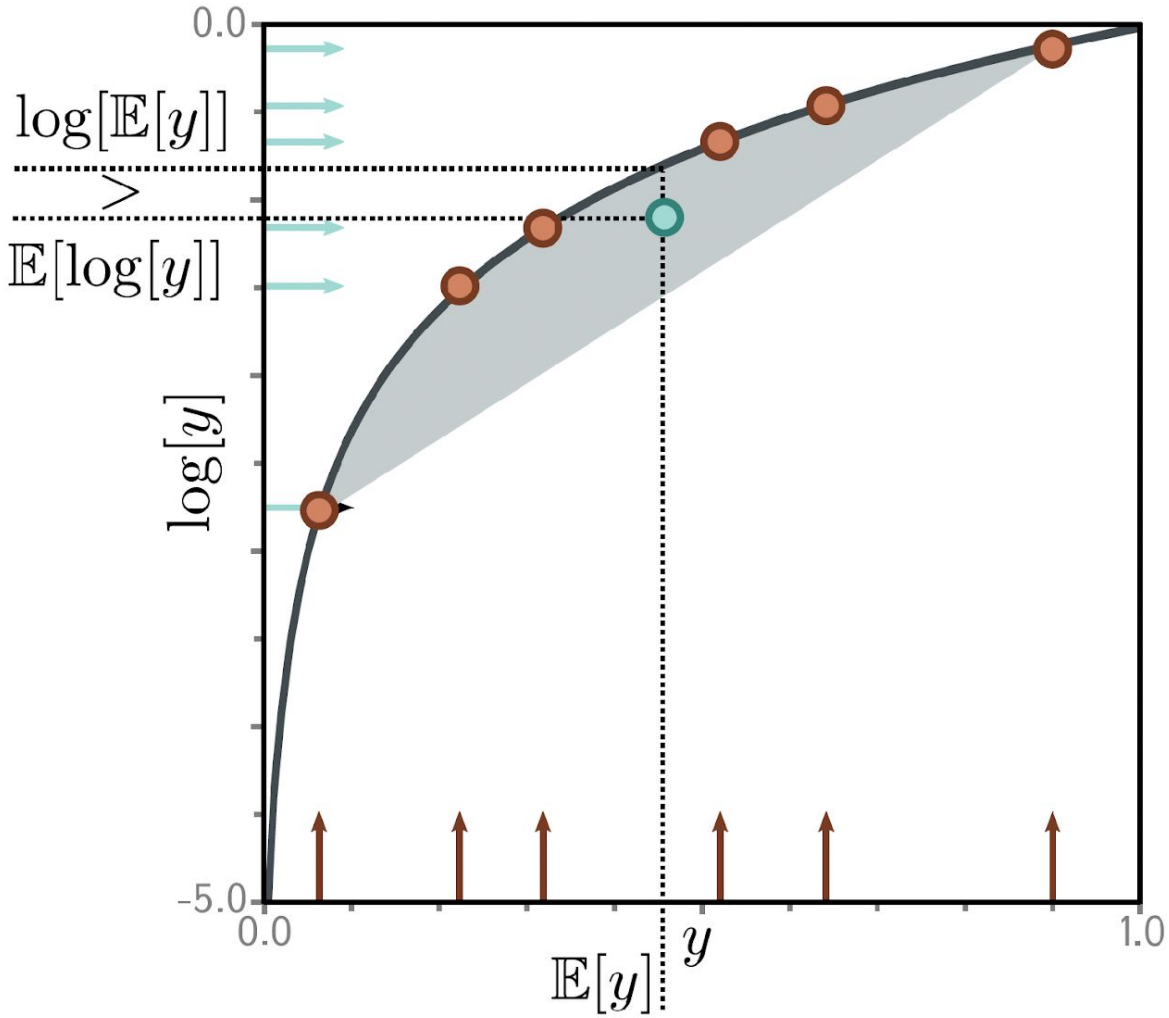


Figure 17.3 Generation from nonlinear latent variable model. a) We draw a sample z^* from the prior probability $Pr(z)$ over the latent variable. b) A sample $\mathbf{x}^*$ is then drawn from $Pr(\mathbf{x}|z^*, \boldsymbol{\phi})$. This is a spherical Gaussian with a mean that is a nonlinear function $\mathbf{f}[\bullet, \boldsymbol{\phi}]$ of z^* and a fixed variance $\sigma^2\mathbf{I}$. c) If we repeat this process many times, we recover the density $Pr(\mathbf{x}|\boldsymbol{\phi})$.

This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.

Figure 17.4 Jensen's inequality (discrete case). The logarithm (black curve) is a concave function; you can draw a straight line between any two points on the curve, and this line will always lie underneath it. It follows that any convex combination (weighted sum with positive weights that sum to one) of the six points on the log function must lie in the gray region under the curve. Here, we have weighted the points equally (i.e., taken the mean) to yield the cyan point. Since this point lies below the curve, $\log[\mathbb{E}[y]] > \mathbb{E}[\log[y]]$.



This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.

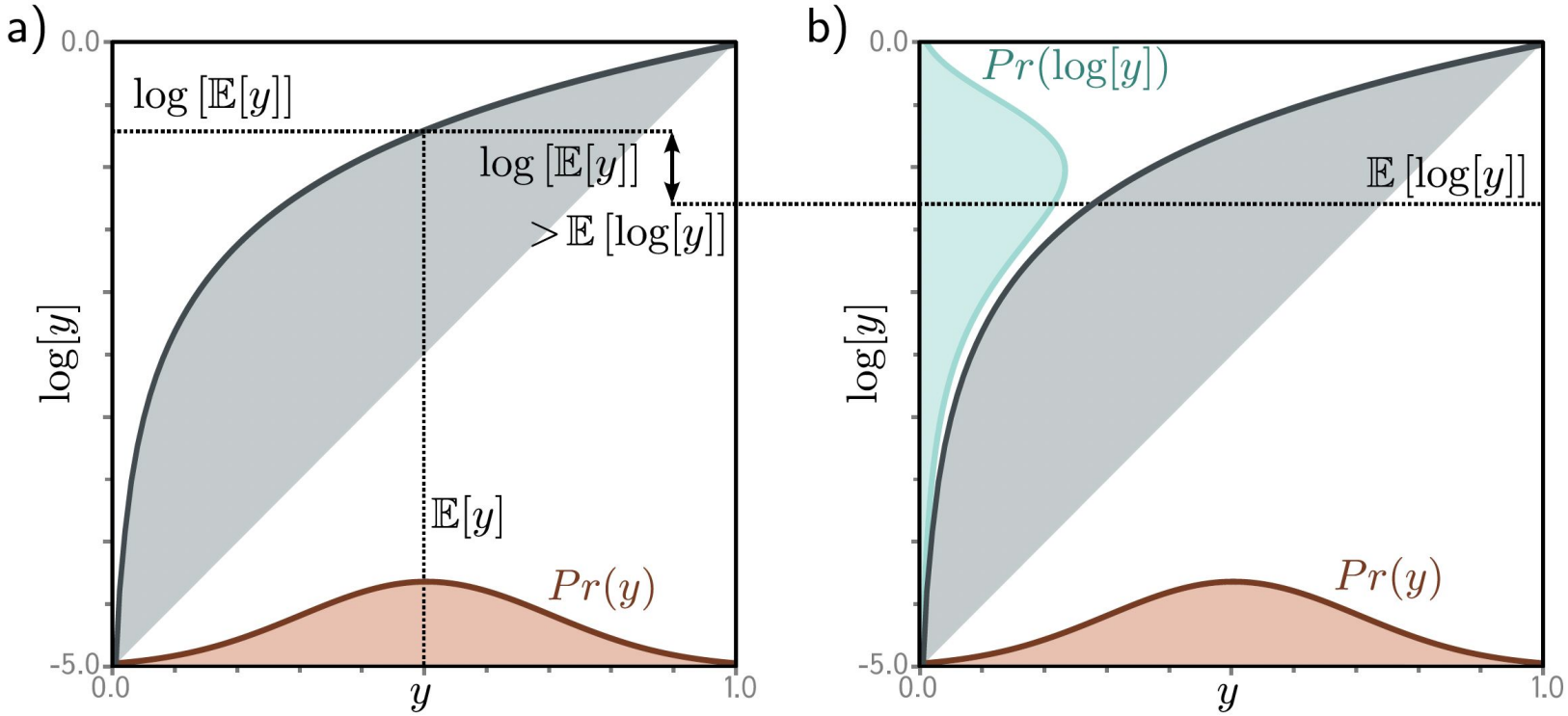


Figure 17.5 Jensen's inequality (continuous case). For a concave function, computing the expectation of a distribution $Pr(y)$ and passing it through the function gives a result greater than or equal to transforming the variable y by the function and then computing the expectation of the new variable. In the case of the logarithm, we have $\log[\mathbb{E}[y]] \geq \mathbb{E}[\log[y]]$. The left-hand side of the figure corresponds to the left-hand side of this inequality and the right-hand side of the figure to the right-hand side. One way of thinking about this is to consider that we are taking a convex combination of the points in the orange distribution defined over $y \in [0, 1]$. By the logic of figure 17.4, this must lie under the curve. Alternatively, we can think about the concave function as compressing the high values of y relative to the low values, so the expected value is lower when we pass y through the function first.

This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.

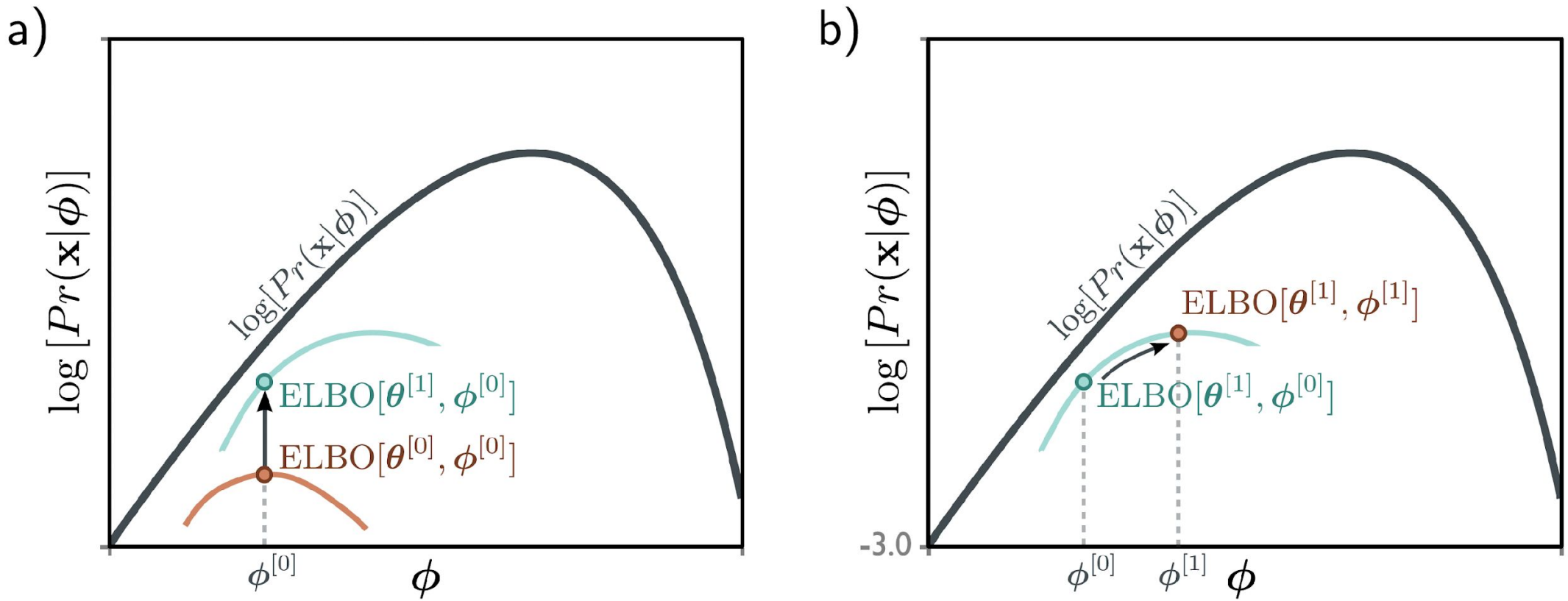


Figure 17.6 Evidence lower bound (ELBO). The goal is to maximize the log-likelihood $\log[Pr(\mathbf{x}|\boldsymbol{\phi})]$ (black curve) with respect to the parameters $\boldsymbol{\phi}$. The ELBO is a function that lies everywhere below the log-likelihood. It is a function of both $\boldsymbol{\phi}$ and a second set of parameters $\boldsymbol{\theta}$. For fixed $\boldsymbol{\theta}$, we get a function of $\boldsymbol{\phi}$ (two colored curves for different values of $\boldsymbol{\theta}$). Consequently, we can increase the log-likelihood by either improving the ELBO with respect to a) the new parameters $\boldsymbol{\theta}$ (moving from colored curve to colored curve) or b) the original parameters $\boldsymbol{\phi}$ (moving along the current colored curve).

This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.

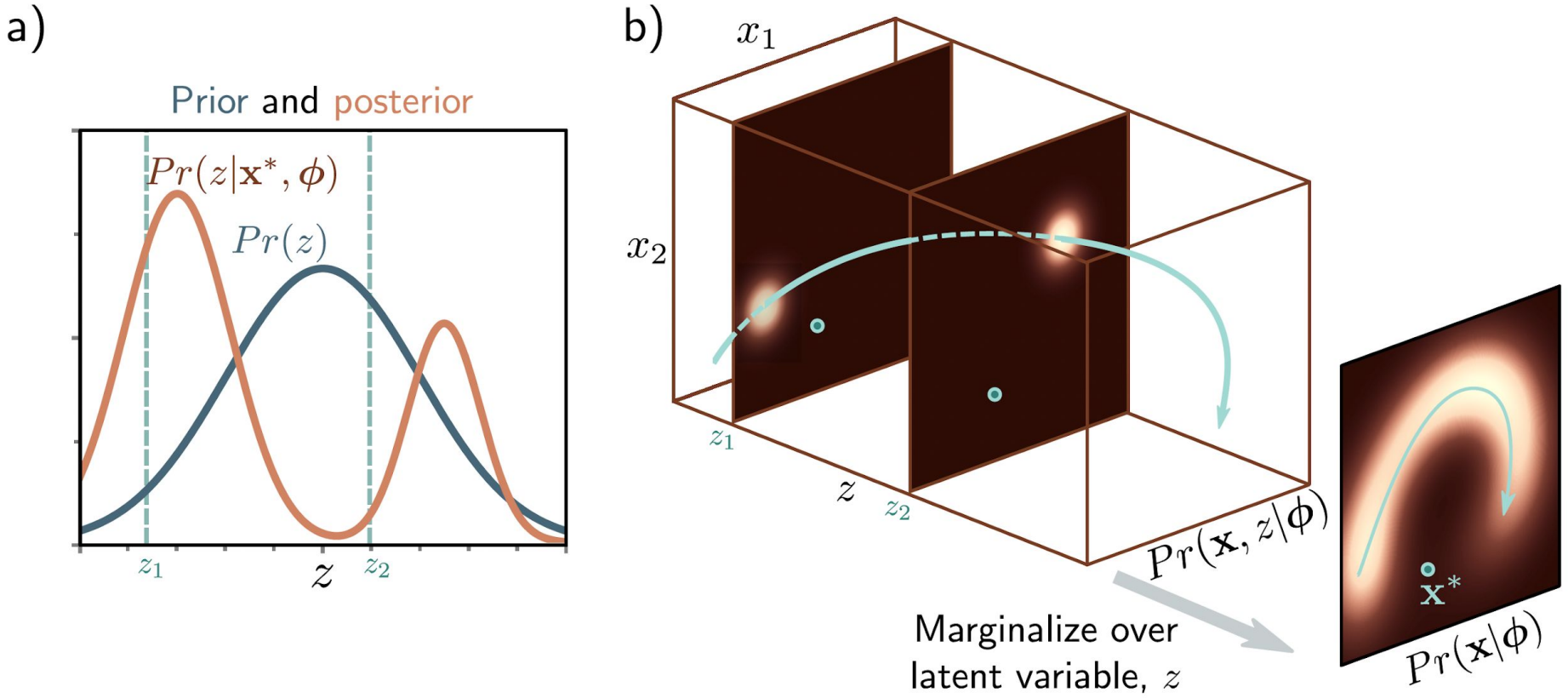


Figure 17.7 Posterior distribution over latent variable. a) The posterior distribution $Pr(z|\mathbf{x}^*, \boldsymbol{\phi})$ is the distribution over the values of the latent variable z that could be responsible for a data point $\mathbf{x}^*$. We calculate this via Bayes' rule $Pr(z|\mathbf{x}^*, \boldsymbol{\phi}) \propto Pr(\mathbf{x}^*|z, \boldsymbol{\phi})Pr(z)$. b) We compute the first term on the right-hand side (the likelihood) by assessing the probability of $\mathbf{x}^*$ against the symmetric Gaussian associated with each value of z. Here, it was more likely to have been created from z_1 than z_2. The second term is the prior probability $Pr(z)$ over the latent variable. Combining these two factors and normalizing so the distribution sums to one gives us the posterior $Pr(z|\mathbf{x}^*, \boldsymbol{\phi})$.

This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.

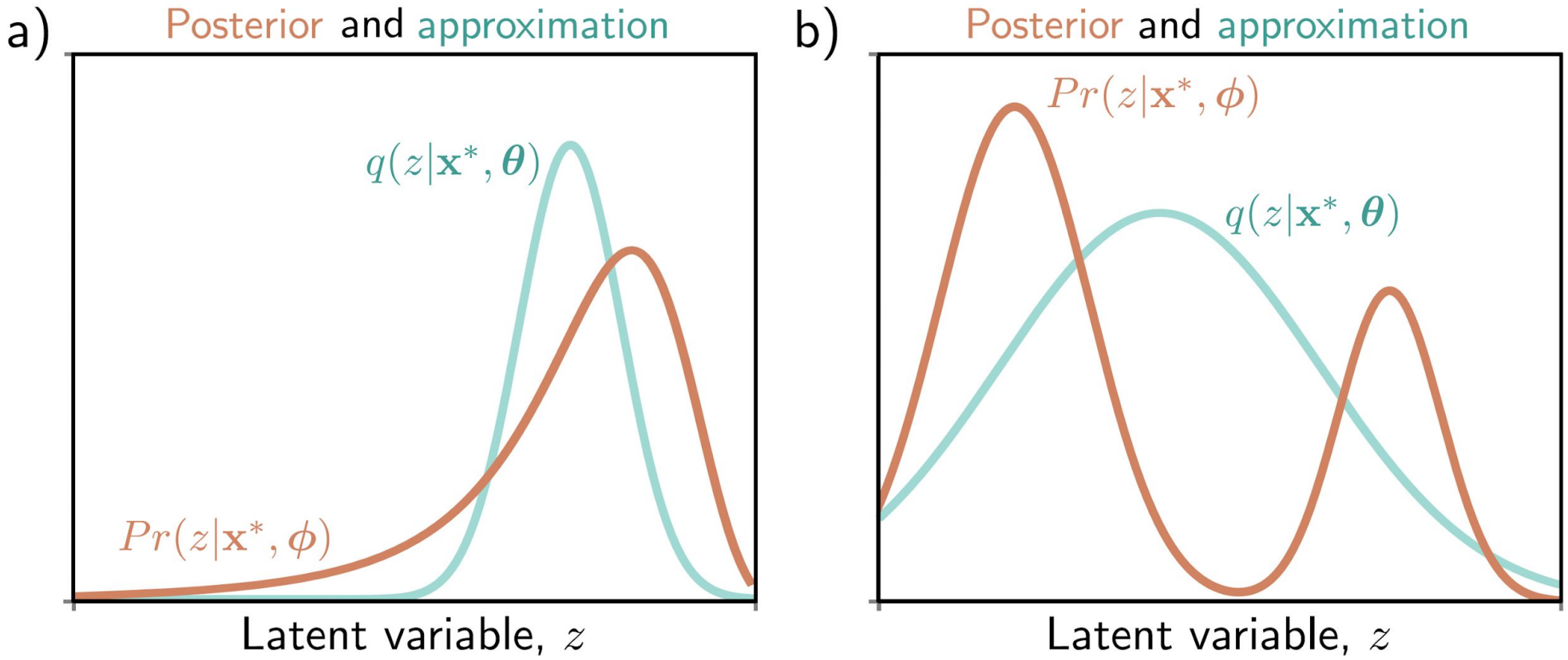


Figure 17.8 Variational approximation. The posterior $Pr(\mathbf{z}|\mathbf{x}^*, \boldsymbol{\phi})$ can't be computed in closed form. The variational approximation chooses a family of distributions $q(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})$ (here Gaussians) and tries to find the closest member of this family to the true posterior. a) Sometimes, the approximation (cyan curve) is good and lies close to the true posterior (orange curve). b) However, if the posterior is multi-modal (as in figure 17.7), then the Gaussian approximation will be poor.

This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.

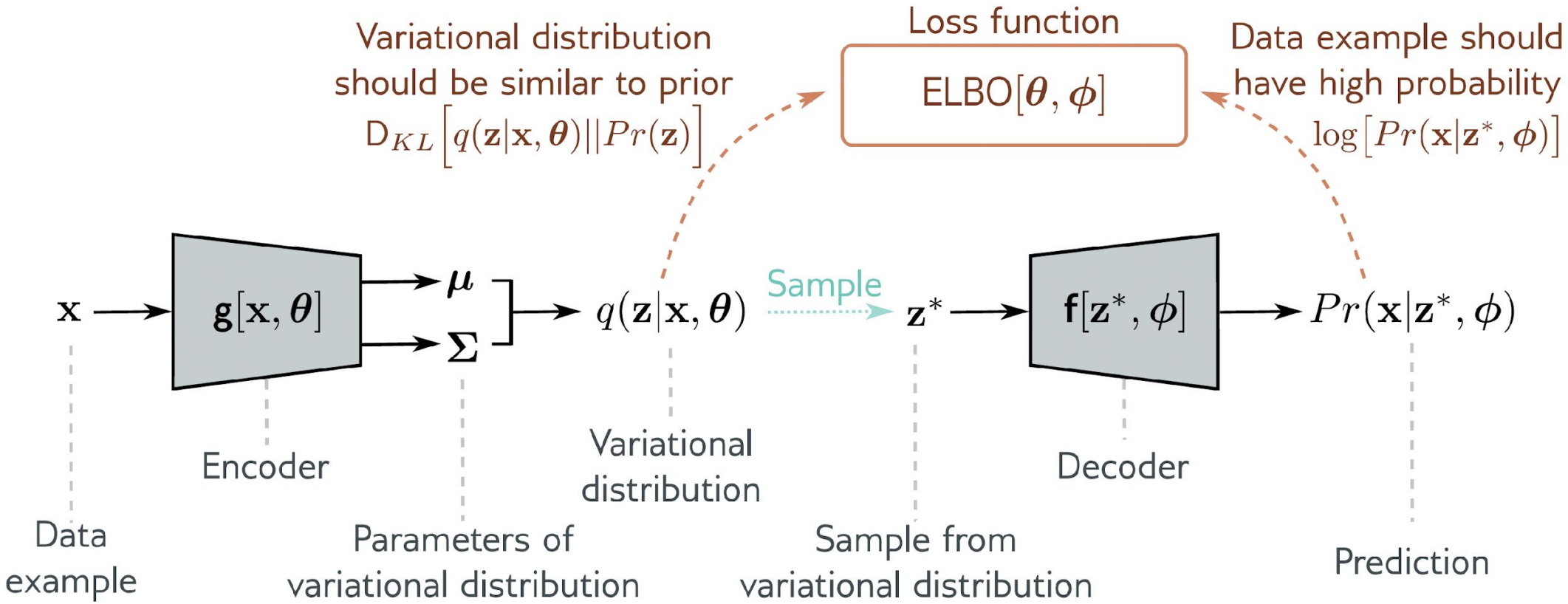
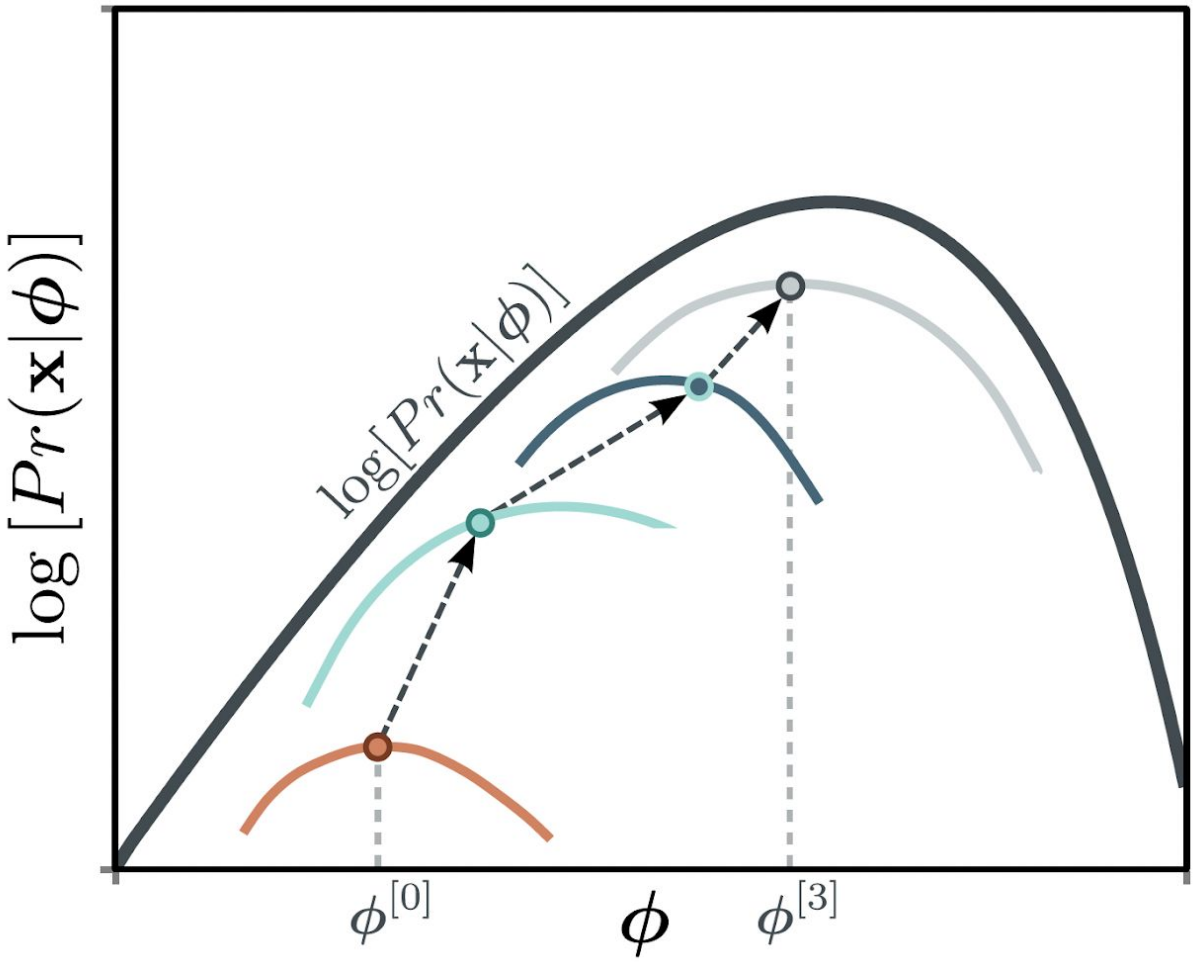


Figure 17.9 Variational autoencoder. The encoder $\mathbf{g}[\mathbf{x},\boldsymbol{\theta}]$ takes a training example $\mathbf{x}$ and predicts the parameters $\boldsymbol{\mu},\boldsymbol{\Sigma}$ of the variational distribution $q(\mathbf{z}|\mathbf{x},\boldsymbol{\theta})$. We sample from this distribution and then use the decoder $\mathbf{f}[\mathbf{z},\boldsymbol{\phi}]$ to predict the data $\mathbf{x}$. The loss function is the negative ELBO, which depends on how accurate this prediction is and how similar the variational distribution $q(\mathbf{z}|\mathbf{x},\boldsymbol{\theta})$ is to the prior $Pr(\mathbf{z})$ (equation 17.21).

This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.

Figure 17.10 The VAE updates both factors that determine the lower bound at each iteration. Both the parameters ϕ of the decoder and the parameters θ of the encoder are manipulated to increase this lower bound.



This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.

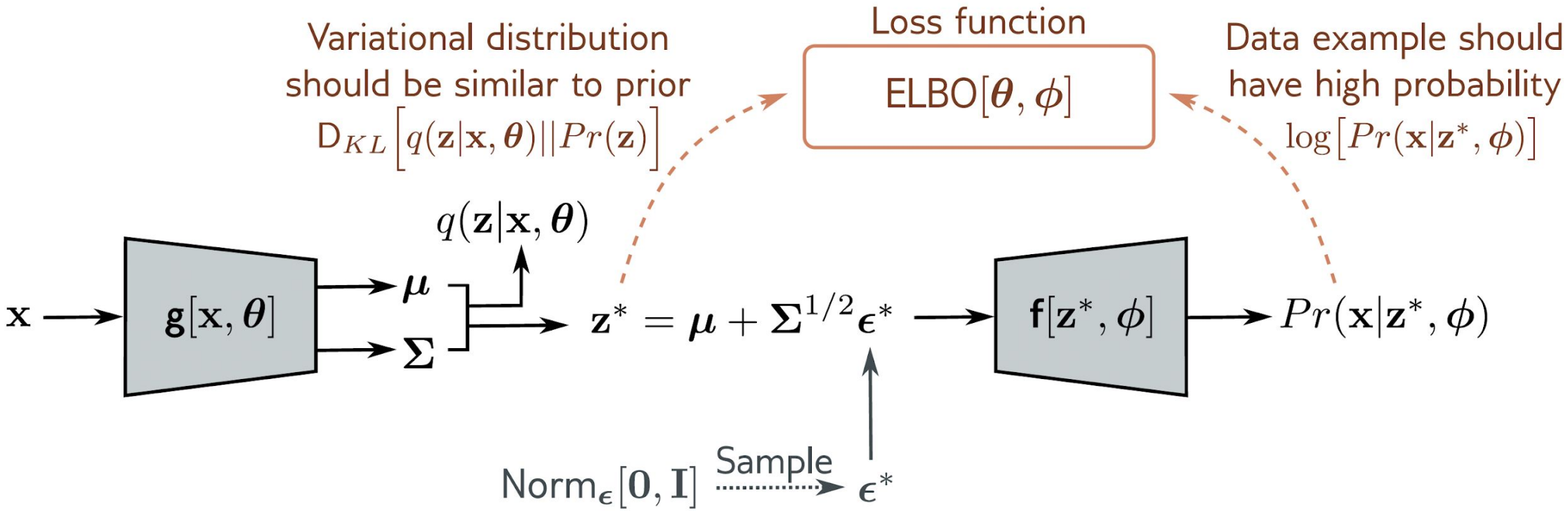


Figure 17.11 Reparameterization trick. With the original architecture (figure 17.9), we cannot easily backpropagate through the sampling step. The reparameterization trick removes the sampling step from the main pipeline; we draw from a standard normal and combine this with the predicted mean and covariance to get a sample from the variational distribution.

This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.

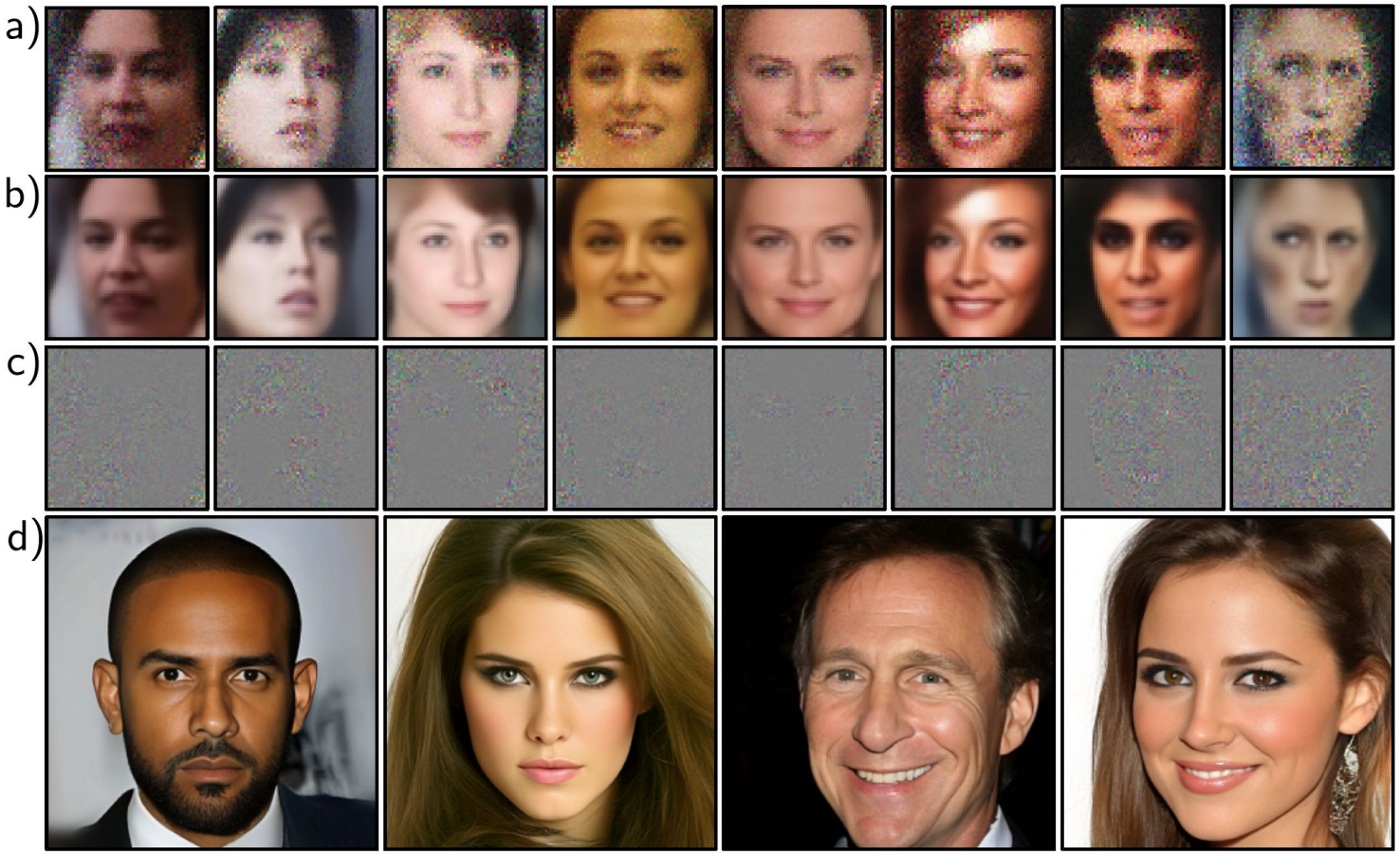


Figure 17.12 Sampling from a standard VAE trained on CELEBA. In each column, a latent variable $\mathbf{z}^*$ is drawn and passed through the model to predict the mean $\mathbf{f}[\mathbf{z}^*, \boldsymbol{\phi}]$ before adding independent Gaussian noise (see figure 17.3). a) A set of samples that are the sum of b) the predicted means and c) spherical Gaussian noise vectors. The images look too smooth before we add the noise and too noisy afterward. This is typical, and usually, the noise-free version is shown since the noise is considered to represent aspects of the image that are not modeled. Adapted from Dorta et al. (2018). d) It is now possible to generate high-quality images from VAEs using hierarchical priors, specialized architecture, and careful regularization. Adapted from Vahdat & Kautz (2020).

This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.

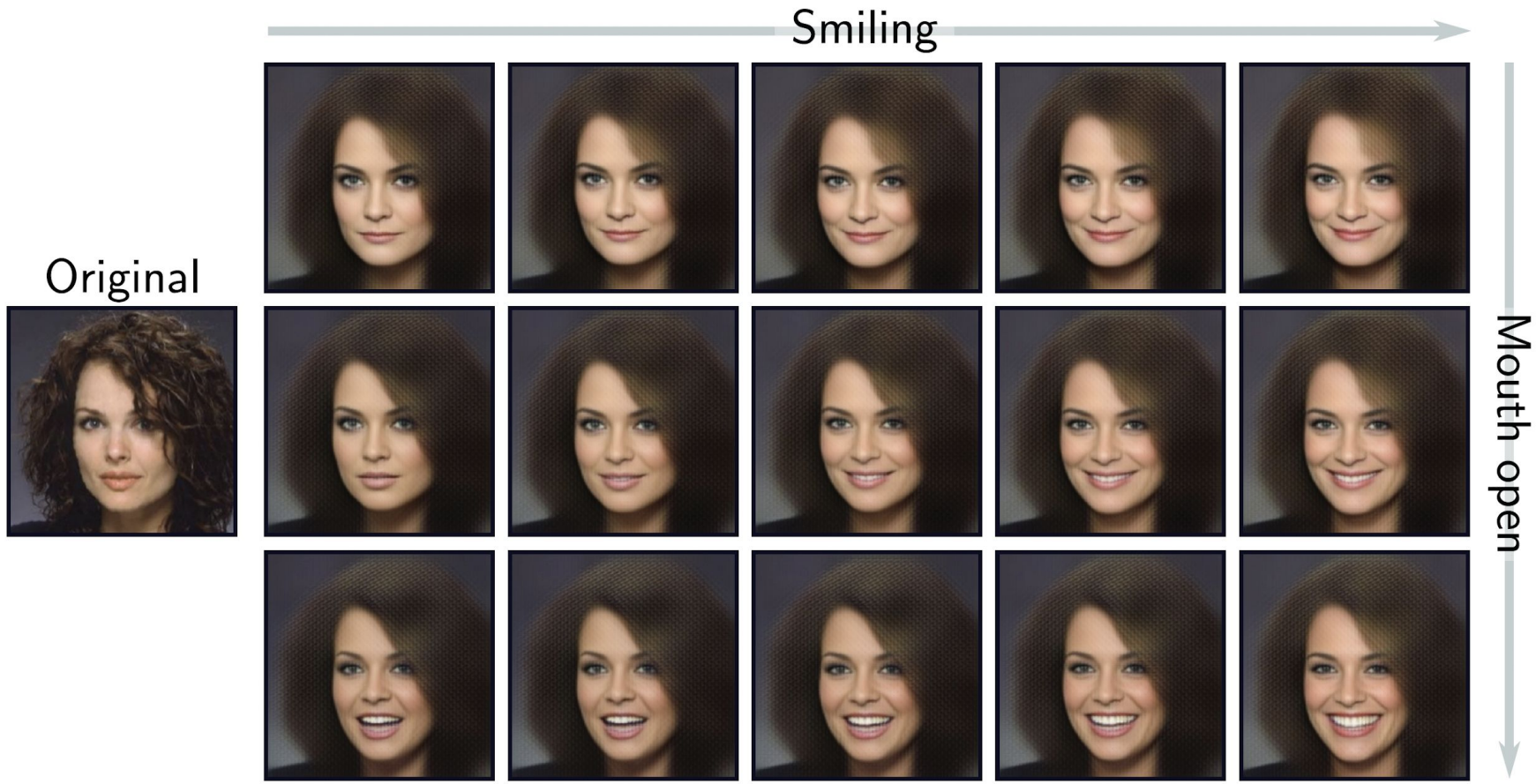


Figure 17.13 Resynthesis. The original image on the left is projected into the latent space using the encoder, and the mean of the predicted Gaussian is chosen to represent the image. The center-left image in the grid is the reconstruction of the input. The other images are reconstructions after manipulating the latent space in directions representing smiling/neutral (horizontal) and mouth open/closed (vertical). Adapted from White (2016).

This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.

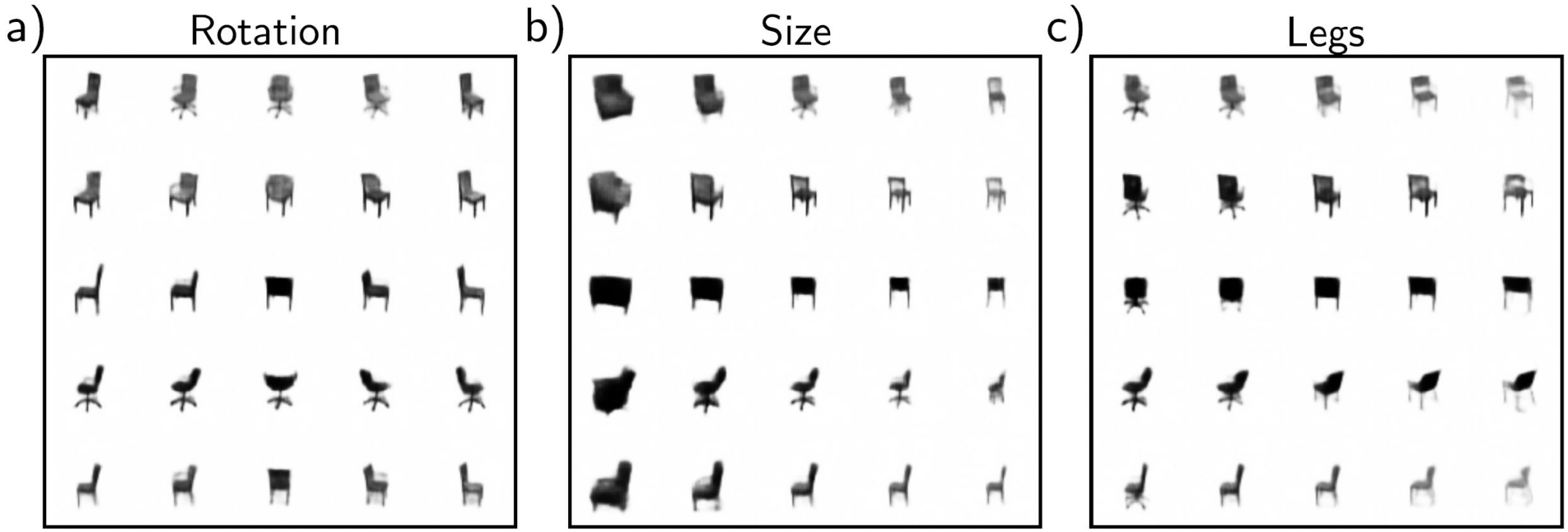
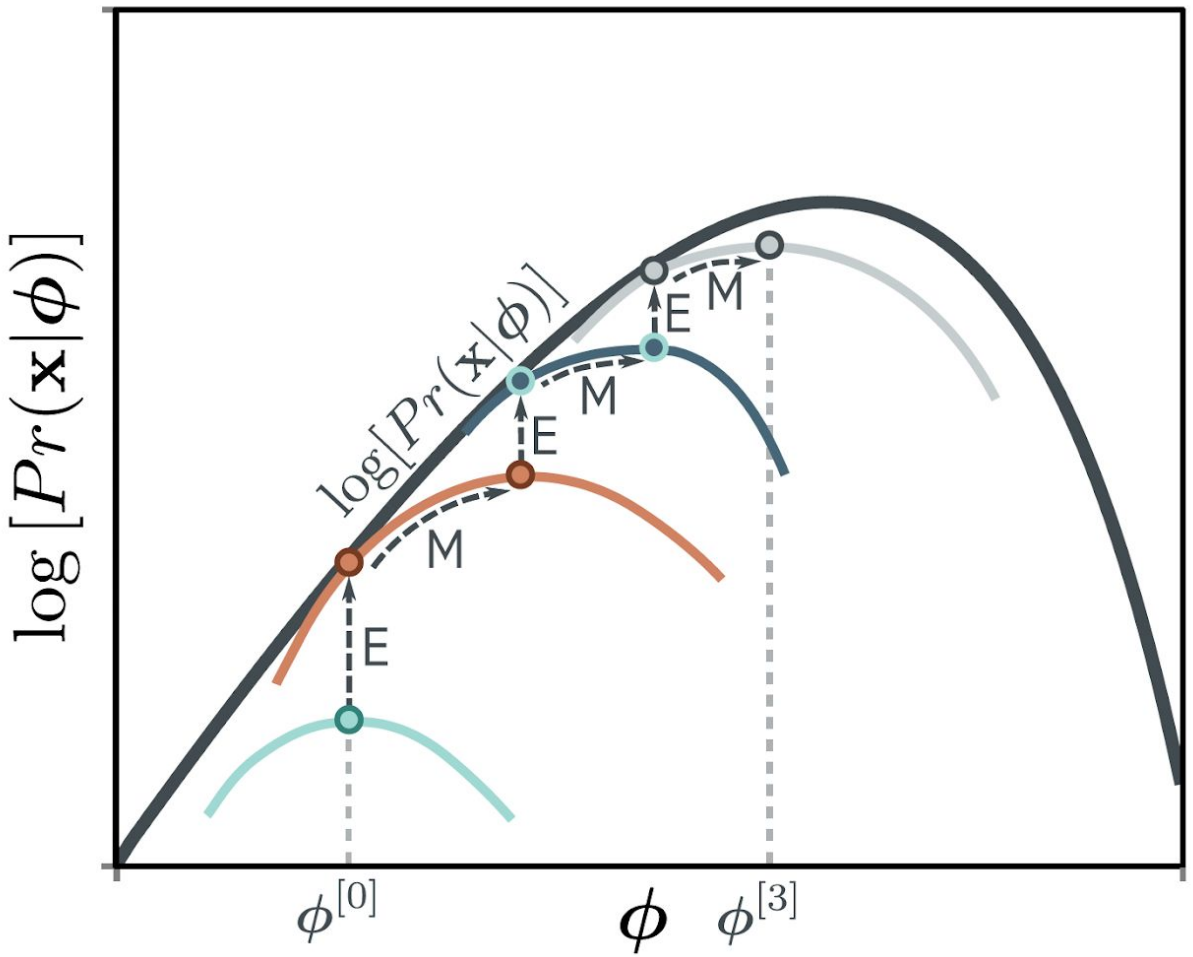


Figure 17.14 Disentanglement in the total correlation VAE. The VAE model is modified so that the loss function encourages the total correlation of the latent variables to be minimized and hence encourages disentanglement. When trained on a dataset of images of chairs, several of the latent dimensions have clear real-world interpretations, including a) rotation, b) overall size, and c) legs (swivel chair versus normal). In each case, the central column depicts samples from the model, and as we move left to right, we are subtracting or adding a coordinate vector in latent space. Adapted from Chen et al. (2018d).

This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.

Figure 17.15 Expectation maximization (EM) algorithm. The EM algorithm alternately adjusts the auxiliary parameters $\boldsymbol{\theta}$ (moves between colored curves) and model parameters $\boldsymbol{\phi}$ (moves along colored curves) until the a maximum is reached. These adjustments are known as the E-step and the M-step, respectively. Because the E-Step uses the posterior distribution $Pr(h|\mathbf{x}, \boldsymbol{\phi})$ for $q(h|\mathbf{x}, \boldsymbol{\theta})$, the bound is tight, and the colored curve touches the black likelihood curve after each E-Step.



This work is subject to a Creative Commons CC-BY-NC-ND license. (C) MIT Press.