component mentioned in some papers (e.g., [23, 48]), a different approach would be to train the prior with an adversarial loss. Further, [47] present various ideas how auto-encoders could benefit from adversarial learning.

4.4 Improving Variational Auto-Encoders

4.4.1 Priors

Insights from Rewriting the ELBO

One of the crucial components of VAEs is the marginal distribution over $\mathbf{z}$'s. Now, we will take a closer look at this distribution, also called the *prior*. Before we start thinking about improving it, we inspect the ELBO one more time. We can write ELBO as follows:

$$\mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}[\ln p(\mathbf{x})] \geq \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\ln p_\theta(\mathbf{x}|\mathbf{z}) + \ln p_\lambda(\mathbf{z}) - \ln q_\phi(\mathbf{z}|\mathbf{x})\right]\right], \tag{4.33}$$

where we explicitly highlight the summation over training data, namely, the expected value with respect to $\mathbf{x}$'s from the empirical distribution $p_{data}(\mathbf{x}) = \frac{1}{N}\sum_{n=1}^{N}\delta(\mathbf{x}-\mathbf{x}_n)$, and $\delta(\cdot)$ is the Dirac delta.

The ELBO consists of two parts, namely, the reconstruction error:

$$RE \triangleq \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\ln p_\theta(\mathbf{x}|\mathbf{z})\right]\right], \tag{4.34}$$

and the regularization term between the encoder and the prior:

$$\Omega \triangleq \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\ln p_\lambda(\mathbf{z}) - \ln q_\phi(\mathbf{z}|\mathbf{x})\right]\right]. \tag{4.35}$$

Further, let us play a little bit with the regularization term Ω:

$$\Omega = \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\ln p_\lambda(\mathbf{z}) - \ln q_\phi(\mathbf{z}|\mathbf{x})\right]\right] \tag{4.36}$$

$$= \int p_{data}(\mathbf{x}) \int q_\phi(\mathbf{z}|\mathbf{x})\left[\ln p_\lambda(\mathbf{z}) - \ln q_\phi(\mathbf{z}|\mathbf{x})\right] d\mathbf{z}\, d\mathbf{x} \tag{4.37}$$

$$= \iint p_{data}(\mathbf{x}) q_\phi(\mathbf{z}|\mathbf{x})\left[\ln p_\lambda(\mathbf{z}) - \ln q_\phi(\mathbf{z}|\mathbf{x})\right] d\mathbf{z}\, d\mathbf{x} \tag{4.38}$$

$$= \iint \frac{1}{N}\sum_n \delta(\mathbf{x}-\mathbf{x}_n) q_\phi(\mathbf{z}|\mathbf{x})\left[\ln p_\lambda(\mathbf{z}) - \ln q_\phi(\mathbf{z}|\mathbf{x})\right] d\mathbf{z}\, d\mathbf{x} \tag{4.39}$$

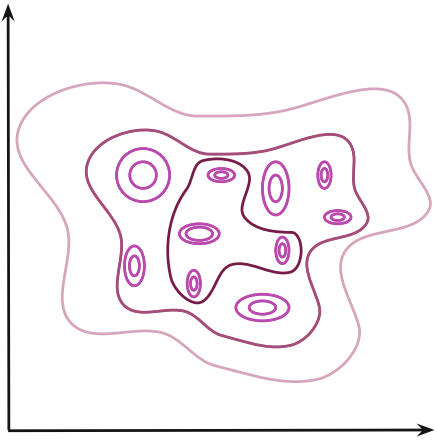

Fig. 4.5 An example of the aggregated posterior. Individual points are encoded as Gaussians in the 2D latent space (magenta), and the mixture of variational posteriors (the aggregated posterior) is presented by contours

$$= \int \frac{1}{N} \sum_{n=1}^{N} q_\phi(\mathbf{z}|\mathbf{x}_n) \left[\ln p_\lambda(\mathbf{z}) - \ln q_\phi(\mathbf{z}|\mathbf{x}_n) \right] \mathrm{d}\mathbf{z} \tag{4.40}$$

$$= \int \frac{1}{N} \sum_{n=1}^{N} q_\phi(\mathbf{z}|\mathbf{x}_n) \ln p_\lambda(\mathbf{z}) \mathrm{d}\mathbf{z} - \int \frac{1}{N} \sum_{n=1}^{N} q_\phi(\mathbf{z}|\mathbf{x}_n) \ln q_\phi(\mathbf{z}|\mathbf{x}_n) \mathrm{d}\mathbf{z} \tag{4.41}$$

$$= \int q_\phi(\mathbf{z}) \ln p_\lambda(\mathbf{z}) \mathrm{d}\mathbf{z} - \int \sum_{n=1}^{N} \frac{1}{N} q_\phi(\mathbf{z}|\mathbf{x}_n) \ln q_\phi(\mathbf{z}|\mathbf{x}_n) \mathrm{d}\mathbf{z} \tag{4.42}$$

$$= -\mathbb{CE} \left[q_\phi(\mathbf{z}) || p_\lambda(\mathbf{z}) \right] + \mathbb{H} \left[q_\phi(\mathbf{z}|\mathbf{x}) \right], \tag{4.43}$$

where we use the property of the Dirac delta: $\int \delta(a - a') f(a) \mathrm{d}a = f(a')$, and we use the notion of the **aggregated posterior** [47, 48] defined as follows:

$$q(\mathbf{z}) = \frac{1}{N} \sum_{n=1}^{N} q_\phi \left(\mathbf{z}|\mathbf{x}_n \right). \tag{4.44}$$

An example of the aggregated posterior is schematically depicted in Fig. 4.5.

Eventually, we obtain two terms:

(i) The first one, $\mathbb{CE} \left[q_\phi(\mathbf{z}) || p_\lambda(\mathbf{z}) \right]$, is the cross-entropy between the aggregated posterior and the prior.
(ii) The second term, $\mathbb{H} \left[q_\phi(\mathbf{z}|\mathbf{x}) \right]$, is the conditional entropy of $q_\phi(\mathbf{z}|\mathbf{x})$ with the empirical distribution $p_{data}(\mathbf{x})$.

I highly recommend doing this derivation step-by-step, as it helps a lot in understanding what is going on here. Interestingly, there is another possibility to

rewrite Ω using three terms, with the total correlation [49]. We will not use it here, so it is left as a "homework."

Anyway, one may ask why is it useful to rewrite the ELBO? The answer is rather straightforward: We can analyze it from a different perspective! In this section, we will focus on the **prior**, an important component in the generative part that is very often neglected. Many Bayesianists are stating that a prior should not be learned. But VAEs are not Bayesian models, please remember that! Besides, who says we cannot learn the prior? As we will see shortly, a non-learnable prior could be pretty annoying, especially for the generation process.

What Does ELBO Tell Us About the Prior?

Alright, we see that Ω consists of the cross-entropy and the entropy. Let us start with the entropy since it is easier to be analyzed. While optimizing, we want to maximize the ELBO and, hence, we maximize the entropy:

$$\mathbb{H}\left[q_\phi(\mathbf{z}|\mathbf{x})\right] = -\int \sum_{n=1}^{N} \frac{1}{N} q_\phi(\mathbf{z}|\mathbf{x}_n) \ln q_\phi(\mathbf{z}|\mathbf{x}_n) \mathrm{d}\mathbf{z}. \tag{4.45}$$

Before we make any conclusions, we should remember that we consider Gaussian encoders, $q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\mathbf{z}|\mu(\mathbf{x}), \sigma^2(\mathbf{x})\right)$. The entropy of a Gaussian distribution with a diagonal covariance matrix is equal to $\frac{1}{2}\sum_i \ln\left(2e\pi\sigma_i^2\right)$. Then, the question is when this quantity is maximized? The answer is: $\sigma_i^2 \to +\infty$. In other words, the entropy terms tries to stretch the encoders as much as possible by enlarging their variances! Of course, this does not happen in practice because we use the encoder together with the decoder in the RE term and the decoder tries to make the encoder as peaky as possible (i.e., ideally one $\mathbf{x}$ for one $\mathbf{z}$, like in the non-stochastic auto-encoder).

The second term in Ω is the cross-entropy:

$$\mathbb{CE}\left[q_\phi(\mathbf{z})||p_\lambda(\mathbf{z})\right] = -\int q_\phi(\mathbf{z}) \ln p_\lambda(\mathbf{z}) \mathrm{d}\mathbf{z}. \tag{4.46}$$

The cross-entropy term influences the VAE in a different manner. First, we can ask the question how to interpret the cross-entropy between $q_\phi(\mathbf{z})$ and $p_\lambda(\mathbf{z})$. In general, the cross-entropy tells us the average number of bits (or rather nats because we use the natural logarithm) needed to identify an event drawn from $q_\phi(\mathbf{z}$ if a coding scheme used for it is $p_\lambda(\mathbf{z})$. Notice that in Ω we have the negative cross-entropy. Since we maximize the ELBO, it means that we aim for minimizing $\mathbb{CE}\left[q_\phi(\mathbf{z})||p_\lambda(\mathbf{z})\right]$. This makes sense because we would like $q_\phi(\mathbf{z})$ to match $p_\lambda(\mathbf{z})$. And we have accidentally touched upon the most important issue here: What do we really want here? The cross-entropy forces the aggregated posterior to **match** the prior! That is the reason why we have this term here. If you think about it, it is a beautiful result that gives another connection between VAEs and the information theory.

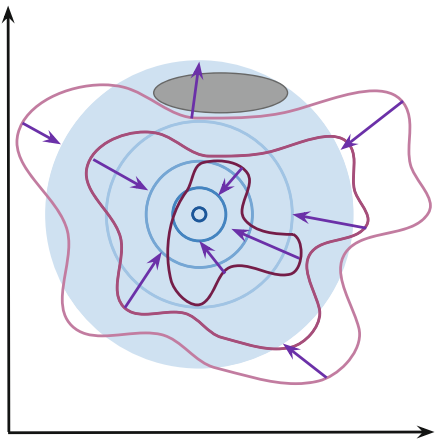

Fig. 4.6 An example of the effect of the cross-entropy optimization with a non-learnable prior. The aggregated posterior (purple contours) tries to match the non-learnable prior (in blue). The purple arrows indicate the change of the aggregated posterior. An example of a hole is presented as a dark gray ellipse

Alright, so we see what the cross-entropy does, but there are two possibilities here. First, the prior is fixed (**non-learnable**), e.g., the standard Gaussian prior. Then, optimizing the cross-entropy *pushes* the aggregated posterior to match the prior. It is schematically depicted in Fig. 4.6. The prior acts like an anchor and the *amoeba* of the aggregated posterior moves so that to fit the prior. In practice, this optimization process is troublesome because the decoder forces the encoder to be peaked and, in the end, it is almost impossible to match a fixed-shaped prior. As a result, we obtain **holes**, namely, regions in the latent space where the aggregated posterior assigns low probability while the prior assigns (relatively) high probability (see a dark gray ellipse in Fig. 4.6). This issue is especially apparent in generations because sampling from the prior, from the hole, may result in a sample that is of an extremely low quality. You can read more about it in [12].

On the other hand, if we consider a learnable **prior**, the situation looks different. The optimization allows to change the aggregated posterior **and** the prior. As the consequence, both distributions try to match each other (see Fig. 4.7). The problem of holes is then less apparent, especially if the prior is flexible enough. However, we can face other optimization issues when the prior and the aggregated posteriors chase each other. In practice, the learnable prior seems to be a better option, but it is still an open question whether training all components at once is the best approach. Moreover, the learnable prior does not impose any specific constraint on the latent representation, e.g., sparsity. This could be another problem that would result in undesirable problems (e.g., non-smooth encoders).

Eventually, we can ask the fundamental question: What is the *best* prior then?! The answer is already known and is hidden in the cross-entropy term: It is the aggregated posterior. If we take $p_\lambda(\mathbf{z}) = \sum_{n=1}^{N} \frac{1}{N} q_\phi(\mathbf{z}|\mathbf{x}_n)$, then, theoretically, the cross-entropy equals the entropy of $q_\phi(\mathbf{z})$ and the regularization term Ω is smallest. However, in practice, this is infeasible because:

- We cannot sum over tens of thousands of points and backpropagate through them.

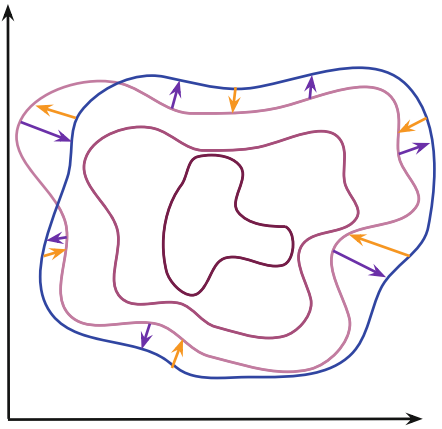

Fig. 4.7 An example of the effect of the cross-entropy optimization with a learnable prior. The aggregated posterior (purple contours) tries to match the learnable prior (blue contours). Notice that the aggregated posterior is modified to fit the prior (purple arrows), but also the prior is updated to cover the aggregated posterior (orange arrows)

- This result is fine from the theoretical point of view; however, the optimization process is stochastic and could cause additional errors.
- As mentioned earlier, choosing the aggregated posterior as the prior does not constrain the latent representation in any obvious manner and, thus, the encoder could behave unpredictably.
- The aggregated posterior may work well if the get $N \rightarrow +\infty$ points, because then we can get any distribution; however, this is not the case in practice and it contradicts also the first bullet.

As a result, we can keep this theoretical solution in mind and formulate **approximations** to it that are computationally tractable. In the next sections, we will discuss a few of them.

4.4.1.1 Standard Gaussian

The vanilla implementation of the VAE assumes a standard Gaussian marginal (prior) over $\mathbf{z}$, $p_\lambda(\mathbf{z}) = \mathcal{N}(\mathbf{z}|0, \mathrm{I})$. This prior is simple, non-trainable (i.e., no extra parameters to learn), and easy to implement. In other words, it is amazing! However, as discussed above, the standard normal distribution could lead to very poor hidden representations with holes resulting from the mismatch between the aggregated posterior and the prior.

To strengthen our discussion, we trained a small VAE with the standard Gaussian prior and a two-dimensional latent space. In Fig. 4.8, we present samples from the encoder for the test data (black dots) and the contour plot for the standard prior. We can spot holes where the aggregated posterior does not assign any points (i.e., the mismatch between the prior and the aggregated posterior).