

Note that our encoder distributions $q(\mathbf{x}_t|\mathbf{x}_{t-1})$ are no longer parameterized by ϕ , as they are completely modeled as Gaussians with defined mean and variance parameters at each timestep. Therefore, in a VDM, we are only interested in learning conditionals $p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t)$, so that we can simulate new data. After optimizing the VDM, the sampling procedure is as simple as sampling Gaussian noise from $p(\mathbf{x}_T)$ and iteratively running the denoising transitions $p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t)$ for T steps to generate a novel \mathbf{x}_0 .

Like any HVAE, the VDM can be optimized by maximizing the ELBO, which can be derived as:

$$\log p(\mathbf{x}) = \log \int p(\mathbf{x}_{0:T}) d\mathbf{x}_{1:T} \quad \text{marginalize} \quad (34)$$

$$= \log \int \frac{p(\mathbf{x}_{0:T}) q(\mathbf{x}_{1:T}|\mathbf{x}_0)}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} d\mathbf{x}_{1:T} \quad (35)$$

$$= \log \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\frac{p(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \right] \quad (36)$$

$$\geq \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \right] \quad (37)$$

$$= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T) \prod_{t=1}^T p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t)}{\prod_{t=1}^T q(\mathbf{x}_t|\mathbf{x}_{t-1})} \right] \quad (38)$$

$$= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T) p_{\theta}(\mathbf{x}_0|\mathbf{x}_1) \prod_{t=2}^T p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_T|\mathbf{x}_{T-1}) \prod_{t=1}^{T-1} q(\mathbf{x}_t|\mathbf{x}_{t-1})} \right] \quad (39)$$

$$= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T) p_{\theta}(\mathbf{x}_0|\mathbf{x}_1) \prod_{t=1}^{T-1} p_{\theta}(\mathbf{x}_t|\mathbf{x}_{t+1})}{q(\mathbf{x}_T|\mathbf{x}_{T-1}) \prod_{t=1}^{T-1} q(\mathbf{x}_t|\mathbf{x}_{t-1})} \right] \quad (40)$$

$$= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T) p_{\theta}(\mathbf{x}_0|\mathbf{x}_1)}{q(\mathbf{x}_T|\mathbf{x}_{T-1})} \right] + \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \prod_{t=1}^{T-1} \frac{p_{\theta}(\mathbf{x}_t|\mathbf{x}_{t+1})}{q(\mathbf{x}_t|\mathbf{x}_{t-1})} \right] \quad (41)$$

$$= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} [\log p_{\theta}(\mathbf{x}_0|\mathbf{x}_1)] + \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T)}{q(\mathbf{x}_T|\mathbf{x}_{T-1})} \right] + \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\sum_{t=1}^{T-1} \log \frac{p_{\theta}(\mathbf{x}_t|\mathbf{x}_{t+1})}{q(\mathbf{x}_t|\mathbf{x}_{t-1})} \right] \quad (42)$$

$$= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} [\log p_{\theta}(\mathbf{x}_0|\mathbf{x}_1)] + \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T)}{q(\mathbf{x}_T|\mathbf{x}_{T-1})} \right] + \sum_{t=1}^{T-1} \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p_{\theta}(\mathbf{x}_t|\mathbf{x}_{t+1})}{q(\mathbf{x}_t|\mathbf{x}_{t-1})} \right] \quad (43)$$

$$= \mathbb{E}_{q(\mathbf{x}_1|\mathbf{x}_0)} [\log p_{\theta}(\mathbf{x}_0|\mathbf{x}_1)] + \mathbb{E}_{q(\mathbf{x}_{T-1}, \mathbf{x}_T|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T)}{q(\mathbf{x}_T|\mathbf{x}_{T-1})} \right] + \sum_{t=1}^{T-1} \mathbb{E}_{q(\mathbf{x}_{t-1}, \mathbf{x}_t, \mathbf{x}_{t+1}|\mathbf{x}_0)} \left[\log \frac{p_{\theta}(\mathbf{x}_t|\mathbf{x}_{t+1})}{q(\mathbf{x}_t|\mathbf{x}_{t-1})} \right] \quad (44)$$

$$\begin{aligned} &= \underbrace{\mathbb{E}_{q(\mathbf{x}_1|\mathbf{x}_0)} [\log p_{\theta}(\mathbf{x}_0|\mathbf{x}_1)]}_{\text{reconstruction term}} - \underbrace{\mathbb{E}_{q(\mathbf{x}_{T-1}|\mathbf{x}_0)} [D_{\text{KL}}(q(\mathbf{x}_T|\mathbf{x}_{T-1}) \| p(\mathbf{x}_T))]}_{\text{prior matching term}} \int \left[\log \frac{p(\mathbf{x}_T)}{q(\mathbf{x}_T|\mathbf{x}_{T-1})} \right] q_{\theta}(\mathbf{x}_T|\mathbf{x}_{T-1}) d\mathbf{x}_T \\ &\quad = \int \mathbb{E}_{q(\mathbf{x}_T|\mathbf{x}_{T-1})} \left[\log \frac{p(\mathbf{x}_T)}{q(\mathbf{x}_T|\mathbf{x}_{T-1})} \right] \cdot q_{\theta}(\mathbf{x}_T|\mathbf{x}_{T-1}) d\mathbf{x}_{T-1} \\ &\quad - \sum_{t=1}^{T-1} \underbrace{\mathbb{E}_{q(\mathbf{x}_{t-1}, \mathbf{x}_{t+1}|\mathbf{x}_0)} [D_{\text{KL}}(q(\mathbf{x}_t|\mathbf{x}_{t-1}) \| p_{\theta}(\mathbf{x}_t|\mathbf{x}_{t+1}))]}_{\text{consistency term}} = - \mathbb{E}_{q(\mathbf{x}_{T-1}|\mathbf{x}_0)} [D_{\text{KL}}(q_{\theta}(\mathbf{x}_T|\mathbf{x}_{T-1}) \| p(\mathbf{x}_T))] \end{aligned} \quad (45)$$

two random variables

The derived form of the ELBO can be interpreted in terms of its individual components:

- $\mathbb{E}_{q(\mathbf{x}_1|\mathbf{x}_0)} [\log p_{\theta}(\mathbf{x}_0|\mathbf{x}_1)]$ can be interpreted as a *reconstruction term*, predicting the log probability of the original data sample given the first-step latent. This term also appears in a vanilla VAE, and can be trained similarly.
- $\mathbb{E}_{q(\mathbf{x}_{T-1}|\mathbf{x}_0)} [D_{\text{KL}}(q(\mathbf{x}_T|\mathbf{x}_{T-1}) \| p(\mathbf{x}_T))]$ is a *prior matching term*; it is minimized when the final latent distribution matches the Gaussian prior. This term requires no optimization, as it has no trainable parameters; furthermore, as we have assumed a large enough T such that the final distribution is Gaussian, this term effectively becomes zero.
- $\mathbb{E}_{q(\mathbf{x}_{t-1}, \mathbf{x}_{t+1}|\mathbf{x}_0)} [D_{\text{KL}}(q(\mathbf{x}_t|\mathbf{x}_{t-1}) \| p_{\theta}(\mathbf{x}_t|\mathbf{x}_{t+1}))]$ is a *consistency term*; it endeavors to make the distribution at \mathbf{x}_t consistent, from both forward and backward processes. That is, a denoising step from a noisier image should match the corresponding noising step from a cleaner image, for every intermediate timestep; this is reflected mathematically by the KL Divergence. This term is minimized when we train $p_{\theta}(\mathbf{x}_t|\mathbf{x}_{t+1})$ to match the Gaussian distribution $q(\mathbf{x}_t|\mathbf{x}_{t-1})$, which is defined in Equation 31.

$$p_{\theta}(\mathbf{x}_t|\mathbf{x}_{t+1}) \xrightarrow{\text{trained}} q_{\theta}(\mathbf{x}_t|\mathbf{x}_{t-1}) \quad (\text{Eq. 31})$$

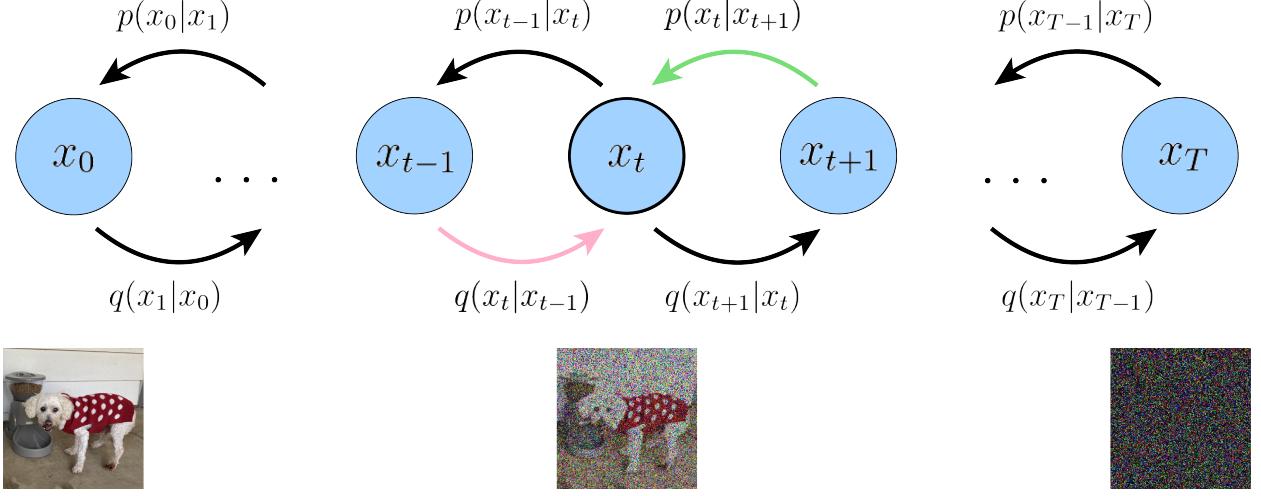


Figure 4: Under our first derivation, a VDM can be optimized by ensuring that for every intermediate \mathbf{x}_t , the posterior from the latent above it $p_\theta(\mathbf{x}_t|\mathbf{x}_{t+1})$ matches the Gaussian corruption of the latent before it $q(\mathbf{x}_t|\mathbf{x}_{t-1})$. In this figure, for each intermediate \mathbf{x}_t , we minimize the difference between the distributions represented by the pink and green arrows.

Visually, this interpretation of the ELBO is depicted in Figure 4. The cost of optimizing a VDM is primarily dominated by the third term, since we must optimize over all timesteps t .

Under this derivation, all terms of the ELBO are computed as expectations, and can therefore be approximated using Monte Carlo estimates. However, actually optimizing the ELBO using the terms we just derived might be suboptimal; because the consistency term is computed as an expectation over two random variables $\{\mathbf{x}_{t-1}, \mathbf{x}_{t+1}\}$ for every timestep, the variance of its Monte Carlo estimate could potentially be higher than a term that is estimated using only one random variable per timestep. As it is computed by summing up $T - 1$ consistency terms, the final estimated value of the ELBO may have high variance for large T values.

Let us instead try to derive a form for our ELBO where each term is computed as an expectation over only one random variable at a time. The key insight is that we can rewrite encoder transitions as $q(\mathbf{x}_t|\mathbf{x}_{t-1}) = q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0)$, where the extra conditioning term is superfluous due to the Markov property. Then, according to Bayes rule, we can rewrite each transition as:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0) = \frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)q(\mathbf{x}_t|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_0)} \quad (46)$$

Armed with this new equation, we can retry the derivation resuming from the ELBO in Equation 37:

$$\log p(\mathbf{x}) \geq \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \right] \quad (47)$$

$$= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T) \prod_{t=1}^T p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{\prod_{t=1}^T q(\mathbf{x}_t|\mathbf{x}_{t-1})} \right] \quad (48)$$

$$= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T)p_\theta(\mathbf{x}_0|\mathbf{x}_1)\prod_{t=2}^T p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_1|\mathbf{x}_0)\prod_{t=2}^T q(\mathbf{x}_t|\mathbf{x}_{t-1})} \right] \text{differ from Eq.39} \quad (49)$$

$$= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T)p_\theta(\mathbf{x}_0|\mathbf{x}_1)\prod_{t=2}^T p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_1|\mathbf{x}_0)\prod_{t=2}^T q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0)} \right] \quad (50)$$

$$= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p_\theta(\mathbf{x}_T)p_\theta(\mathbf{x}_0|\mathbf{x}_1)}{q(\mathbf{x}_1|\mathbf{x}_0)} + \log \prod_{t=2}^T \frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0)} \right] \quad (51)$$

$$= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T)p_\theta(\mathbf{x}_0|\mathbf{x}_1)}{q(\mathbf{x}_1|\mathbf{x}_0)} + \log \prod_{t=2}^T \frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{\frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)q(\mathbf{x}_t|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_0)}} \right] \quad (52)$$

$$= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T)p_{\theta}(\mathbf{x}_0|\mathbf{x}_1)}{q(\mathbf{x}_1|\mathbf{x}_0)} + \log \prod_{t=2}^T \frac{p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t)}{\frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)q(\mathbf{x}_t|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_0)}} \right] \quad (53)$$

$$= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T)p_{\theta}(\mathbf{x}_0|\mathbf{x}_1)}{q(\mathbf{x}_1|\mathbf{x}_0)} + \log \frac{q(\mathbf{x}_1|\mathbf{x}_0)}{q(\mathbf{x}_T|\mathbf{x}_0)} + \log \prod_{t=2}^T \frac{p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)} \right] \quad (54)$$

$$= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T)p_{\theta}(\mathbf{x}_0|\mathbf{x}_1)}{q(\mathbf{x}_T|\mathbf{x}_0)} + \sum_{t=2}^T \log \frac{p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)} \right] \quad (55)$$

$$= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} [\log p_{\theta}(\mathbf{x}_0|\mathbf{x}_1)] + \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T)}{q(\mathbf{x}_T|\mathbf{x}_0)} \right] + \sum_{t=2}^T \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)} \right] \quad (56)$$

$$= \mathbb{E}_{q(\mathbf{x}_1|\mathbf{x}_0)} [\log p_{\theta}(\mathbf{x}_0|\mathbf{x}_1)] + \mathbb{E}_{q(\mathbf{x}_T|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T)}{q(\mathbf{x}_T|\mathbf{x}_0)} \right] + \sum_{t=2}^T \mathbb{E}_{q(\mathbf{x}_t, \mathbf{x}_{t-1}|\mathbf{x}_0)} \left[\log \frac{p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)} \right] \quad (57)$$

one random variable

$$= \underbrace{\mathbb{E}_{q(\mathbf{x}_1|\mathbf{x}_0)} [\log p_{\theta}(\mathbf{x}_0|\mathbf{x}_1)]}_{\text{reconstruction term}} - \underbrace{D_{\text{KL}}(q(\mathbf{x}_T|\mathbf{x}_0) \parallel p(\mathbf{x}_T))}_{\text{prior matching term}} - \sum_{t=2}^T \underbrace{\mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)} [D_{\text{KL}}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \parallel p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t))]}_{\text{denoising matching term}} \quad (58)$$

We have therefore successfully derived an interpretation for the ELBO that can be estimated with lower variance, as each term is computed as an expectation of at most one random variable at a time. This formulation also has an elegant interpretation, which is revealed when inspecting each individual term:

1. $\mathbb{E}_{q(\mathbf{x}_1|\mathbf{x}_0)} [\log p_{\theta}(\mathbf{x}_0|\mathbf{x}_1)]$ can be interpreted as a **reconstruction term**; like its analogue in the ELBO of a vanilla VAE, this term can be approximated and optimized using a **Monte Carlo estimate**.
2. $D_{\text{KL}}(q(\mathbf{x}_T|\mathbf{x}_0) \parallel p(\mathbf{x}_T))$ represents how close the distribution of the final noisified input is to the standard Gaussian prior. It has no trainable parameters, and is also equal to zero under our assumptions.
3. $\mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)} [D_{\text{KL}}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \parallel p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t))]$ is a **denoising matching term**. We learn desired denoising transition step $p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t)$ as an approximation to tractable, ground-truth denoising transition step $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$. The $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ transition step can act as a ground-truth signal, since it defines how to denoise a noisy image \mathbf{x}_t with access to what the final, completely denoised image \mathbf{x}_0 should be. This term is therefore minimized when the two denoising steps match as closely as possible, as measured by their KL Divergence. $P_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t) \rightarrow q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ *ground truth*

As a side note, one observes that in the process of both ELBO derivations (Equation 45 and Equation 58), only the Markov assumption is used; as a result these formulae will hold true for any arbitrary Markovian HVAE. Furthermore, when we set $T = 1$, both of the ELBO interpretations for a VDM exactly recreate the ELBO equation of a vanilla VAE, as written in Equation 19.

In this derivation of the ELBO, the bulk of the optimization cost once again lies in the summation term, which dominates the reconstruction term. Whereas each KL Divergence term $D_{\text{KL}}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \parallel p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t))$ is difficult to minimize for arbitrary posteriors in arbitrarily complex Markovian HVAEs due to the added complexity of simultaneously learning the encoder, in a VDM we can leverage the Gaussian transition assumption to make optimization tractable. By Bayes rule, we have:

if not including \mathbf{x}_0 *Gaussian encoder*

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0)q(\mathbf{x}_{t-1}|\mathbf{x}_0)}{q(\mathbf{x}_t|\mathbf{x}_0)}$$

$q(\mathbf{x}_t)$ and $q(\mathbf{x}_{t-1})$ cannot be determined

As we already know that $q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0) = q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{\alpha_t}\mathbf{x}_{t-1}, (1-\alpha_t)\mathbf{I})$ from our assumption regarding encoder transitions (Equation 31), what remains is deriving for the forms of $q(\mathbf{x}_t|\mathbf{x}_0)$ and $q(\mathbf{x}_{t-1}|\mathbf{x}_0)$. Fortunately, these are also made tractable by utilizing the fact that the encoder transitions of a VDM are **linear Gaussian models**. Recall that under the reparameterization trick, samples $\mathbf{x}_t \sim q(\mathbf{x}_t|\mathbf{x}_{t-1})$ can be rewritten as:

$$\mathbf{x}_t = \sqrt{\alpha_t}\mathbf{x}_{t-1} + \sqrt{1-\alpha_t}\boldsymbol{\epsilon} \quad \text{with } \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{\epsilon}; \mathbf{0}, \mathbf{I}) \quad (59)$$

and that similarly, samples $\mathbf{x}_{t-1} \sim q(\mathbf{x}_{t-1}|\mathbf{x}_{t-2})$ can be rewritten as:

$$\mathbf{x}_{t-1} = \sqrt{\alpha_{t-1}}\mathbf{x}_{t-2} + \sqrt{1-\alpha_{t-1}}\boldsymbol{\epsilon} \quad \text{with } \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{\epsilon}; \mathbf{0}, \mathbf{I}) \quad (60)$$

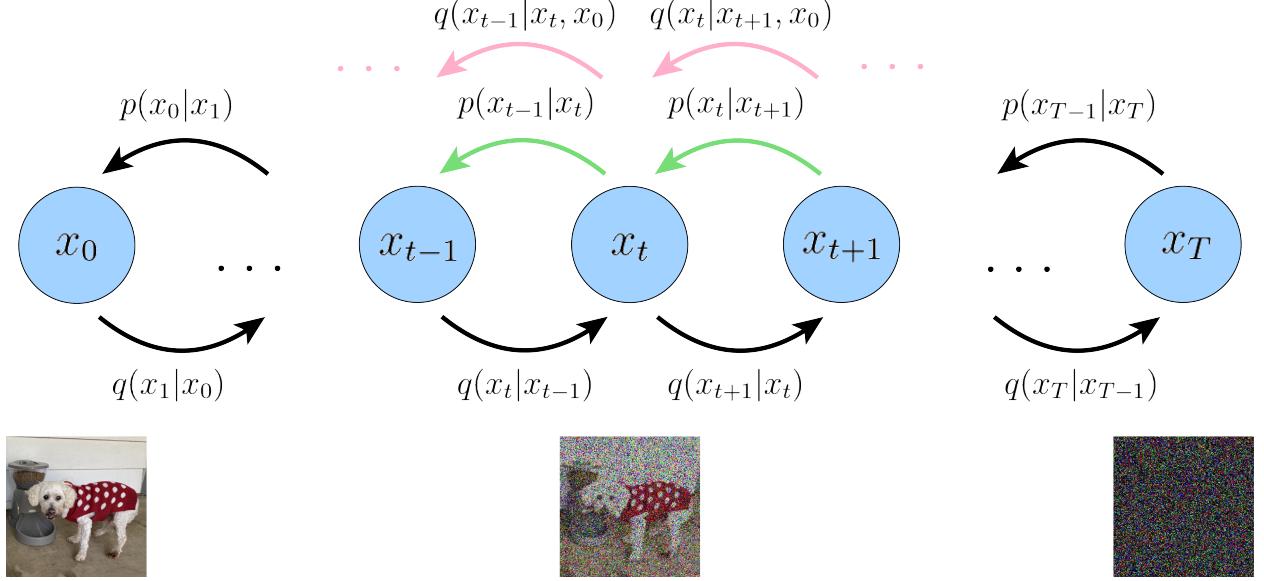


Figure 5: Depicted is an alternate, lower-variance method to optimize a VDM; we compute the form of ground-truth denoising step $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ using Bayes rule, and minimize its KL Divergence with our approximate denoising step $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$. This is once again denoted visually by matching the distributions represented by the green arrows with those of the pink arrows. Artistic liberty is at play here; in the full picture, each pink arrow must also stem from \mathbf{x}_0 , as it is also a conditioning term.

Determination of $q_u(\mathbf{x}_t|\mathbf{x}_0)$ and $q_u(\mathbf{x}_{t-1}|\mathbf{x}_0)$

Then, the form of $q(\mathbf{x}_t|\mathbf{x}_0)$ can be recursively derived through repeated applications of the reparameterization trick. Suppose that we have access to $2T$ random noise variables $\{\boldsymbol{\epsilon}_t^*, \boldsymbol{\epsilon}_t\}_{t=0}^T \stackrel{\text{iid}}{\sim} \mathcal{N}(\boldsymbol{\epsilon}; \mathbf{0}, \mathbf{I})$. Then, for an arbitrary sample $\mathbf{x}_t \sim q(\mathbf{x}_t|\mathbf{x}_0)$, we can rewrite it as:

$$\mathbf{x}_t = \sqrt{\alpha_t} \mathbf{x}_{t-1} + \sqrt{1 - \alpha_t} \boldsymbol{\epsilon}_t^* \quad \mathcal{N}(\mathbf{0}, (\alpha_t - \alpha_{t-1}) \mathbf{I}) \quad (61)$$

$$= \sqrt{\alpha_t} \left(\sqrt{\alpha_{t-1}} \mathbf{x}_{t-2} + \sqrt{1 - \alpha_{t-1}} \boldsymbol{\epsilon}_{t-2}^* \right) + \sqrt{1 - \alpha_t} \boldsymbol{\epsilon}_t^* \quad (62)$$

$$= \sqrt{\alpha_t \alpha_{t-1}} \mathbf{x}_{t-2} + \sqrt{\alpha_t - \alpha_t \alpha_{t-1}} \boldsymbol{\epsilon}_{t-2}^* + \sqrt{1 - \alpha_t} \boldsymbol{\epsilon}_t^* \sim \mathcal{N}(\mathbf{0}, (1 - \alpha_t) \mathbf{I}) \quad (63)$$

$$= \sqrt{\alpha_t \alpha_{t-1}} \mathbf{x}_{t-2} + \sqrt{\sqrt{\alpha_t - \alpha_t \alpha_{t-1}}^2 + \sqrt{1 - \alpha_t}^2} \boldsymbol{\epsilon}_{t-2} \quad (64)$$

$$= \sqrt{\alpha_t \alpha_{t-1}} \mathbf{x}_{t-2} + \sqrt{\alpha_t - \alpha_t \alpha_{t-1} + 1 - \alpha_t} \boldsymbol{\epsilon}_{t-2} \quad (65)$$

$$= \sqrt{\alpha_t \alpha_{t-1}} \mathbf{x}_{t-2} + \sqrt{1 - \alpha_t \alpha_{t-1}} \boldsymbol{\epsilon}_{t-2} \sim \mathcal{N}(\mathbf{0}, (1 - \alpha_t \alpha_{t-1}) \mathbf{I}) \quad (66)$$

$$= \dots \quad (67)$$

$$= \sqrt{\prod_{i=1}^t \alpha_i} \mathbf{x}_0 + \sqrt{1 - \prod_{i=1}^t \alpha_i} \boldsymbol{\epsilon}_0 \quad (68)$$

$$= \sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \boldsymbol{\epsilon}_0 \quad (69)$$

$$\sim \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t} \mathbf{x}_0, (1 - \bar{\alpha}_t) \mathbf{I}) \quad (70)$$

where in Equation 64 we have utilized the fact that the sum of two independent Gaussian random variables remains a Gaussian with mean being the sum of the two means, and variance being the sum of the two variances. Interpreting $\sqrt{1 - \alpha_t} \boldsymbol{\epsilon}_t^*$ as a sample from Gaussian $\mathcal{N}(\mathbf{0}, (1 - \alpha_t) \mathbf{I})$, and $\sqrt{\alpha_t - \alpha_t \alpha_{t-1}} \boldsymbol{\epsilon}_{t-2}$ as a sample from Gaussian $\mathcal{N}(\mathbf{0}, (\alpha_t - \alpha_t \alpha_{t-1}) \mathbf{I})$, we can then treat their sum as a random variable sampled from Gaussian $\mathcal{N}(\mathbf{0}, (1 - \alpha_t + \alpha_t - \alpha_t \alpha_{t-1}) \mathbf{I}) = \mathcal{N}(\mathbf{0}, (1 - \alpha_t \alpha_{t-1}) \mathbf{I})$. A sample from this distribution can then be represented using the reparameterization trick as $\sqrt{1 - \alpha_t \alpha_{t-1}} \boldsymbol{\epsilon}_{t-2}$, as in Equation 66.

We have therefore derived the Gaussian form of $q(\mathbf{x}_t | \mathbf{x}_0)$. This derivation can be modified to also yield the Gaussian parameterization describing $q(\mathbf{x}_{t-1} | \mathbf{x}_0)$. Now, knowing the forms of both $q(\mathbf{x}_t | \mathbf{x}_0)$ and $q(\mathbf{x}_{t-1} | \mathbf{x}_0)$, we can proceed to calculate the form of $q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0)$ by substituting into the Bayes rule expansion:

$$q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) = \frac{q(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{x}_0)q(\mathbf{x}_{t-1} | \mathbf{x}_0)}{q(\mathbf{x}_t | \mathbf{x}_0)} \quad (71)$$

$$= \frac{\mathcal{N}(\mathbf{x}_t; \sqrt{\alpha_t} \mathbf{x}_{t-1}, (1 - \alpha_t) \mathbf{I}) \mathcal{N}(\mathbf{x}_{t-1}; \sqrt{\bar{\alpha}_{t-1}} \mathbf{x}_0, (1 - \bar{\alpha}_{t-1}) \mathbf{I})}{\mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t} \mathbf{x}_0, (1 - \bar{\alpha}_t) \mathbf{I})} \quad (72)$$

$$\propto \exp \left\{ - \left[\frac{(\mathbf{x}_t - \sqrt{\alpha_t} \mathbf{x}_{t-1})^2}{2(1 - \alpha_t)} + \frac{(\mathbf{x}_{t-1} - \sqrt{\bar{\alpha}_{t-1}} \mathbf{x}_0)^2}{2(1 - \bar{\alpha}_{t-1})} - \frac{(\mathbf{x}_t - \sqrt{\bar{\alpha}_t} \mathbf{x}_0)^2}{2(1 - \bar{\alpha}_t)} \right] \right\} \quad (73)$$

$$= \exp \left\{ - \frac{1}{2} \left[\frac{(\mathbf{x}_t - \sqrt{\alpha_t} \mathbf{x}_{t-1})^2}{1 - \alpha_t} + \frac{(\mathbf{x}_{t-1} - \sqrt{\bar{\alpha}_{t-1}} \mathbf{x}_0)^2}{1 - \bar{\alpha}_{t-1}} - \frac{(\mathbf{x}_t - \sqrt{\bar{\alpha}_t} \mathbf{x}_0)^2}{1 - \bar{\alpha}_t} \right] \right\} \quad (74)$$

$$= \exp \left\{ - \frac{1}{2} \left[\frac{(-2\sqrt{\alpha_t} \mathbf{x}_t \mathbf{x}_{t-1} + \alpha_t \mathbf{x}_{t-1}^2)}{1 - \alpha_t} + \frac{(\mathbf{x}_{t-1}^2 - 2\sqrt{\bar{\alpha}_{t-1}} \mathbf{x}_{t-1} \mathbf{x}_0)}{1 - \bar{\alpha}_{t-1}} + C(\mathbf{x}_t, \mathbf{x}_0) \right] \right\} \quad (75)$$

$$\propto \exp \left\{ - \frac{1}{2} \left[-\frac{2\sqrt{\alpha_t} \mathbf{x}_t \mathbf{x}_{t-1}}{1 - \alpha_t} + \frac{\alpha_t \mathbf{x}_{t-1}^2}{1 - \alpha_t} + \frac{\mathbf{x}_{t-1}^2}{1 - \bar{\alpha}_{t-1}} - \frac{2\sqrt{\bar{\alpha}_{t-1}} \mathbf{x}_{t-1} \mathbf{x}_0}{1 - \bar{\alpha}_{t-1}} \right] \right\} \quad (76)$$

$$= \exp \left\{ - \frac{1}{2} \left[\left(\frac{\alpha_t}{1 - \alpha_t} + \frac{1}{1 - \bar{\alpha}_{t-1}} \right) \mathbf{x}_{t-1}^2 - 2 \left(\frac{\sqrt{\alpha_t} \mathbf{x}_t}{1 - \alpha_t} + \frac{\sqrt{\bar{\alpha}_{t-1}} \mathbf{x}_0}{1 - \bar{\alpha}_{t-1}} \right) \mathbf{x}_{t-1} \right] \right\} \quad (77)$$

$$= \exp \left\{ - \frac{1}{2} \left[\frac{\alpha_t(1 - \bar{\alpha}_{t-1}) + 1 - \alpha_t}{(1 - \alpha_t)(1 - \bar{\alpha}_{t-1})} \mathbf{x}_{t-1}^2 - 2 \left(\frac{\sqrt{\alpha_t} \mathbf{x}_t}{1 - \alpha_t} + \frac{\sqrt{\bar{\alpha}_{t-1}} \mathbf{x}_0}{1 - \bar{\alpha}_{t-1}} \right) \mathbf{x}_{t-1} \right] \right\} \quad (78)$$

$$= \exp \left\{ - \frac{1}{2} \left[\frac{\alpha_t - \bar{\alpha}_t + 1 - \alpha_t}{(1 - \alpha_t)(1 - \bar{\alpha}_{t-1})} \mathbf{x}_{t-1}^2 - 2 \left(\frac{\sqrt{\alpha_t} \mathbf{x}_t}{1 - \alpha_t} + \frac{\sqrt{\bar{\alpha}_{t-1}} \mathbf{x}_0}{1 - \bar{\alpha}_{t-1}} \right) \mathbf{x}_{t-1} \right] \right\} \quad (79)$$

$$= \exp \left\{ - \frac{1}{2} \left[\frac{1 - \bar{\alpha}_t}{(1 - \alpha_t)(1 - \bar{\alpha}_{t-1})} \mathbf{x}_{t-1}^2 - 2 \left(\frac{\sqrt{\alpha_t} \mathbf{x}_t}{1 - \alpha_t} + \frac{\sqrt{\bar{\alpha}_{t-1}} \mathbf{x}_0}{1 - \bar{\alpha}_{t-1}} \right) \mathbf{x}_{t-1} \right] \right\} \quad (80)$$

$$= \exp \left\{ - \frac{1}{2} \left(\frac{1 - \bar{\alpha}_t}{(1 - \alpha_t)(1 - \bar{\alpha}_{t-1})} \right) \left[\mathbf{x}_{t-1}^2 - 2 \frac{\left(\frac{\sqrt{\alpha_t} \mathbf{x}_t}{1 - \alpha_t} + \frac{\sqrt{\bar{\alpha}_{t-1}} \mathbf{x}_0}{1 - \bar{\alpha}_{t-1}} \right) \mathbf{x}_{t-1}}{\frac{1 - \bar{\alpha}_t}{(1 - \alpha_t)(1 - \bar{\alpha}_{t-1})}} \right] \right\} \quad (81)$$

$$= \exp \left\{ - \frac{1}{2} \left(\frac{1 - \bar{\alpha}_t}{(1 - \alpha_t)(1 - \bar{\alpha}_{t-1})} \right) \left[\mathbf{x}_{t-1}^2 - 2 \frac{\left(\frac{\sqrt{\alpha_t} \mathbf{x}_t}{1 - \alpha_t} + \frac{\sqrt{\bar{\alpha}_{t-1}} \mathbf{x}_0}{1 - \bar{\alpha}_{t-1}} \right) (1 - \alpha_t)(1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} \mathbf{x}_{t-1} \right] \right\} \quad (82)$$

$$= \exp \left\{ - \frac{1}{2} \left(\frac{1}{\frac{(1 - \alpha_t)(1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t}} \right) \left[\mathbf{x}_{t-1}^2 - 2 \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1}) \mathbf{x}_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t) \mathbf{x}_0}{1 - \bar{\alpha}_t} \mathbf{x}_{t-1} \right] \right\} \quad (83)$$

$$\propto \mathcal{N}(\mathbf{x}_{t-1}; \underbrace{\frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1}) \mathbf{x}_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t) \mathbf{x}_0}{1 - \bar{\alpha}_t}}_{\mu_q(\mathbf{x}_t, \mathbf{x}_0)}, \underbrace{\frac{(1 - \alpha_t)(1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} \mathbf{I}}_{\Sigma_q(t)}) \quad (84)$$

where in Equation 75, $C(\mathbf{x}_t, \mathbf{x}_0)$ is a constant term with respect to \mathbf{x}_{t-1} computed as a combination of only \mathbf{x}_t , \mathbf{x}_0 , and α values; this term is implicitly returned in Equation 84 to complete the square.

We have therefore shown that at each step, $\mathbf{x}_{t-1} \sim q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0)$ is normally distributed, with mean $\mu_q(\mathbf{x}_t, \mathbf{x}_0)$ that is a function of \mathbf{x}_t and \mathbf{x}_0 , and variance $\Sigma_q(t)$ as a function of α coefficients. These α coefficients are known and fixed at each timestep; they are either set permanently when modeled as hyperparameters, or treated as the current inference output of a network that seeks to model them. Following Equation 84, we can rewrite our variance equation as $\Sigma_q(t) = \sigma_q^2(t) \mathbf{I}$, where:

$$\sigma_q^2(t) = \frac{(1 - \alpha_t)(1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} \quad (85)$$

In order to match approximate denoising transition step $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)$ to ground-truth denoising transition step $q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0)$ as closely as possible, we can also model it as a Gaussian. Furthermore, as all α terms are known to be frozen at each timestep, we can immediately construct the variance of the approximate denoising transition step to also be $\Sigma_q(t) = \sigma_q^2(t) \mathbf{I}$. We must parameterize its mean $\mu_\theta(\mathbf{x}_t, t)$ as a function of \mathbf{x}_t , however, since $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)$ does not condition on \mathbf{x}_0 .

Recall that the KL Divergence between two Gaussian distributions is:

$$D_{\text{KL}}(\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x) \| \mathcal{N}(\mathbf{y}; \boldsymbol{\mu}_y, \boldsymbol{\Sigma}_y)) = \frac{1}{2} \left[\log \frac{|\boldsymbol{\Sigma}_y|}{|\boldsymbol{\Sigma}_x|} - d + \text{tr}(\boldsymbol{\Sigma}_y^{-1} \boldsymbol{\Sigma}_x) + (\boldsymbol{\mu}_y - \boldsymbol{\mu}_x)^T \boldsymbol{\Sigma}_y^{-1} (\boldsymbol{\mu}_y - \boldsymbol{\mu}_x) \right] \quad (86)$$

In our case, where we can set the variances of the two Gaussians to match exactly, optimizing the KL Divergence term reduces to minimizing the difference between the means of the two distributions:

$$\begin{aligned} & \arg \min_{\theta} D_{\text{KL}}(q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) \| p_{\theta}(\mathbf{x}_{t-1} | \mathbf{x}_t)) \\ &= \arg \min_{\theta} D_{\text{KL}}(\mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q(t)) \| \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_{\theta}, \boldsymbol{\Sigma}_q(t))) \end{aligned} \quad (87)$$

$$= \arg \min_{\theta} \frac{1}{2} \left[\log \frac{|\boldsymbol{\Sigma}_q(t)|}{|\boldsymbol{\Sigma}_q(t)|} - d + \text{tr}(\boldsymbol{\Sigma}_q(t)^{-1} \boldsymbol{\Sigma}_q(t)) + (\boldsymbol{\mu}_{\theta} - \boldsymbol{\mu}_q)^T \boldsymbol{\Sigma}_q(t)^{-1} (\boldsymbol{\mu}_{\theta} - \boldsymbol{\mu}_q) \right] \quad (88)$$

$$= \arg \min_{\theta} \frac{1}{2} [\log 1 - d + d + (\boldsymbol{\mu}_{\theta} - \boldsymbol{\mu}_q)^T \boldsymbol{\Sigma}_q(t)^{-1} (\boldsymbol{\mu}_{\theta} - \boldsymbol{\mu}_q)] \quad (89)$$

$$= \arg \min_{\theta} \frac{1}{2} [(\boldsymbol{\mu}_{\theta} - \boldsymbol{\mu}_q)^T \boldsymbol{\Sigma}_q(t)^{-1} (\boldsymbol{\mu}_{\theta} - \boldsymbol{\mu}_q)] \quad (90)$$

$$= \arg \min_{\theta} \frac{1}{2} [(\boldsymbol{\mu}_{\theta} - \boldsymbol{\mu}_q)^T (\sigma_q^2(t) \mathbf{I})^{-1} (\boldsymbol{\mu}_{\theta} - \boldsymbol{\mu}_q)] \quad (91)$$

$$= \arg \min_{\theta} \frac{1}{2 \sigma_q^2(t)} [\|\boldsymbol{\mu}_{\theta} - \boldsymbol{\mu}_q\|_2^2] \quad (92)$$

where we have written $\boldsymbol{\mu}_q$ as shorthand for $\boldsymbol{\mu}_q(\mathbf{x}_t, \mathbf{x}_0)$, and $\boldsymbol{\mu}_{\theta}$ as shorthand for $\boldsymbol{\mu}_{\theta}(\mathbf{x}_t, t)$ for brevity. In other words, we want to optimize a $\boldsymbol{\mu}_{\theta}(\mathbf{x}_t, t)$ that matches $\boldsymbol{\mu}_q(\mathbf{x}_t, \mathbf{x}_0)$, which from our derived Equation 84, takes the form:

$$\boldsymbol{\mu}_q(\mathbf{x}_t, \mathbf{x}_0) = \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})\mathbf{x}_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)\mathbf{x}_0}{1 - \bar{\alpha}_t} \quad (93)$$

As $\boldsymbol{\mu}_{\theta}(\mathbf{x}_t, t)$ also conditions on \mathbf{x}_t , we can match $\boldsymbol{\mu}_q(\mathbf{x}_t, \mathbf{x}_0)$ closely by setting it to the following form:

$$\boldsymbol{\mu}_{\theta}(\mathbf{x}_t, t) = \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})\mathbf{x}_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)\hat{\mathbf{x}}_{\theta}(\mathbf{x}_t, t)}{1 - \bar{\alpha}_t} \quad (94)$$

where $\hat{\mathbf{x}}_{\theta}(\mathbf{x}_t, t)$ is parameterized by a neural network that seeks to predict \mathbf{x}_0 from noisy image \mathbf{x}_t and time index t . Then, the optimization problem simplifies to:

$$\begin{aligned} & \arg \min_{\theta} D_{\text{KL}}(q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) \| p_{\theta}(\mathbf{x}_{t-1} | \mathbf{x}_t)) \\ &= \arg \min_{\theta} D_{\text{KL}}(\mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q(t)) \| \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_{\theta}, \boldsymbol{\Sigma}_q(t))) \end{aligned} \quad (95)$$

$$= \arg \min_{\theta} \frac{1}{2 \sigma_q^2(t)} \left[\left\| \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})\mathbf{x}_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)\hat{\mathbf{x}}_{\theta}(\mathbf{x}_t, t)}{1 - \bar{\alpha}_t} - \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})\mathbf{x}_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)\mathbf{x}_0}{1 - \bar{\alpha}_t} \right\|_2^2 \right] \quad (96)$$

$$= \arg \min_{\theta} \frac{1}{2 \sigma_q^2(t)} \left[\left\| \frac{\sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)\hat{\mathbf{x}}_{\theta}(\mathbf{x}_t, t)}{1 - \bar{\alpha}_t} - \frac{\sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)\mathbf{x}_0}{1 - \bar{\alpha}_t} \right\|_2^2 \right] \quad (97)$$

$$= \arg \min_{\theta} \frac{1}{2 \sigma_q^2(t)} \left[\left\| \frac{\sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)}{1 - \bar{\alpha}_t} (\hat{\mathbf{x}}_{\theta}(\mathbf{x}_t, t) - \mathbf{x}_0) \right\|_2^2 \right] \quad (98)$$

$$= \arg \min_{\theta} \frac{1}{2 \sigma_q^2(t)} \frac{\bar{\alpha}_{t-1}(1 - \alpha_t)^2}{(1 - \bar{\alpha}_t)^2} [\|\hat{\mathbf{x}}_{\theta}(\mathbf{x}_t, t) - \mathbf{x}_0\|_2^2] \quad \text{I predict the ground truth} \quad (99)$$

Therefore, optimizing a VDM boils down to learning a neural network to predict the original ground truth image from an arbitrarily noisified version of it [5]. Furthermore, minimizing the summation term of our derived ELBO objective (Equation 58) across all noise levels can be approximated by minimizing the expectation over all timesteps:

$$\arg \min_{\theta} \mathbb{E}_{t \sim U\{2, T\}} [\mathbb{E}_{q(\mathbf{x}_t | \mathbf{x}_0)} [D_{\text{KL}}(q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) \| p_{\theta}(\mathbf{x}_{t-1} | \mathbf{x}_t))]] \quad (100)$$

which can then be optimized using stochastic samples over timesteps.

Learning Diffusion Noise Parameters

Let us investigate how the noise parameters of a VDM can be jointly learned. One potential approach is to model α_t using a neural network $\hat{\alpha}_\eta(t)$ with parameters η . However, this is inefficient as inference must be performed multiple times at each timestep t to compute $\bar{\alpha}_t$. Whereas caching can mitigate this computational cost, we can also derive an alternate way to learn the diffusion noise parameters. By substituting our variance equation from Equation 85 into our derived per-timestep objective in Equation 99, we can reduce:

$$\frac{1}{2\sigma_q^2(t)} \frac{\bar{\alpha}_{t-1}(1-\alpha_t)^2}{(1-\bar{\alpha}_t)^2} [\|\hat{x}_\theta(\mathbf{x}_t, t) - \mathbf{x}_0\|_2^2] = \frac{1}{2 \frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}} \frac{\bar{\alpha}_{t-1}(1-\alpha_t)^2}{(1-\bar{\alpha}_t)^2} [\|\hat{x}_\theta(\mathbf{x}_t, t) - \mathbf{x}_0\|_2^2] \quad (101)$$

$$= \frac{1}{2} \frac{1-\bar{\alpha}_t}{(1-\alpha_t)(1-\bar{\alpha}_{t-1})} \frac{\bar{\alpha}_{t-1}(1-\alpha_t)^2}{(1-\bar{\alpha}_t)^2} [\|\hat{x}_\theta(\mathbf{x}_t, t) - \mathbf{x}_0\|_2^2] \quad (102)$$

$$= \frac{1}{2} \frac{\bar{\alpha}_{t-1}(1-\alpha_t)}{(1-\bar{\alpha}_{t-1})(1-\bar{\alpha}_t)} [\|\hat{x}_\theta(\mathbf{x}_t, t) - \mathbf{x}_0\|_2^2] \quad (103)$$

$$= \frac{1}{2} \frac{\bar{\alpha}_{t-1} - \bar{\alpha}_t}{(1-\bar{\alpha}_{t-1})(1-\bar{\alpha}_t)} [\|\hat{x}_\theta(\mathbf{x}_t, t) - \mathbf{x}_0\|_2^2] \quad (104)$$

$$= \frac{1}{2} \frac{\bar{\alpha}_{t-1} - \bar{\alpha}_{t-1}\bar{\alpha}_t + \bar{\alpha}_{t-1}\bar{\alpha}_t - \bar{\alpha}_t}{(1-\bar{\alpha}_{t-1})(1-\bar{\alpha}_t)} [\|\hat{x}_\theta(\mathbf{x}_t, t) - \mathbf{x}_0\|_2^2] \quad (105)$$

$$= \frac{1}{2} \frac{\bar{\alpha}_{t-1}(1-\bar{\alpha}_t) - \bar{\alpha}_t(1-\bar{\alpha}_{t-1})}{(1-\bar{\alpha}_{t-1})(1-\bar{\alpha}_t)} [\|\hat{x}_\theta(\mathbf{x}_t, t) - \mathbf{x}_0\|_2^2] \quad (106)$$

$$= \frac{1}{2} \left(\frac{\bar{\alpha}_{t-1}(1-\bar{\alpha}_t)}{(1-\bar{\alpha}_{t-1})(1-\bar{\alpha}_t)} - \frac{\bar{\alpha}_t(1-\bar{\alpha}_{t-1})}{(1-\bar{\alpha}_{t-1})(1-\bar{\alpha}_t)} \right) [\|\hat{x}_\theta(\mathbf{x}_t, t) - \mathbf{x}_0\|_2^2] \quad (107)$$

$$= \frac{1}{2} \left(\frac{\bar{\alpha}_{t-1}}{1-\bar{\alpha}_{t-1}} - \frac{\bar{\alpha}_t}{1-\bar{\alpha}_t} \right) [\|\hat{x}_\theta(\mathbf{x}_t, t) - \mathbf{x}_0\|_2^2] \quad (108)$$

Recall from Equation 70 that $q(\mathbf{x}_t | \mathbf{x}_0)$ is a Gaussian of form $\mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t} \mathbf{x}_0, (1-\bar{\alpha}_t) \mathbf{I})$. Then, following the definition of the signal-to-noise ratio (SNR) as $\text{SNR} = \frac{\mu^2}{\sigma^2}$, we can write the SNR at each timestep t as:

$$\text{SNR}(t) = \frac{\bar{\alpha}_t}{1-\bar{\alpha}_t} \quad (109)$$

Then, our derived Equation 108 (and Equation 99) can be simplified as:

$$\frac{1}{2\sigma_q^2(t)} \frac{\bar{\alpha}_{t-1}(1-\alpha_t)^2}{(1-\bar{\alpha}_t)^2} [\|\hat{x}_\theta(\mathbf{x}_t, t) - \mathbf{x}_0\|_2^2] = \frac{1}{2} (\text{SNR}(t-1) - \text{SNR}(t)) [\|\hat{x}_\theta(\mathbf{x}_t, t) - \mathbf{x}_0\|_2^2] \quad (110)$$

As the name implies, the SNR represents the ratio between the original signal and the amount of noise present; a higher SNR represents more signal and a lower SNR represents more noise. In a diffusion model, we require the SNR to monotonically decrease as timestep t increases; this formalizes the notion that perturbed input \mathbf{x}_t becomes increasingly noisy over time, until it becomes identical to a standard Gaussian at $t = T$.

Following the simplification of the objective in Equation 110, we can directly parameterize the SNR at each timestep using a neural network, and learn it jointly along with the diffusion model. As the SNR must monotonically decrease over time, we can represent it as:

$$\text{SNR}(t) = \exp(-\omega_\eta(t)) \quad (111)$$

where $\omega_\eta(t)$ is modeled as a monotonically increasing neural network with parameters η . Negating $\omega_\eta(t)$ results in a monotonically decreasing function, whereas the exponential forces the resulting term to be positive. Note that the objective in Equation 100 must now optimize over η as well. By combining our parameterization of SNR in Equation 111 with our definition of SNR in Equation 109, we can also explicitly derive elegant forms for the value of $\bar{\alpha}_t$ as well as for the value of $1-\bar{\alpha}_t$:

$$\frac{\bar{\alpha}_t}{1-\bar{\alpha}_t} = \exp(-\omega_\eta(t)) \quad (112)$$

$$\therefore \bar{\alpha}_t = \text{sigmoid}(-\omega_\eta(t)) \quad (113)$$

$$\therefore 1-\bar{\alpha}_t = \text{sigmoid}(\omega_\eta(t)) \quad (114)$$

These terms are necessary for a variety of computations; for example, during optimization, they are used to create arbitrarily noisy \mathbf{x}_t from input \mathbf{x}_0 using the reparameterization trick, as derived in Equation 69.

DDPM Diffusion : $q_t(x_t | x_0) = \mathcal{N}(x_t; \sqrt{\alpha_t} x_0, (1 - \bar{\alpha}_t) I)$

Reverse : $q_t(x_{t-1} | x_t, x_0) = \frac{q_t(x_t | x_{t-1}, x_0) q_t(x_{t-1} | x_0)}{q_t(x_t | x_0)} = \mathcal{N}(x_{t-1}; \mu(x_t, x_0), \sigma_t^2 I)$

Three Equivalent Interpretations

As we previously proved, a Variational Diffusion Model can be trained by simply learning a neural network to predict the original natural image x_0 from an arbitrary noised version x_t and its time index t . However, x_0 has two other equivalent parameterizations, which leads to two further interpretations for a VDM.

Firstly, we can utilize the reparameterization trick. In our derivation of the form of $q(x_t | x_0)$, we can rearrange Equation 69 to show that:

$$\hat{x}_{0|t} = \frac{1}{\sqrt{\bar{\alpha}_t}} (x_t - \sqrt{1 - \bar{\alpha}_t} \epsilon_0) \quad \text{U-net}$$

$$x_0 = \frac{x_t - \sqrt{1 - \bar{\alpha}_t} \epsilon_0}{\sqrt{\bar{\alpha}_t}} \quad (115)$$

Plugging this into our previously derived true denoising transition mean $\mu_q(x_t, x_0)$, we can rederive as:

$$\mu_q(x_t, x_0) = \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})x_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)x_0}{1 - \bar{\alpha}_t} \quad (116)$$

$$= \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})x_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t) \frac{x_t - \sqrt{1 - \bar{\alpha}_t} \epsilon_0}{\sqrt{\bar{\alpha}_t}}}{1 - \bar{\alpha}_t} \quad (117)$$

$$= \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})x_t + (1 - \alpha_t) \frac{x_t - \sqrt{1 - \bar{\alpha}_t} \epsilon_0}{\sqrt{\bar{\alpha}_t}}}{1 - \bar{\alpha}_t} \quad (118)$$

$$= \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})x_t}{1 - \bar{\alpha}_t} + \frac{(1 - \alpha_t)x_t}{(1 - \bar{\alpha}_t)\sqrt{\alpha_t}} - \frac{(1 - \alpha_t)\sqrt{1 - \bar{\alpha}_t}\epsilon_0}{(1 - \bar{\alpha}_t)\sqrt{\alpha_t}} \quad (119)$$

$$= \left(\frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} + \frac{1 - \alpha_t}{(1 - \bar{\alpha}_t)\sqrt{\alpha_t}} \right) x_t - \frac{(1 - \alpha_t)\sqrt{1 - \bar{\alpha}_t}\epsilon_0}{(1 - \bar{\alpha}_t)\sqrt{\alpha_t}} \quad (120)$$

$$= \left(\frac{\alpha_t(1 - \bar{\alpha}_{t-1})}{(1 - \bar{\alpha}_t)\sqrt{\alpha_t}} + \frac{1 - \alpha_t}{(1 - \bar{\alpha}_t)\sqrt{\alpha_t}} \right) x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}\sqrt{\alpha_t}} \epsilon_0 \quad (121)$$

$$= \frac{\alpha_t - \bar{\alpha}_t + 1 - \alpha_t}{(1 - \bar{\alpha}_t)\sqrt{\alpha_t}} x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}\sqrt{\alpha_t}} \epsilon_0 \quad (122)$$

$$= \frac{1 - \bar{\alpha}_t}{(1 - \bar{\alpha}_t)\sqrt{\alpha_t}} x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}\sqrt{\alpha_t}} \epsilon_0 \quad (123)$$

$$= \frac{1}{\sqrt{\alpha_t}} x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}\sqrt{\alpha_t}} \epsilon_0 \quad (124)$$

Therefore, we can set our approximate denoising transition mean $\mu_\theta(x_t, t)$ as:

$$\underline{\mu_\theta(x_t, t)} = \frac{1}{\sqrt{\alpha_t}} x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}\sqrt{\alpha_t}} \underline{\hat{\epsilon}_\theta(x_t, t)} \quad (125)$$

and the corresponding optimization problem becomes:

$$\begin{aligned} & \arg \min_{\theta} D_{\text{KL}}(q(x_{t-1} | x_t, x_0) \| p_{\theta}(x_{t-1} | x_t)) \\ &= \arg \min_{\theta} D_{\text{KL}}(\mathcal{N}(x_{t-1}; \mu_q, \Sigma_q(t)) \| \mathcal{N}(x_{t-1}; \mu_\theta, \Sigma_q(t))) \end{aligned} \quad \text{refer to Eq. 92} \quad (126)$$

$$= \arg \min_{\theta} \frac{1}{2\sigma_q^2(t)} \left[\left\| \frac{1}{\sqrt{\alpha_t}} x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}\sqrt{\alpha_t}} \hat{\epsilon}_\theta(x_t, t) - \frac{1}{\sqrt{\alpha_t}} x_t + \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}\sqrt{\alpha_t}} \epsilon_0 \right\|_2^2 \right] \quad (127)$$

$$= \arg \min_{\theta} \frac{1}{2\sigma_q^2(t)} \left[\left\| \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}\sqrt{\alpha_t}} \epsilon_0 - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}\sqrt{\alpha_t}} \hat{\epsilon}_\theta(x_t, t) \right\|_2^2 \right] \quad (128)$$

$$= \arg \min_{\theta} \frac{1}{2\sigma_q^2(t)} \left[\left\| \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}\sqrt{\alpha_t}} (\epsilon_0 - \hat{\epsilon}_\theta(x_t, t)) \right\|_2^2 \right] \quad (129)$$

$$= \arg \min_{\theta} \frac{1}{2\sigma_q^2(t)} \frac{(1 - \alpha_t)^2}{(1 - \bar{\alpha}_t)\alpha_t} [\|\epsilon_0 - \hat{\epsilon}_\theta(x_t, t)\|_2^2] \quad \text{II. predict the noise } \epsilon_0 \quad (130)$$

Here, $\hat{\epsilon}_\theta(\mathbf{x}_t, t)$ is a neural network that learns to predict the source noise $\epsilon_0 \sim \mathcal{N}(\epsilon; \mathbf{0}, \mathbf{I})$ that determines \mathbf{x}_t from \mathbf{x}_0 . We have therefore shown that learning a VDM by predicting the original image \mathbf{x}_0 is equivalent to learning to predict the noise; empirically, however, some works have found that predicting the noise resulted in better performance [5, 7]. 

To derive the third common interpretation of Variational Diffusion Models, we appeal to Tweedie's Formula [8]. In English, Tweedie's Formula states that the true mean of an exponential family distribution, given samples drawn from it, can be estimated by the maximum likelihood estimate of the samples (aka empirical mean) plus some correction term involving the score of the estimate. In the case of just one observed sample, the empirical mean is just the sample itself. It is commonly used to mitigate sample bias; if observed samples all lie on one end of the underlying distribution, then the negative score becomes large and corrects the naive maximum likelihood estimate of the samples towards the true mean.

Mathematically, for a Gaussian variable $\mathbf{z} \sim \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$, Tweedie's Formula states that:

$$\mathbb{E}[\boldsymbol{\mu}_z | \mathbf{z}] = \mathbf{z} + \boldsymbol{\Sigma}_z \nabla_{\mathbf{z}} \log p(\mathbf{z}) \quad \text{Bayes correction}$$

$\boldsymbol{\mu}_z$

In this case, we apply it to predict the true posterior mean of \mathbf{x}_t given its samples. From Equation 70, we know that:

$$q(\mathbf{x}_t | \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t} \mathbf{x}_0, (1 - \bar{\alpha}_t) \mathbf{I})$$

Then, by Tweedie's Formula, we have:

$$\mathbb{E}[\boldsymbol{\mu}_{x_t} | \mathbf{x}_t] = \mathbf{x}_t + (1 - \bar{\alpha}_t) \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t) \quad (131)$$

where we write $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t)$ as $\nabla \log p(\mathbf{x}_t)$ for notational simplicity. According to Tweedie's Formula, the best estimate for the true mean that \mathbf{x}_t is generated from, $\boldsymbol{\mu}_{x_t} = \sqrt{\bar{\alpha}_t} \mathbf{x}_0$, is defined as:

$$\sqrt{\bar{\alpha}_t} \mathbf{x}_0 = \mathbf{x}_t + (1 - \bar{\alpha}_t) \nabla \log p(\mathbf{x}_t) \quad (132)$$

$$\therefore \mathbf{x}_0 = \frac{\mathbf{x}_t + (1 - \bar{\alpha}_t) \nabla \log p(\mathbf{x}_t)}{\sqrt{\bar{\alpha}_t}} \quad (133)$$

Then, we can plug Equation 133 into our ground-truth denoising transition mean $\boldsymbol{\mu}_q(\mathbf{x}_t, \mathbf{x}_0)$ once again and derive a new form:

$$\boldsymbol{\mu}_q(\mathbf{x}_t, \mathbf{x}_0) = \frac{\sqrt{\bar{\alpha}_t}(1 - \bar{\alpha}_{t-1})\mathbf{x}_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)\mathbf{x}_0}{1 - \bar{\alpha}_t} \quad (134)$$

$$= \frac{\sqrt{\bar{\alpha}_t}(1 - \bar{\alpha}_{t-1})\mathbf{x}_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t) \frac{\mathbf{x}_t + (1 - \bar{\alpha}_t) \nabla \log p(\mathbf{x}_t)}{\sqrt{\bar{\alpha}_t}}}{1 - \bar{\alpha}_t} \quad (135)$$

$$= \frac{\sqrt{\bar{\alpha}_t}(1 - \bar{\alpha}_{t-1})\mathbf{x}_t + (1 - \alpha_t) \frac{\mathbf{x}_t + (1 - \bar{\alpha}_t) \nabla \log p(\mathbf{x}_t)}{\sqrt{\bar{\alpha}_t}}}{1 - \bar{\alpha}_t} \quad (136)$$

$$= \frac{\sqrt{\bar{\alpha}_t}(1 - \bar{\alpha}_{t-1})\mathbf{x}_t}{1 - \bar{\alpha}_t} + \frac{(1 - \alpha_t)\mathbf{x}_t}{(1 - \bar{\alpha}_t)\sqrt{\bar{\alpha}_t}} + \frac{(1 - \alpha_t)(1 - \bar{\alpha}_t) \nabla \log p(\mathbf{x}_t)}{(1 - \bar{\alpha}_t)\sqrt{\bar{\alpha}_t}} \quad (137)$$

$$= \left(\frac{\sqrt{\bar{\alpha}_t}(1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} + \frac{1 - \alpha_t}{(1 - \bar{\alpha}_t)\sqrt{\bar{\alpha}_t}} \right) \mathbf{x}_t + \frac{1 - \alpha_t}{\sqrt{\bar{\alpha}_t}} \nabla \log p(\mathbf{x}_t) \quad (138)$$

$$= \left(\frac{\alpha_t(1 - \bar{\alpha}_{t-1})}{(1 - \bar{\alpha}_t)\sqrt{\bar{\alpha}_t}} + \frac{1 - \alpha_t}{(1 - \bar{\alpha}_t)\sqrt{\bar{\alpha}_t}} \right) \mathbf{x}_t + \frac{1 - \alpha_t}{\sqrt{\bar{\alpha}_t}} \nabla \log p(\mathbf{x}_t) \quad (139)$$

$$= \frac{\alpha_t - \bar{\alpha}_t + 1 - \alpha_t}{(1 - \bar{\alpha}_t)\sqrt{\bar{\alpha}_t}} \mathbf{x}_t + \frac{1 - \alpha_t}{\sqrt{\bar{\alpha}_t}} \nabla \log p(\mathbf{x}_t) \quad (140)$$

$$= \frac{1 - \bar{\alpha}_t}{(1 - \bar{\alpha}_t)\sqrt{\bar{\alpha}_t}} \mathbf{x}_t + \frac{1 - \alpha_t}{\sqrt{\bar{\alpha}_t}} \nabla \log p(\mathbf{x}_t) \quad (141)$$

$$= \frac{1}{\sqrt{\bar{\alpha}_t}} \mathbf{x}_t + \frac{1 - \alpha_t}{\sqrt{\bar{\alpha}_t}} \nabla \log p(\mathbf{x}_t) \quad (142)$$

Therefore, we can also set our approximate denoising transition mean $\mu_\theta(\mathbf{x}_t, t)$ as:

$$\underline{\mu_\theta(\mathbf{x}_t, t)} = \frac{1}{\sqrt{\alpha_t}} \mathbf{x}_t + \frac{1 - \alpha_t}{\sqrt{\alpha_t}} \underline{s_\theta(\mathbf{x}_t, t)} \quad (143)$$

and the corresponding optimization problem becomes:

$$\begin{aligned} & \arg \min_{\theta} D_{\text{KL}}(q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) \| p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)) \\ &= \arg \min_{\theta} D_{\text{KL}}(\mathcal{N}(\mathbf{x}_{t-1}; \mu_q, \Sigma_q(t)) \| \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta, \Sigma_q(t))) \end{aligned} \quad (144)$$

$$= \arg \min_{\theta} \frac{1}{2\sigma_q^2(t)} \left[\left\| \frac{1}{\sqrt{\alpha_t}} \mathbf{x}_t + \frac{1 - \alpha_t}{\sqrt{\alpha_t}} s_\theta(\mathbf{x}_t, t) - \frac{1}{\sqrt{\alpha_t}} \mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{\alpha_t}} \nabla \log p(\mathbf{x}_t) \right\|_2^2 \right] \quad (145)$$

$$= \arg \min_{\theta} \frac{1}{2\sigma_q^2(t)} \left[\left\| \frac{1 - \alpha_t}{\sqrt{\alpha_t}} s_\theta(\mathbf{x}_t, t) - \frac{1 - \alpha_t}{\sqrt{\alpha_t}} \nabla \log p(\mathbf{x}_t) \right\|_2^2 \right] \quad (146)$$

$$= \arg \min_{\theta} \frac{1}{2\sigma_q^2(t)} \left[\left\| \frac{1 - \alpha_t}{\sqrt{\alpha_t}} (s_\theta(\mathbf{x}_t, t) - \nabla \log p(\mathbf{x}_t)) \right\|_2^2 \right] \quad (147)$$

$$= \arg \min_{\theta} \frac{1}{2\sigma_q^2(t)} \frac{(1 - \alpha_t)^2}{\alpha_t} \left[\|s_\theta(\mathbf{x}_t, t) - \nabla \log p(\mathbf{x}_t)\|_2^2 \right] \quad \text{III. predict the score function } \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t) \quad (148)$$

Here, $s_\theta(\mathbf{x}_t, t)$ is a neural network that learns to predict the score function $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t)$, which is the gradient of \mathbf{x}_t in data space, for any arbitrary noise level t .

The astute reader will notice that the score function $\nabla \log p(\mathbf{x}_t)$ looks remarkably similar in form to the source noise ϵ_0 . This can be shown explicitly by combining Tweedie's Formula (Equation 133) with the reparameterization trick (Equation 115):

$$\mathbf{x}_0 = \frac{\mathbf{x}_t + (1 - \bar{\alpha}_t) \nabla \log p(\mathbf{x}_t)}{\sqrt{\bar{\alpha}_t}} = \frac{\mathbf{x}_t - \sqrt{1 - \bar{\alpha}_t} \epsilon_0}{\sqrt{\bar{\alpha}_t}} \quad (149)$$

$$\therefore (1 - \bar{\alpha}_t) \nabla \log p(\mathbf{x}_t) = -\sqrt{1 - \bar{\alpha}_t} \epsilon_0 \quad (150)$$

$$\underline{\nabla \log p(\mathbf{x}_t)} = -\frac{1}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_0 \quad (151)$$

As it turns out, the two terms are off by a constant factor that scales with time! The score function measures how to move in data space to maximize the log probability; intuitively, since the source noise is added to a natural image to corrupt it, moving in its opposite direction "denoises" the image and would be the best update to increase the subsequent log probability. Our mathematical proof justifies this intuition; we have explicitly shown that learning to model the score function is equivalent to modeling the negative of the source noise (up to a scaling factor).

We have therefore derived three equivalent objectives to optimize a VDM: learning a neural network to predict the original image \mathbf{x}_0 , the source noise ϵ_0 , or the score of the image at an arbitrary noise level $\nabla \log p(\mathbf{x}_t)$. The VDM can be scalably trained by stochastically sampling timesteps t and minimizing the norm of the prediction with the ground truth target.

Score-based Generative Models

We have shown that a Variational Diffusion Model can be learned simply by optimizing a neural network $s_\theta(\mathbf{x}_t, t)$ to predict the score function $\nabla \log p(\mathbf{x}_t)$. However, in our derivation, the score term arrived from an application of Tweedie's Formula; this doesn't necessarily provide us with great intuition or insight into what exactly the score function is or why it is worth modeling. Fortunately, we can look to another class of generative models, Score-based Generative Models [9, 10, 11], for exactly this intuition. As it turns out, we can show that the VDM formulation we have previously derived has an equivalent Score-based Generative Modeling formulation, allowing us to flexibly switch between these two interpretations at will.