Appendix

A. Conditional Entropy Bounds Derivation

The conditional entropy $H_q\left(\mathbf{X}^{(t-1)}|\mathbf{X}^{(t)}\right)$ of a step in the reverse trajectory is

$$H_q\left(\mathbf{X}^{(t-1)},\mathbf{X}^{(t)}\right) = H_q\left(\mathbf{X}^{(t)},\mathbf{X}^{(t-1)}\right) \tag{24}$$

$$H_q\left(\mathbf{X}^{(t-1)}|\mathbf{X}^{(t)}\right) + H_q\left(\mathbf{X}^{(t)}\right) = H_q\left(\mathbf{X}^{(t)}|\mathbf{X}^{(t-1)}\right) + H_q\left(\mathbf{X}^{(t-1)}\right) \tag{25}$$

$$H_q\left(\mathbf{X}^{(t-1)}|\mathbf{X}^{(t)}\right) = H_q\left(\mathbf{X}^{(t)}|\mathbf{X}^{(t-1)}\right) + H_q\left(\mathbf{X}^{(t-1)}\right) - H_q\left(\mathbf{X}^{(t)}\right) \tag{26}$$

An upper bound on the entropy change can be constructed by observing that $\pi\left(\mathbf{y}\right)$ is the maximum entropy distribution. This holds without qualification for the binomial distribution, and holds for variance 1 training data for the Gaussian case. For the Gaussian case, training data must therefore be scaled to have unit norm for the following equalities to hold. It need not be whitened. The upper bound is derived as follows,

$$H_q\left(\mathbf{X}^{(t)}\right) \geq H_q\left(\mathbf{X}^{(t-1)}\right) \tag{27}$$

$$H_q\left(\mathbf{X}^{(t-1)}\right) - H_q\left(\mathbf{X}^{(t)}\right) \leq 0 \tag{28}$$

$$H_q\left(\mathbf{X}^{(t-1)}|\mathbf{X}^{(t)}\right) \leq H_q\left(\mathbf{X}^{(t)}|\mathbf{X}^{(t-1)}\right). \tag{29}$$

A lower bound on the entropy difference can be established by observing that additional steps in a Markov chain do not increase the information available about the initial state in the chain, and thus do not decrease the conditional entropy of the initial state,

$$H_q\left(\mathbf{X}^{(0)}|\mathbf{X}^{(t)}\right) \geq H_q\left(\mathbf{X}^{(0)}|\mathbf{X}^{(t-1)}\right) \tag{30}$$

$$H_q\left(\mathbf{X}^{(t-1)}\right) - H_q\left(\mathbf{X}^{(t)}\right) \geq H_q\left(\mathbf{X}^{(0)}|\mathbf{X}^{(t-1)}\right) + H_q\left(\mathbf{X}^{(t-1)}\right) - H_q\left(\mathbf{X}^{(0)}|\mathbf{X}^{(t)}\right) - H_q\left(\mathbf{X}^{(t)}\right) \tag{31}$$

$$H_q\left(\mathbf{X}^{(t-1)}\right) - H_q\left(\mathbf{X}^{(t)}\right) \geq H_q\left(\mathbf{X}^{(0)},\mathbf{X}^{(t-1)}\right) - H_q\left(\mathbf{X}^{(0)},\mathbf{X}^{(t)}\right) \tag{32}$$

$$H_q\left(\mathbf{X}^{(t-1)}\right) - H_q\left(\mathbf{X}^{(t)}\right) \geq H_q\left(\mathbf{X}^{(t-1)}|\mathbf{X}^{(0)}\right) - H_q\left(\mathbf{X}^{(t)}|\mathbf{X}^{(0)}\right) \tag{33}$$

$$H_q\left(\mathbf{X}^{(t-1)}|\mathbf{X}^{(t)}\right) \geq H_q\left(\mathbf{X}^{(t)}|\mathbf{X}^{(t-1)}\right) + H_q\left(\mathbf{X}^{(t-1)}|\mathbf{X}^{(0)}\right) - H_q\left(\mathbf{X}^{(t)}|\mathbf{X}^{(0)}\right). \tag{34}$$

Combining these expressions, we bound the conditional entropy for a single step,

$$H_q\left(\mathbf{X}^{(t)}|\mathbf{X}^{(t-1)}\right) \geq H_q\left(\mathbf{X}^{(t-1)}|\mathbf{X}^{(t)}\right) \geq H_q\left(\mathbf{X}^{(t)}|\mathbf{X}^{(t-1)}\right) + H_q\left(\mathbf{X}^{(t-1)}|\mathbf{X}^{(0)}\right) - H_q\left(\mathbf{X}^{(t)}|\mathbf{X}^{(0)}\right), \tag{35}$$

where both the upper and lower bounds depend only on the conditional forward trajectory $q\left(\mathbf{x}^{(1\cdots T)}|\mathbf{x}^{(0)}\right)$, and can be analytically computed.

B. Log Likelihood Lower Bound

The lower bound on the log likelihood is

$$L \geq K \tag{36}$$

$$K = \int d\mathbf{x}^{(0\cdots T)} q\left(\mathbf{x}^{(0\cdots T)}\right) \log\left[p\left(\mathbf{x}^{(T)}\right)\prod_{t=1}^{T}\frac{p\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right)}{q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right)}\right] \tag{37}$$

$$\tag{38}$$

B.1. Entropy of $p\left(\mathbf{X}^{(T)}\right)$

We can peel off the contribution from $p\left(\mathbf{X}^{(T)}\right)$, and rewrite it as an entropy,

$$K = \int d\mathbf{x}^{(0\cdots T)} q\left(\mathbf{x}^{(0\cdots T)}\right) \sum_{t=1}^{T} \log\left[\frac{p\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right)}{q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right)}\right] + \int d\mathbf{x}^{(T)} q\left(\mathbf{x}^{(T)}\right) \log p\left(\mathbf{x}^{(T)}\right) \tag{39}$$

$$= \int d\mathbf{x}^{(0\cdots T)} q\left(\mathbf{x}^{(0\cdots T)}\right) \sum_{t=1}^{T} \log\left[\frac{p\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right)}{q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right)}\right] + \int d\mathbf{x}^{(T)} q\left(\mathbf{x}^{(T)}\right) \log \pi\left(\mathbf{x}^{T}\right) \tag{40}$$

. (41)

By design, the cross entropy to $\pi\left(\mathbf{x}^{(t)}\right)$ is constant under our diffusion kernels, and equal to the entropy of $p\left(\mathbf{x}^{(T)}\right)$. Therefore,

$$K = \sum_{t=1}^{T} \int d\mathbf{x}^{(0\cdots T)} q\left(\mathbf{x}^{(0\cdots T)}\right) \log\left[\frac{p\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right)}{q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right)}\right] - H_p\left(\mathbf{X}^{(T)}\right). \tag{42}$$

B.2. Remove the edge effect at $t = 0$

In order to avoid edge effects, we set the final step of the reverse trajectory to be identical to the corresponding forward diffusion step,

$$p\left(\mathbf{x}^{(0)}|\mathbf{x}^{(1)}\right) = q\left(\mathbf{x}^{(1)}|\mathbf{x}^{(0)}\right) \frac{\pi\left(\mathbf{x}^{(0)}\right)}{\pi\left(\mathbf{x}^{(1)}\right)} = T_\pi\left(\mathbf{x}^{(0)}|\mathbf{x}^{(1)}; \beta_1\right). \tag{43}$$

We then use this equivalence to remove the contribution of the first time-step in the sum,

$$K = \sum_{t=2}^{T} \int d\mathbf{x}^{(0\cdots T)} q\left(\mathbf{x}^{(0\cdots T)}\right) \log\left[\frac{p\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right)}{q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right)}\right] + \int d\mathbf{x}^{(0)} d\mathbf{x}^{(1)} q\left(\mathbf{x}^{(0)}, \mathbf{x}^{(1)}\right) \log\left[\frac{q\left(\mathbf{x}^{(1)}|\mathbf{x}^{(0)}\right) \pi\left(\mathbf{x}^{(0)}\right)}{q\left(\mathbf{x}^{(1)}|\mathbf{x}^{(0)}\right) \pi\left(\mathbf{x}^{(1)}\right)}\right] - H_p\left(\mathbf{X}^{(T)}\right) \tag{44}$$

$$= \sum_{t=2}^{T} \int d\mathbf{x}^{(0\cdots T)} q\left(\mathbf{x}^{(0\cdots T)}\right) \log\left[\frac{p\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right)}{q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right)}\right] - H_p\left(\mathbf{X}^{(T)}\right), \tag{45}$$

where we again used the fact that by design $-\int d\mathbf{x}^{(t)} q\left(\mathbf{x}^{(t)}\right) \log \pi\left(\mathbf{x}^{(t)}\right) = H_p\left(\mathbf{X}^{(T)}\right)$ is a constant for all t.

B.3. Rewrite in terms of posterior $q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(0)}\right)$

Because the forward trajectory is a Markov process,

$$K = \sum_{t=2}^{T} \int d\mathbf{x}^{(0\cdots T)} q\left(\mathbf{x}^{(0\cdots T)}\right) \log\left[\frac{p\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right)}{q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}, \mathbf{x}^{(0)}\right)}\right] - H_p\left(\mathbf{X}^{(T)}\right). \tag{46}$$

Using Bayes' rule we can rewrite this in terms of a posterior and marginals from the forward trajectory,

$$K = \sum_{t=2}^{T} \int d\mathbf{x}^{(0\cdots T)} q\left(\mathbf{x}^{(0\cdots T)}\right) \log\left[\frac{p\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right)}{q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}, \mathbf{x}^{(0)}\right)} \frac{q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(0)}\right)}{q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(0)}\right)}\right] - H_p\left(\mathbf{X}^{(T)}\right). \tag{47}$$

B.4. Rewrite in terms of KL divergences and entropies

We then recognize that several terms are conditional entropies,

$$K = \sum_{t=2}^{T} \int d\mathbf{x}^{(0\cdots T)} q\left(\mathbf{x}^{(0\cdots T)}\right) \log\left[\frac{p\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right)}{q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)},\mathbf{x}^{(0)}\right)}\right] + \sum_{t=2}^{T}\left[H_q\left(\mathbf{X}^{(t)}|\mathbf{X}^{(0)}\right) - H_q\left(\mathbf{X}^{(t-1)}|\mathbf{X}^{(0)}\right)\right] - H_p\left(\mathbf{X}^{(T)}\right) \tag{48}$$

$$= \sum_{t=2}^{T} \int d\mathbf{x}^{(0\cdots T)} q\left(\mathbf{x}^{(0\cdots T)}\right) \log\left[\frac{p\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right)}{q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)},\mathbf{x}^{(0)}\right)}\right] + H_q\left(\mathbf{X}^{(T)}|\mathbf{X}^{(0)}\right) - H_q\left(\mathbf{X}^{(1)}|\mathbf{X}^{(0)}\right) - H_p\left(\mathbf{X}^{(T)}\right). \tag{49}$$

Finally we transform the log ratio of probability distributions into a KL divergence,

$$\begin{aligned} K = &-\sum_{t=2}^{T} \int d\mathbf{x}^{(0)} d\mathbf{x}^{(t)} q\left(\mathbf{x}^{(0)},\mathbf{x}^{(t)}\right) D_{KL}\left(q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)},\mathbf{x}^{(0)}\right)\middle\|p\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right)\right) \\ &+ H_q\left(\mathbf{X}^{(T)}|\mathbf{X}^{(0)}\right) - H_q\left(\mathbf{X}^{(1)}|\mathbf{X}^{(0)}\right) - H_p\left(\mathbf{X}^{(T)}\right). \end{aligned} \tag{50}$$

Note that the entropies can be analytically computed, and the KL divergence can be analytically computed given $\mathbf{x}^{(0)}$ and $\mathbf{x}^{(t)}$.

C. Markov Kernel of Perturbed Distribution

In Equations 19 and 20, the perturbed diffusion kernels are set as follows (unlike in the text body, we include the normalization constant)

$$\tilde{q}\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right) = \frac{q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right) r\left(\mathbf{x}^{(t)}\right)}{\int d\mathbf{x}^{(t)} q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right) r\left(\mathbf{x}^{(t)}\right)}, \tag{51}$$

$$\tilde{q}\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right) = \frac{q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right) r\left(\mathbf{x}^{(t-1)}\right)}{\int d\mathbf{x}^{(t-1)} q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right) r\left(\mathbf{x}^{(t-1)}\right)}, \tag{52}$$

or writing them instead in terms of the original transitions,

$$q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right) = \frac{\tilde{q}\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right) \int d\mathbf{x}^{(t)} q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right) r\left(\mathbf{x}^{(t)}\right)}{r\left(\mathbf{x}^{(t)}\right)}, \tag{53}$$

$$q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right) = \frac{\tilde{q}\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right) \int d\mathbf{x}^{(t-1)} q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right) r\left(\mathbf{x}^{(t-1)}\right)}{r\left(\mathbf{x}^{(t-1)}\right)}. \tag{54}$$

Similarly, we write Equation 16 in terms of the original forward distributions,

$$q\left(\mathbf{x}^{(t)}\right) = \frac{\tilde{q}\left(\mathbf{x}^{(t)}\right) \tilde{Z}_t}{r\left(\mathbf{x}^{(t)}\right)}. \tag{55}$$

We substitute into Equation 17,

$$q\left(\mathbf{x}^{(t-1)}\right) q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right) = q\left(\mathbf{x}^{(t)}\right) q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right), \tag{56}$$

$$\frac{\tilde{q}\left(\mathbf{x}^{(t-1)}\right) \tilde{Z}_{t-1}}{r\left(\mathbf{x}^{(t-1)}\right)} \frac{\tilde{q}\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right) \int d\mathbf{x}^{(t)} q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right) r\left(\mathbf{x}^{(t)}\right)}{r\left(\mathbf{x}^{(t)}\right)} = \frac{\tilde{q}\left(\mathbf{x}^{(t)}\right) \tilde{Z}_t}{r\left(\mathbf{x}^{(t)}\right)} \frac{\tilde{q}\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right) \int d\mathbf{x}^{(t-1)} q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right) r\left(\mathbf{x}^{(t-1)}\right)}{r\left(\mathbf{x}^{(t-1)}\right)}, \tag{57}$$

$$\tilde{q}\left(\mathbf{x}^{(t-1)}\right) \tilde{q}\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right) \tilde{Z}_{t-1} \int d\mathbf{x}^{(t)} q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right) r\left(\mathbf{x}^{(t)}\right) = \tilde{q}\left(\mathbf{x}^{(t)}\right) \tilde{q}\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right) \tilde{Z}_t \int d\mathbf{x}^{(t-1)} q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right) r\left(\mathbf{x}^{(t-1)}\right). \tag{58}$$

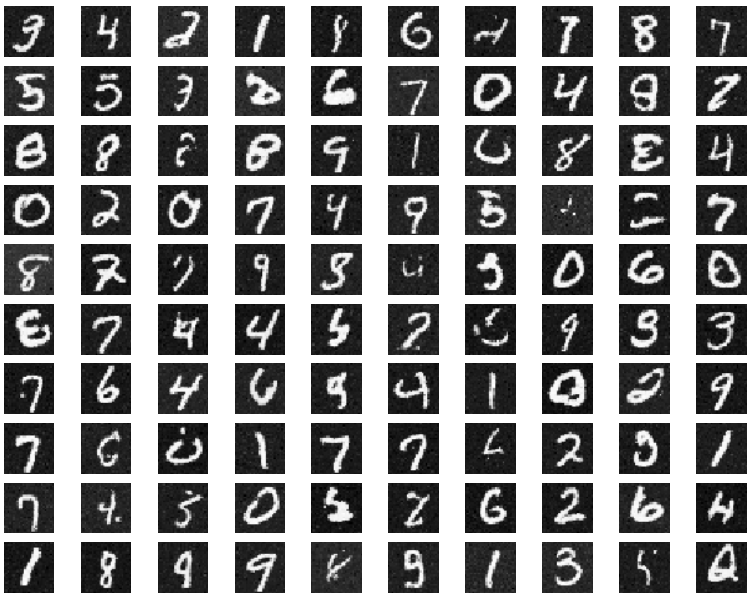

Figure App.1. Samples from a diffusion probabilistic model trained on MNIST digits. Note that unlike many MNIST sample figures, these are true samples rather than the mean of the Gaussian or binomial distribution from which samples would be drawn.

We then substitute $\tilde{Z}_t = \int d\mathbf{x}^{(t)} q\left(\mathbf{x}^{(t)}\right) r\left(\mathbf{x}^{(t)}\right)$,

$$\tilde{q}\left(\mathbf{x}^{(t-1)}\right)\tilde{q}\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right)\int d\mathbf{x}^{(t-1)} q\left(\mathbf{x}^{(t-1)}\right) r\left(\mathbf{x}^{(t-1)}\right)\int d\mathbf{x}^{(t)} q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right) r\left(\mathbf{x}^{(t)}\right) \tag{59}$$
$$= \tilde{q}\left(\mathbf{x}^{(t)}\right)\tilde{q}\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right)\int d\mathbf{x}^{(t)} q\left(\mathbf{x}^{(t)}\right) r\left(\mathbf{x}^{(t)}\right)\int d\mathbf{x}^{(t-1)} q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right) r\left(\mathbf{x}^{(t-1)}\right),$$

$$\tilde{q}\left(\mathbf{x}^{(t-1)}\right)\tilde{q}\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right)\int d\mathbf{x}^{(t-1)} d\mathbf{x}^{(t)} q\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right) q\left(\mathbf{x}^{(t-1)}\right) r\left(\mathbf{x}^{(t)}\right) r\left(\mathbf{x}^{(t-1)}\right) \tag{60}$$
$$= \tilde{q}\left(\mathbf{x}^{(t)}\right)\tilde{q}\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right)\int d\mathbf{x}^{(t-1)} d\mathbf{x}^{(t)} q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right) q\left(\mathbf{x}^{(t)}\right) r\left(\mathbf{x}^{(t-1)}\right) r\left(\mathbf{x}^{(t)}\right),$$

$$\tilde{q}\left(\mathbf{x}^{(t-1)}\right)\tilde{q}\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right)\int d\mathbf{x}^{(t-1)} d\mathbf{x}^{(t)} q\left(\mathbf{x}^{(t-1)}, \mathbf{x}^{(t)}\right) r\left(\mathbf{x}^{(t)}\right) r\left(\mathbf{x}^{(t-1)}\right) \tag{61}$$
$$= \tilde{q}\left(\mathbf{x}^{(t)}\right)\tilde{q}\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right)\int d\mathbf{x}^{(t-1)} d\mathbf{x}^{(t)} q\left(\mathbf{x}^{(t-1)}, \mathbf{x}^{(t)}\right) r\left(\mathbf{x}^{(t-1)}\right) r\left(\mathbf{x}^{(t)}\right).$$

We can now cancel the identical integrals on each side, achieving our goal of showing that the choice of perturbed Markov transitions in Equations 19 and 20 satisfy Equation 18,

$$\tilde{q}\left(\mathbf{x}^{(t-1)}\right)\tilde{q}\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}\right) = \tilde{q}\left(\mathbf{x}^{(t)}\right)\tilde{q}\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right). \tag{62}$$

		Gaussian	*Binomial*
Well behaved (analytically tractable) distribution	$\pi\left(\mathbf{x}^{(T)}\right) =$	$\mathcal{N}\left(\mathbf{x}^{(T)}; \mathbf{0}, \mathbf{I}\right)$	$\mathcal{B}\left(\mathbf{x}^{(T)}; 0.5\right)$
Forward diffusion kernel	$q\left(\mathbf{x}^{(t)} \vert \mathbf{x}^{(t-1)}\right) =$	$\mathcal{N}\left(\mathbf{x}^{(t)}; \mathbf{x}^{(t-1)}\sqrt{1-\beta_t}, \mathbf{I}\beta_t\right)$	$\mathcal{B}\left(\mathbf{x}^{(t)}; \mathbf{x}^{(t-1)}\left(1-\beta_t\right) + 0.5\beta_t\right)$
Reverse diffusion kernel	$p\left(\mathbf{x}^{(t-1)} \vert \mathbf{x}^{(t)}\right) =$	$\mathcal{N}\left(\mathbf{x}^{(t-1)}; \mathbf{f}_\mu\left(\mathbf{x}^{(t)}, t\right), \mathbf{f}_\Sigma\left(\mathbf{x}^{(t)}, t\right)\right)$	$\mathcal{B}\left(\mathbf{x}^{(t-1)}; \mathbf{f}_b\left(\mathbf{x}^{(t)}, t\right)\right)$
Training targets		$\mathbf{f}_\mu\left(\mathbf{x}^{(t)}, t\right), \mathbf{f}_\Sigma\left(\mathbf{x}^{(t)}, t\right), \beta_{1\cdots T}$	$\mathbf{f}_b\left(\mathbf{x}^{(t)}, t\right)$
Forward distribution	$q\left(\mathbf{x}^{(0\cdots T)}\right) =$	$q\left(\mathbf{x}^{(0)}\right)\prod_{t=1}^{T} q\left(\mathbf{x}^{(t)} \vert \mathbf{x}^{(t-1)}\right)$	
Reverse distribution	$p\left(\mathbf{x}^{(0\cdots T)}\right) =$	$\pi\left(\mathbf{x}^{(T)}\right)\prod_{t=1}^{T} p\left(\mathbf{x}^{(t-1)} \vert \mathbf{x}^{(t)}\right)$	
Log likelihood	$L =$	$\int d\mathbf{x}^{(0)} q\left(\mathbf{x}^{(0)}\right) \log p\left(\mathbf{x}^{(0)}\right)$	
Lower bound on log likelihood	$K =$	$-\sum_{t=2}^{T} \mathbb{E}_{q\left(\mathbf{x}^{(0)}, \mathbf{x}^{(t)}\right)}\left[D_{KL}\left(q\left(\mathbf{x}^{(t-1)} \vert \mathbf{x}^{(t)}, \mathbf{x}^{(0)}\right) \middle\vert\middle\vert p\left(\mathbf{x}^{(t-1)} \vert \mathbf{x}^{(t)}\right)\right)\right] + H_q\left(\mathbf{X}^{(T)} \vert \mathbf{X}^{(0)}\right) - H_q\left(\mathbf{X}^{(1)} \vert \mathbf{X}^{(0)}\right) - H_p\left(\mathbf{X}^{(T)}\right)$	
Perturbed forward diffusion kernel	$\tilde{q}\left(\mathbf{x}^{(t)} \vert \mathbf{x}^{(t-1)}\right) =$	$\mathcal{N}\left(\mathbf{x}^{(t)}; \mathbf{x}^{(t-1)}\sqrt{1-\beta_t} + \left(\frac{\beta_t}{2}\right)^{\frac{1}{2}} \frac{\partial \log r\left(\mathbf{x}^{(t)}\right)}{\partial \mathbf{x}^{(t)}}, \mathbf{I}\beta_t\right)$	$\mathcal{B}\left(x_i^{(t)}; \frac{b_i^t d_i^t}{x_i^t d_i^t + (1-b_i^t)(1-d_i^t)}\right)$
Perturbed reverse diffusion kernel	$\tilde{p}\left(\mathbf{x}^{(t-1)} \vert \mathbf{x}^{(t)}\right) =$	$\mathcal{N}\left(\mathbf{x}^{(t-1)}; \mathbf{f}_\mu\left(\mathbf{x}^{(t)}, t\right) + \left(\frac{\mathbf{f}_\Sigma\left(\mathbf{x}^{(t)}, t\right)}{2}\right)^{\frac{1}{2}} \frac{\partial \log r\left(\mathbf{x}^{(t-1)}\right)}{\partial \mathbf{x}^{(t)}}, \mathbf{f}_\Sigma\left(\mathbf{x}^{(t)}, t\right)\right)$	$\mathcal{B}\left(x_i^{(t-1)}; \frac{c_i^{t-1} d_i^{t-1}}{x_i^{t-1} d_i^{t-1} + (1-c_i^{t-1})(1-d_i^{t-1})}\right)$

Table C.1. The key equations in this paper for the specific cases of Gaussian and binomial diffusion processes. $\mathcal{N}\left(u; \mu, \Sigma\right)$ is a Gaussian distribution with mean μ and covariance Σ. $\mathcal{B}\left(u; r\right)$ is the distribution for a single Bernoulli trial, with $u = 1$ occurring with probability r, and $u = 0$ occurring with probability $1 - r$. Finally, for the perturbed Bernoulli trials $b_i^t = \mathbf{x}^{(t-1)}\left(1-\beta_t\right) + 0.5\beta_t$, $c_i^t = \left[\mathbf{f}_b\left(\mathbf{x}^{(t+1)}, t\right)\right]_i$, and $d_i^t = r\left(x_i^{(t)} = 1\right)$, and the distribution is given for a single bit i.

D. Experimental Details

D.1. Toy Problems

D.1.1. Swiss Roll

A probabilistic model was built of a two dimensional swiss roll distribution. The generative model $p\left(\mathbf{x}^{(0\cdots T)}\right)$ consisted of 40 time steps of Gaussian diffusion initialized at an identity-covariance Gaussian distribution. A (normalized) radial basis function network with a single hidden layer and 16 hidden units was trained to generate the mean and covariance functions $\mathbf{f}_\mu\left(\mathbf{x}^{(t)}, t\right)$ and a diagonal $\mathbf{f}_\Sigma\left(\mathbf{x}^{(t)}, t\right)$ for the reverse trajectory. The top, readout, layer for each function was learned independently for each time step, but for all other layers weights were shared across all time steps and both functions. The top layer output of $\mathbf{f}_\Sigma\left(\mathbf{x}^{(t)}, t\right)$ was passed through a sigmoid to restrict it between 0 and 1. As can be seen in Figure 1, the swiss roll distribution was successfully learned.

D.1.2. Binary Heartbeat Distribution

A probabilistic model was trained on simple binary sequences of length 20, where a 1 occurs every 5th time bin, and the remainder of the bins are 0. The generative model consisted of 2000 time steps of binomial diffusion initialized at an independent binomial distribution with the same mean activity as the data ($p\left(x_i^{(T)} = 1\right) = 0.2$). A multilayer perceptron with sigmoid nonlinearities, 20 input units and three hidden layers with 50 units each was trained to generate the Bernoulli rates $\mathbf{f}_b\left(\mathbf{x}^{(t)}, t\right)$ of the reverse trajectory. The top, readout, layer was learned independently for each time step, but for all other layers weights were shared across all time steps. The top layer output was passed through a sigmoid to restrict it between 0 and 1. As can be seen in Figure 2, the heartbeat distribution was successfully learned. The log likelihood under the true generating process is $\log_2\left(\frac{1}{5}\right) = -2.322$ bits per sequence. As can be seen in Figure 2 and Table 1 learning was nearly perfect.

D.2. Images

D.2.1. Architecture

Readout In all cases, a convolutional network was used to produce a vector of outputs $\mathbf{y}_i \in \mathcal{R}^{2J}$ for each image pixel i. The entries in $\mathbf{y}_i$ are divided into two equal sized subsets, $\mathbf{y}^\mu$ and $\mathbf{y}^\Sigma$.

Temporal Dependence The convolution output $\mathbf{y}^\mu$ is used as per-pixel weighting coefficients in a sum over time-dependent "bump" functions, generating an output $\mathbf{z}_i^\mu \in \mathcal{R}$ for each pixel i,

$$\mathbf{z}_i^\mu = \sum_{j=1}^{J} \mathbf{y}_{ij}^\mu g_j\left(t\right). \tag{63}$$

The bump functions consist of

$$g_j\left(t\right) = \frac{\exp\left(-\frac{1}{2w^2}\left(t-\tau_j\right)^2\right)}{\sum_{k=1}^{J}\exp\left(-\frac{1}{2w^2}\left(t-\tau_k\right)^2\right)}, \tag{64}$$

where $\tau_j \in (0, T)$ is the bump center, and w is the spacing between bump centers. $\mathbf{z}^\Sigma$ is generated in an identical way, but using y^Σ.

For all image experiments a number of timesteps $T = 1000$ was used, except for the bark dataset which used $T = 500$.

Mean and Variance Finally, these outputs are combined to produce a diffusion mean and variance prediction for each pixel i,

$$\Sigma_{ii} = \sigma\left(z_i^\Sigma + \sigma^{-1}\left(\beta_t\right)\right), \tag{65}$$

$$\mu_i = \left(x_i - z_i^\mu\right)\left(1 - \Sigma_{ii}\right) + z_i^\mu. \tag{66}$$

where both Σ and μ are parameterized as a perturbation around the forward diffusion kernel $T_\pi\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}; \beta_t\right)$, and z_i^μ is the mean of the equilibrium distribution that would result from applying $p\left(\mathbf{x}^{(t-1)}|\mathbf{x}^{(t)}\right)$ many times. Σ is restricted to be a diagonal matrix.

Multi-Scale Convolution We wish to accomplish goals that are often achieved with pooling networks – specifically, we wish to discover and make use of long-range and multi-scale dependencies in the training data. However, since the network output is a vector of coefficients for every pixel it is important to generate a full resolution rather than down-sampled feature map. We therefore define multi-scale-convolution layers that consist of the following steps:

1. Perform mean pooling to downsample the image to multiple scales. Downsampling is performed in powers of two.
2. Performing convolution at each scale.
3. Upsample all scales to full resolution, and sum the resulting images.
4. Perform a pointwise nonlinear transformation, consisting of a soft relu ($\log\left[1 + \exp\left(\cdot\right)\right]$).

The composition of the first three linear operations resembles convolution by a multiscale convolution kernel, up to blocking artifacts introduced by upsampling. This method of achieving multiscale convolution was described in (Barron et al., 2013).

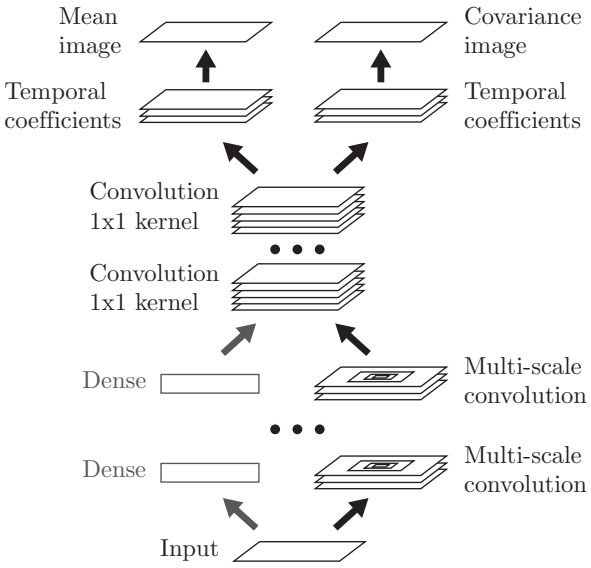

Figure D.1. Network architecture for mean function $\mathbf{f}_\mu\left(\mathbf{x}^{(t)}, t\right)$ and covariance function $\mathbf{f}_\Sigma\left(\mathbf{x}^{(t)}, t\right)$, for experiments in Section 3.2. The input image $\mathbf{x}^{(t)}$ passes through several layers of multi-scale convolution (Section D.2.1). It then passes through several convolutional layers with 1×1 kernels. This is equivalent to a dense transformation performed on each pixel. A linear transformation generates coefficients for readout of both mean $\mu^{(t)}$ and covariance $\Sigma^{(t)}$ for each pixel. Finally, a time dependent readout function converts those coefficients into mean and covariance images, as described in Section D.2.1. For CIFAR-10 a dense (or fully connected) pathway was used in parallel to the multi-scale convolutional pathway. For MNIST, the dense pathway was used to the exclusion of the multi-scale convolutional pathway.

Dense Layers Dense (acting on the full image vector) and kernel-width-1 convolutional (acting separately on the feature vector for each pixel) layers share the same form. They consist of a linear transformation, followed by a tanh nonlinearity.