NoProp Forward Process

Jeff Beck

October 12, 2025

Abstract

This is a short write up of an alternative derivation of the Continuous Time NoProp algorithm for learning to invert a normalizing flow without backpropagation. The original NoProp paper started with a discrete formulation of the problem and then took the limit of a large number of layers. This derivation starts with an inhomogeneous but asymptotically stationary continuous time Ornstein-Uhlenbeck process for the backward process and repeats the derivation ultimately arriving at the same objective function for learning, but a different forward model for prediction. We then show that this difference in the forward models is typically only relevant for the early phase of the denoising process when utilizing common noise schedules as its dominant effect is on the time scale of the dynamics not the endpoint.

1 Introduction

NoProp is amazing. Everyone should be using it. There is just one little problem with the continuous time version, but, do not worry, here is the fix and it does not really impact performance for commonly used noise schedules.

2 The Ornstein-Uhlenbeck Process

Like diffusion training protocols, the cornerstone of the continuous time NoProp algorithm is an OU process that progressively adds noise to a given target, z_T until it becomes indistinguishable from a unit Gaussian at time $t = 0$. This backward process is then iteratively denoised by a neural network with the goal of reproducing the the target. The simplicity and ease with which this process can be sampled at any time t is ultimately what drives the efficiency of the algorithm. An asymptotically stationary reverse OU process that tends toward a unit gaussian as $t \to -\infty$ is characterized by the equation,

$$dz = \frac{1}{2}\delta(t)z + \sqrt{\delta(t)}dW_t \tag{1}$$

which implies that the marginal mean and variance at time t are given by

$$\begin{aligned} \mu(t) &= z_T \exp\left(-\frac{1}{2}\Delta(t)\right) && (2) \\ \sigma^2(t) &= 1 - \exp\left(-\Delta(t)\right) && (3) \end{aligned}$$

where $\delta(t)$ provides the noise schedule and $\Delta(t) = \int_t^T \delta(s)ds$ is the cumulative noise added to the reverse process between times t and T. Note that in the original NoProp paper, the variance conditioned on the terminal state is given by $1 - \bar{\alpha}(t)$ so that $\log \bar{\alpha}(t) = -\Delta(t)$ for a continuous time process. The NoProp objective is given by the KL-divergence between a forward process $p(z_t|z_{t-dt}, x)$ and the associated $q(z_t|z_{t-dt}, z_T)$ derived from the reverse process. This q is easily computed from Bayes rule via inspection of the joint distribution conditioned on z_T. Defining $d\Delta(t) = \int_{t-dt}^t \delta(s)ds \approx -\Delta'(t)dt$ we have

$$
\begin{aligned}
\log q(z_t, z_{t-dt}|z_T) &\propto \log q(z_{t-dt}|z_t) + \log q(z_t|z_T) && (4)\\
&= -\frac{1}{2}\frac{\left(z_{t-dt} - z_t \exp\left(-\frac{1}{2}d\Delta(t)\right)\right)^2}{1-\exp\left(-d\Delta(t)\right)} && (5)\\
&\quad -\frac{1}{2}\frac{\left(z_t - z_T \exp\left(-\frac{1}{2}\Delta(t)\right)\right)^2}{1-\exp\left(-\Delta(t)\right)} && (6)\\
& && (7)
\end{aligned}
$$

Collecting the quadratic and linear terms in z_t we find the parameters for $q(z_t|z_{t-dt}, z_T)$ and dropping the explicit time dependence:

$$
\begin{aligned}
\frac{\mu_t}{\sigma_t^2} &= z_{t-dt}\frac{\exp\left(-\frac{1}{2}d\Delta(t)\right)}{1-\exp\left(-d\Delta(t)\right)} + z_T\frac{\exp\left(-\frac{1}{2}\Delta(t)\right)}{1-\exp\left(-\Delta(t)\right)} && (8)\\
\frac{1}{\sigma_t^2} &= \frac{\exp\left(-d\Delta(t)\right)}{1-\exp\left(-d\Delta(t)\right)} + \frac{1}{1-\exp\left(-\Delta(t)\right)} && (9)
\end{aligned}
$$

or equivalently

$$
\begin{aligned}
\mu_t &= \frac{z_{t-dt}e^{-d\Delta(t)/2}(1-e^{-\Delta(t)}) + z_T e^{-\Delta(t)/2}(1-e^{-d\Delta(t)})}{e^{-d\Delta(t)}(1-e^{-\Delta(t)}) + (1-e^{d\Delta(t)})} && (10)\\
\sigma_t^2 &= \frac{(1-e^{-d\Delta(t)})(1-e^{-\Delta(t)})}{e^{-d\Delta(t)}(1-e^{-\Delta(t)}) + (1-e^{-d\Delta(t)})} && (11)
\end{aligned}
$$

Rearranging terms a bit to make it easier to collect $O(dt)$ quantities yields

$$
\begin{aligned}
\mu_t &= \frac{z_{t-dt} + z_{t-dt}(e^{d\Delta(t)/2}-1) + z_T\frac{e^{-\Delta(t)/2}}{1-e^{-\Delta(t)}}(e^{d\Delta(t)}-1)}{1+(e^{d\Delta(t)}-1)/(1-e^{-\Delta(t)})} && (12)\\
\sigma_t^2 &= \frac{(e^{d\Delta(t)}-1)}{1+(e^{d\Delta(t)}-1)/(1-e^{-\Delta(t)})} && (13)
\end{aligned}
$$

which for small $d\Delta(t)$ yields $\sigma_t^2 = d\Delta(t)$ as expected. Similarly, we have

$$
\begin{aligned}
\mu_t &= z_{t-dt} + d\Delta(t)\frac{e^{-\Delta(t)/2}}{1-e^{-\Delta(t)}}z_T + d\Delta(t)z_{t-dt} - \frac{d\Delta(t)}{1-e^{-\Delta(t)}}z_{t-dt} && (14)\\
&= z_{t-dt} + d\Delta(t)\frac{e^{-\Delta(t)/2}}{1-e^{-\Delta(t)}}\left(z_T - z_{t-dt}\cosh(\Delta/2)\right) && (15)
\end{aligned}
$$

As in the original paper, we assume that the only difference between the forward process, $p(z_t|z_{t-dt}, x)$ and the reverse of the backward process $q(z_t|z_{t-dt}, z_T)$ is that the forward process uses a neural network to approximate $z_T \approx u_t(z_{t-dt}, x)$ while the backward process does not. As a result, the KL divergence between forward and backward steps is simply

$$
\begin{aligned}
KL(q(z_t|z_{t-dt}, z_T), p(z_t|z_{t-dt}, x)) &= \frac{1}{2}d\Delta(t)\frac{e^{-\Delta(t)}(t)}{(1-e^{-\Delta(t)})^2}\left|u_t(z_{t-dt}, x) - z_T\right|^2 && (16)\\
&= \frac{1}{2}d\Delta(t)\frac{\bar{\alpha}(t)}{(1-\bar{\alpha}(t))^2}\left|u_t(z_{t-dt}, x) - z_T\right|^2 && (17)
\end{aligned}
$$

where we have used the identity $\bar{\alpha}(t) = \exp -\Delta(t)$

This provides the desired loss term but an additional term related to the instantaneous noise added due to the backward process, $d\Delta(t)$. Fortunately, when everyone is well behaved this positive valued noise addition term can be related directly to $\Delta(t)$ and subsequently $\bar{\alpha}(t)$ since

$$
\begin{aligned}
d\Delta(t) &= -\Delta'(t)dt && (18)\\
&= \frac{\bar{\alpha}'(t)}{\bar{\alpha}(t)}dt && (19)
\end{aligned}
$$

This ultimately results in the original NoProp loss function

$$
\begin{aligned}
KL(q(z_t|z_{t-dt}, z_T), p(z_t|z_{t-dt}, x)) &= \frac{1}{2}\frac{\bar{\alpha}'(t)}{(1-\bar{\alpha}(t))^2}\left|u_t(z_{t-dt}, x) - z_T\right|^2 && (20)\\
&= \frac{1}{2}SNR'(t)\left|u_t(z_{t-dt}, x) - z_T\right|^2 && (21)
\end{aligned}
$$

Since

$$
\begin{aligned}
SNR(t) &= \frac{\bar{\alpha}(t)}{1-\bar{\alpha}(t)} && (22)\\
SNR'(t) &= \frac{\alpha'(t)}{(1-\bar{\alpha}(t))^2} && (23)
\end{aligned}
$$

So yes, everyone is on the same page regarding loss functions. This is good because it is an awesome loss function. Where things differ is in regards to the forward process. It is foundational to the NoProp approach that the only difference between the conditional forward process and the conditional forward process obtained from the backward process is that the backward process has access to the target, z_T, while the forward process must approximate it via the action of a neural network, $u_t(z_{t-dt}, x)$. Written in terms of $\alpha(t)$, the conditional mean of the forward process derived here is distinct from the NoProp approach due to its dependence on $\alpha'(t)$. In particular, the forward process has mean given by

$$
\begin{aligned}
\mu_t &= z_{t-dt} - \Delta'(t)\frac{e^{-\Delta(t)/2}}{1-e^{-\Delta(t)}}\left(u_t(z_{t-dt}, x) - z_{t-dt}\cosh(\Delta/2)\right)dt && (24)\\
&= z_{t-dt} + \frac{\bar{\alpha}'(t)}{\bar{\alpha}(t)(1-\bar{\alpha}(t))}\left(\sqrt{\alpha(t)}u_t(z_{t-dt}, x) - \frac{1+\bar{\alpha}(t)}{2}z_{t-dt}\right) && (25)
\end{aligned}
$$

When written as a dynamics on the mean each term in this equation as a sensible interpretation.

$$
\frac{dz}{dt} = \frac{\bar{\alpha}'(t)}{\bar{\alpha}(t)(1-\bar{\alpha}(t))}\left(\sqrt{\alpha(t)}u_t(z{,}x) - \frac{1+\bar{\alpha}(t)}{2}z\right) \qquad (26)
$$

$\bar{\alpha}(t)$ is an increasing quantity, so the prefactor is positive, which means that z is more or less attracted to a scaled version of the network output. Recall that $\bar{\alpha}'(t)/\bar{\alpha}(t)$ is the noise added during the backward process, so the presence of this term encourages the model to speed up dynamics during particularly noisy phases, i.e. the attraction is stronger when more noise is injected. The $1-\bar{\alpha}(t)$ in the demoninator further strengthens the attractor as the system approaches its endpoint. The term $\sqrt{\bar{\alpha}(t)}$ represents the projection of the network estimate of the target to the current time point. Thus, this quantity represents where the backward process places the mean at time t. The prefactor on z inside the parenthesis is the most mysterious term. It seems to encourage the dynamical system to overshoot the scaled estimate by a factor related to how much noise is still coming.

Finally, when faced with a dynamical system of this form, it is not uncommon to rescale time to eliminate the annoying prefactors that depend on $\bar{\alpha}(t)$. For example, we can let $\bar{\alpha}(t) = \text{sigmoid}(\gamma(t))$ which results in

$$
\frac{dz}{dt} = \gamma'(t)\left(\sqrt{\alpha(t)}u_t(z, x) - \frac{1+\alpha(t)}{2}z\right) \qquad (27)
$$

If $\gamma(t)$ is linear, this gives constant time scaling. For flexible noise schedules, this further motivates the choice of noise schedule paramterization used in the original paper. It is also worth noting that, for typical noise schedules, $\alpha(t)$ is close to 1 as t approaches the end point of the denoising process. This results in approximate dynamics of the form

$$\frac{dz}{dt} \approx \gamma'(t)\,(u_t(z,x) - z) \tag{28}$$

Thus, near the end point of the denoising dynamics, the difference between this approximate dynamics and the dynamics of the original NoProp denoising dynamics is given entirely by a difference in the time constant of integration. That is, both approaches have the same fixed point as $t \to 1$. As a result, this correction has minimal impact on final performance for commonly used noise schedules because it gets the last few denoising steps more or less correct. This is likely why the error was overlooked and why this slightly more rigorous derivation does not merit publication on its own.

Also, there could be a mistake or two in this derivation, so...

References