3 Aug

生成扩散模型漫谈（五）：一般框架之SDE篇

By 苏剑林 | 2022-08-03 | 185258位读者 引用

在写生成扩散模型的第一篇文章时，就有读者在评论区推荐了宋飏博士的论文《Score-Based Generative Modeling through Stochastic Differential Equations》，可以说该论文构建了一个相当一般化的生成扩散模型理论框架，将DDPM、SDE、ODE等诸多结果联系了起来。诚然，这是一篇好论文，但并不是一篇适合初学者的论文，里边直接用到了随机微分方程（SDE）、Fokker-Planck方程、得分匹配等大量结果，上手难度还是颇大的。

不过，在经过了前四篇文章的积累后，现在我们可以尝试去学习一下这篇论文了。在接下来的文章中，笔者将尝试从尽可能少的理论基础出发，尽量复现原论文中的推导结果。

随机微分

在DDPM中，扩散过程被划分为了固定的T步，还是用《生成扩散模型漫谈（一）：DDPM = 拆楼 + 建楼》的类比来说，就是“拆楼”和“建楼”都被事先划分为了T步，这个划分有着相当大的人为性。事实上，真实的“拆”、“建”过程应该是没有刻意划分的步骤的，我们可以将它们理解为一个在时间上连续的变换过程，可以用随机微分方程（Stochastic Differential Equation，SDE）来描述。

为此，我们用下述SDE描述前向过程（“拆楼”）：

$$d\boldsymbol{x} = \boldsymbol{f}_t(\boldsymbol{x})dt + g_t d\boldsymbol{w} \tag{1}$$

相信很多读者都对SDE很陌生，笔者也只是在硕士阶段刚好接触过一段时间，略懂皮毛。不过不懂不要紧，我们只需要将它看成是下述离散形式在$\Delta t \to 0$时的极限：

$$\boldsymbol{x}_{t+\Delta t} - \boldsymbol{x}_t = \boldsymbol{f}_t(\boldsymbol{x}_t)\Delta t + g_t\sqrt{\Delta t}\boldsymbol{\varepsilon},\quad \boldsymbol{\varepsilon}\sim\mathcal{N}(\boldsymbol{0},\boldsymbol{I}) \tag{2}$$

再直白一点，如果假设拆楼需要1天，那么拆楼就是$\boldsymbol{x}$从$t=0$到$t=1$的变化过程，每

一小步的变化我们可以用上述方程描述。至于时间间隔Δt，我们并没有做特殊限制，只是越小的Δt意味着是对原始SDE越好的近似，如果取$\Delta t = 0.001$，那就对应于原来的$T = 1000$，如果是$\Delta t = 0.01$则对应于$T = 100$，等等。也就是说，在连续时间的SDE视角之下，不同的T是SDE不同的离散化程度的体现，它们会自动地导致相似的结果，我们不需要事先指定T，而是根据实际情况下的精确度来取适当的T进行数值计算。

所以，引入SDE形式来描述扩散模型的本质好处是“将理论分析和代码实现分离开来”，我们可以借助连续性SDE的数学工具对它做分析，而实践的时候，则只需要用任意适当的离散化方案对SDE进行数值计算。

对于式(2)，读者可能比较有疑惑的是为什么右端第一项是$\mathcal{O}(\Delta t)$的，而第二项是$\mathcal{O}(\sqrt{\Delta t})$的？也就是说为什么随机项的阶要比确定项的阶要高？这个还真不是那么容易解释，也是SDE比较让人迷惑的地方之一。简单来说，就是$\boldsymbol{\varepsilon}$一直服从标准正态分布，如果随机项的权重也是$\mathcal{O}(\Delta t)$，那么由于标准正态分布的均值为$\boldsymbol{0}$、协方差为$\boldsymbol{I}$，临近的随机效应会相互抵消掉，要放大到$\mathcal{O}(\sqrt{\Delta t})$才能在长期结果中体现出随机效应的作用。

逆向方程

用概率的语言，式(2)意味着条件概率为

$$\begin{aligned} p(\boldsymbol{x}_{t+\Delta t}|\boldsymbol{x}_t) =&\, \mathcal{N}\left(\boldsymbol{x}_{t+\Delta t};\boldsymbol{x}_t + \boldsymbol{f}_t(\boldsymbol{x}_t)\Delta t, g_t^2\Delta t\,\boldsymbol{I}\right)\\ \propto&\, \exp\left(-\frac{\Vert\boldsymbol{x}_{t+\Delta t} - \boldsymbol{x}_t - \boldsymbol{f}_t(\boldsymbol{x}_t)\Delta t\Vert^2}{2g_t^2\Delta t}\right) \end{aligned}\tag{3}$$

简单起见，这里没有写出无关紧要的归一化因子。按照DDPM的思想，我们最终是想要从“拆楼”的过程中学会“建楼”，即得到$p(\boldsymbol{x}_t|\boldsymbol{x}_{t+\Delta t})$，为此，我们像《生成扩散模型漫谈（三）：DDPM = 贝叶斯 + 去噪》一样，用贝叶斯定理：

$$\begin{aligned}
p(\boldsymbol{x}_t|\boldsymbol{x}_{t+\Delta t}) =&\, \frac{p(\boldsymbol{x}_{t+\Delta t}|\boldsymbol{x}_t)p(\boldsymbol{x}_t)}{p(\boldsymbol{x}_{t+\Delta t})} = p(\boldsymbol{x}_{t+\Delta t}|\boldsymbol{x}_t)\exp(\log p(\boldsymbol{x}_t) - \log p(\boldsymbol{x}_{t+\Delta t})) \\
\propto&\, \exp\left(-\frac{\Vert \boldsymbol{x}_{t+\Delta t} - \boldsymbol{x}_t - \boldsymbol{f}_t(\boldsymbol{x}_t)\Delta t\Vert^2}{2g_t^2\Delta t} + \log p(\boldsymbol{x}_t) - \log p(\boldsymbol{x}_{t+\Delta t})\right)
\end{aligned}$$

不难发现，当Δt足够小时，只有当$\boldsymbol{x}_{t+\Delta t}$与$\boldsymbol{x}_t$足够接近时，$p(\boldsymbol{x}_{t+\Delta t}|\boldsymbol{x}_t)$才会明显不等于0，反过来也只有这种情况下$p(\boldsymbol{x}_t|\boldsymbol{x}_{t+\Delta t})$才会明显不等于0。因此，我们只需要对$\boldsymbol{x}_{t+\Delta t}$与$\boldsymbol{x}_t$足够接近时的情形做近似分析，为此，我们可以用泰勒展开：

$$\log p(\boldsymbol{x}_{t+\Delta t}) \approx \log p(\boldsymbol{x}_t) + (\boldsymbol{x}_{t+\Delta t} - \boldsymbol{x}_t)\cdot \nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t) + \Delta t\frac{\partial}{\partial t}\log p(\boldsymbol{x}_t) \tag{5}$$

注意不要忽略了$\frac{\partial}{\partial t}$项，因为$p(\boldsymbol{x}_t)$实际上是“$t$时刻随机变量等于$\boldsymbol{x}_t$的概率密度”，而$p(\boldsymbol{x}_{t+\Delta t})$实际上是“$t+\Delta t$时刻随机变量等于$\boldsymbol{x}_{t+\Delta t}$的概率密度”，也就是说$p(\boldsymbol{x}_t)$实际上同时是$t$和$\boldsymbol{x}_t$的函数，所以要多一项$t$的偏导数。代入到式(4)后，配方得到

$$p(\boldsymbol{x}_t|\boldsymbol{x}_{t+\Delta t}) \propto \exp\Bigg(-\frac{\Vert \boldsymbol{x}_{t+\Delta t} - \boldsymbol{x}_t - \left[\boldsymbol{f}_t(\boldsymbol{x}_t) - g_t^2\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t)\right]\Delta t\Vert^2}{2g_t^2\Delta t} + \mathcal{O}(\Delta t$$

当$\Delta t\to 0$时，$\mathcal{O}(\Delta t)\to 0$不起作用，因此

$$\begin{aligned}
p(\boldsymbol{x}_t|\boldsymbol{x}_{t+\Delta t}) \propto&\, \exp\left(-\frac{\Vert \boldsymbol{x}_{t+\Delta t} - \boldsymbol{x}_t - \left[\boldsymbol{f}_t(\boldsymbol{x}_t) - g_t^2\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t)\right]\Delta t\Vert^2}{2g_t^2\Delta t}\right) \\
\approx&\, \exp\Bigg(-\frac{\Vert \boldsymbol{x}_t - \boldsymbol{x}_{t+\Delta t} + \left[\boldsymbol{f}_{t+\Delta t}(\boldsymbol{x}_{t+\Delta t}) - g_{t+\Delta t}^2\nabla_{\boldsymbol{x}_{t+\Delta t}}\log p(\boldsymbol{x}_{t+\Delta t})\right]\Delta}{2g_{t+\Delta t}^2\Delta t}
\end{aligned}$$

即$p(\boldsymbol{x}_t|\boldsymbol{x}_{t+\Delta t})$近似一个均值为$\boldsymbol{x}_{t+\Delta t} - \left[\boldsymbol{f}_{t+\Delta t}(\boldsymbol{x}_{t+\Delta t}) - g_{t+\Delta t}^2\nabla_{\boldsymbol{x}_{t+\Delta t}}\log p(\boldsymbol{x}_{t+\Delta t})\right]\Delta t$、协方差为$g_{t+\Delta t}^2\Delta t\,\boldsymbol{I}$的正态分布，取$\Delta t\to 0$的极限，那么对应于SDE：

$$d\boldsymbol{x} = \left[\boldsymbol{f}_t(\boldsymbol{x}) - g_t^2\nabla_{\boldsymbol{x}}\log p_t(\boldsymbol{x})\right]dt + g_t d\boldsymbol{w} \tag{8}$$

这就是反向过程对应的SDE，最早出现在

《Reverse-Time Diffusion Equation Models》中。这里我们特意在p处标注了下标t，以突出这是t时刻的分布。

得分匹配

现在我们已经得到了逆向的SDE为(8)，如果进一步知道$\nabla_{\boldsymbol{x}}\log p_t(\boldsymbol{x})$，那么就可以通过离散化格式

$$\boldsymbol{x}_t - \boldsymbol{x}_{t+\Delta t} = -\left[\boldsymbol{f}_{t+\Delta t}(\boldsymbol{x}_{t+\Delta t}) - g_{t+\Delta t}^2\nabla_{\boldsymbol{x}_{t+\Delta t}}\log p(\boldsymbol{x}_{t+\Delta t})\right]\Delta t - g_{t+\Delta t}\sqrt{\Delta t}\boldsymbol{\varepsilon}$$

来逐步完成“建楼”的生成过程【其中$\boldsymbol{\varepsilon}\sim\mathcal{N}(\boldsymbol{0},\boldsymbol{I})$】，从而完成一个生成扩散模型的构建。

那么如何得到$\nabla_{\boldsymbol{x}}\log p_t(\boldsymbol{x})$呢？$t$时刻的$p_t(\boldsymbol{x})$就是前面的$p(\boldsymbol{x}_t)$，它的含义就是$t$时刻的边缘分布。在实际使用时，我们一般会设计能找到$p(\boldsymbol{x}_t|\boldsymbol{x}_0)$解析解的模型，这意味着

$$p(\boldsymbol{x}_t|\boldsymbol{x}_0) = \lim_{\Delta t\to 0}\int\cdots\iint p(\boldsymbol{x}_t|\boldsymbol{x}_{t-\Delta t})p(\boldsymbol{x}_{t-\Delta t}|\boldsymbol{x}_{t-2\Delta t})\cdots p(\boldsymbol{x}_{\Delta t}|\boldsymbol{x}_0)d\boldsymbol{x}_{t-\Delta t}d\boldsymbol{x}_{t-2\Delta t}\cdots d\boldsymbol{x}_{\Delta}$$

是可以直接求出的，比如当$\boldsymbol{f}_t(\boldsymbol{x})$是关于$\boldsymbol{x}$的线性函数时，$p(\boldsymbol{x}_t|\boldsymbol{x}_0)$就可以解析求解。在此前提下，有

$$p(\boldsymbol{x}_t) = \int p(\boldsymbol{x}_t|\boldsymbol{x}_0)\tilde{p}(\boldsymbol{x}_0)d\boldsymbol{x}_0 = \mathbb{E}_{\boldsymbol{x}_0}\left[p(\boldsymbol{x}_t|\boldsymbol{x}_0)\right] \tag{11}$$

于是

$$\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t) = \frac{\mathbb{E}_{\boldsymbol{x}_0}\left[\nabla_{\boldsymbol{x}_t}p(\boldsymbol{x}_t|\boldsymbol{x}_0)\right]}{\mathbb{E}_{\boldsymbol{x}_0}\left[p(\boldsymbol{x}_t|\boldsymbol{x}_0)\right]} = \frac{\mathbb{E}_{\boldsymbol{x}_0}\left[p(\boldsymbol{x}_t|\boldsymbol{x}_0)\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t|\boldsymbol{x}_0)\right]}{\mathbb{E}_{\boldsymbol{x}_0}\left[p(\boldsymbol{x}_t|\boldsymbol{x}_0)\right]} \tag{12}$$

可以看到最后的式子具有“$\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t|\boldsymbol{x}_0)$的加权平均”的形式，由于假设了$p(\boldsymbol{x}_t|\boldsymbol{x}_0)$有解析解，因此上式实际上是能够直接估算的，然而它涉及到对全体训练样本$\boldsymbol{x}_0$的平均，一来计算量大，二来泛化能力也不够好。因此，我们希望用神经网络学一个函数$\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t,t)$，使得它能够直接计算$\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t)$。

很多读者应该对如下结果并不陌生（或者推导一遍也不困难）：

$$\mathbb{E}[\boldsymbol{x}] = \mathop{\text{argmin}}_{\boldsymbol{\mu}} \mathbb{E}_{\boldsymbol{x}}\left[\Vert\boldsymbol{\mu} - \boldsymbol{x}\Vert^2\right] \tag{13}$$

即要让$\boldsymbol{\mu}$等于$\boldsymbol{x}$的均值，只需要最小化$\Vert\boldsymbol{\mu} - \boldsymbol{x}\Vert^2$的均值。同理，要让$\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)$等于$\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t|\boldsymbol{x}_0)$的加权平均【即$\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t)$】，则只需要最小化$\Vert\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) - \nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t|\boldsymbol{x}_0)\Vert^2$的加权平均，即

$$\frac{\mathbb{E}_{\boldsymbol{x}_0}\left[p(\boldsymbol{x}_t|\boldsymbol{x}_0)\Vert\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) - \nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t|\boldsymbol{x}_0)\Vert^2\right]}{\mathbb{E}_{\boldsymbol{x}_0}\left[p(\boldsymbol{x}_t|\boldsymbol{x}_0)\right]} \tag{14}$$

分母的$\mathbb{E}_{\boldsymbol{x}_0}\left[p(\boldsymbol{x}_t|\boldsymbol{x}_0)\right]$只是起到调节Loss权重的作用，简单起见我们可以直接去掉它，这不会影响最优解的结果。最后我们再对$\boldsymbol{x}_t$积分（相当于对于每一个$\boldsymbol{x}_t$都要最小化上述损失），得到最终的损失函数

$$\begin{aligned}
&\int \mathbb{E}_{\boldsymbol{x}_0}\left[p(\boldsymbol{x}_t|\boldsymbol{x}_0)\Vert\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) - \nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t|\boldsymbol{x}_0)\Vert^2\right] d\boldsymbol{x}_t \\
=&\, \mathbb{E}_{\boldsymbol{x}_0,\boldsymbol{x}_t \sim p(\boldsymbol{x}_t|\boldsymbol{x}_0)\tilde{p}(\boldsymbol{x}_0)}\left[\Vert\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) - \nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t|\boldsymbol{x}_0)\Vert^2\right]
\end{aligned} \tag{15}$$

这就是“（条件）得分匹配”的损失函数，之前我们在《从去噪自编码器到生成模型》推导的去噪自编码器的解析解，也是它的一个特例。得分匹配的最早出处可以追溯到2005年的论文《Estimation of Non-Normalized Statistical Models by Score Matching》，至于条件得分匹配的最早出处，笔者追溯到的是2011年的论文《A Connection Between Score Matching and Denoising Autoencoders》。

不过，虽然该结果跟得分匹配是一样的，但其实在这一节的推导中，我们已经抛开了“得分”的概念了，纯粹是由目标自然地引导出来的答案，笔者认为这样的处理过程更有启发性，希望这一推导能降低大家对得分匹配的理解难度。

结果倒推

至此，我们构建了生成扩散模型的一般流程：

1、通过随机微分方程(1)定义“拆楼”（前向过程）；

2、求$p(\boldsymbol{x}_t|\boldsymbol{x}_0)$的表达式；

3、通过损失函数(15)训练$\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t,t)$（得分匹配）；

4、用$\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t,t)$替换式$(8)$的$\nabla_{\boldsymbol{x}}\log p_t(\boldsymbol{x})$，完成“建楼”（反向过程）。

可能大家看到SDE、微分方程等字眼，天然就觉得“恐慌”，但本质上来说，SDE只是个“幌子”，实际上将对SDE的理解转换到式(2)和式(3)上后，完全就可以抛开SDE的概念了，因此概念上其实是没有太大难度的。

不难发现，定义一个随机微分方程(1)是很容易的，但是从(1)求解$p(\boldsymbol{x}_t|\boldsymbol{x}_0)$却是不容易的。原论文的剩余篇幅，主要是对两个有实用性的例子推导和实验。然而，既然求解$p(\boldsymbol{x}_t|\boldsymbol{x}_0)$不容易，那么按照笔者的看法，与其先定义$(1)$再求解$p(\boldsymbol{x}_t|\boldsymbol{x}_0)$，倒不如像DDIM一样，先定义$p(\boldsymbol{x}_t|\boldsymbol{x}_0)$，然后再来反推对应的SDE?

例如，我们先定义

$$p(\boldsymbol{x}_t|\boldsymbol{x}_0)=\mathcal{N}(\boldsymbol{x}_t;\bar{\alpha}_t\boldsymbol{x}_0,\bar{\beta}_t^2\boldsymbol{I})\tag{16}$$

并且不失一般性假设起点是$t=0$，终点是$t=1$，那么$\bar{\alpha}_t,\bar{\beta}_t$要满足的边界就是

$$\bar{\alpha}_0=1,\quad\bar{\alpha}_1=0,\quad\bar{\beta}_0=0,\quad\bar{\beta}_1=1\tag{17}$$

当然，上述边界条件理论上足够近似就行，也不一定非要精确相等，比如上一篇文章我们分析过DDPM相当于选择了$\bar{\alpha}_t=e^{-5t^2}$，当$t=1$时结果为$e^{-5}\approx 0$。

有了$p(\boldsymbol{x}_t|\boldsymbol{x}_0)$，我们去反推$(1)$，本质上就是要求解$p(\boldsymbol{x}_{t+\Delta t}|\boldsymbol{x}_t)$，它要满足

$$p(\boldsymbol{x}_{t+\Delta t}|\boldsymbol{x}_0)=\int p(\boldsymbol{x}_{t+\Delta t}|\boldsymbol{x}_t)p(\boldsymbol{x}_t|\boldsymbol{x}_0)d\boldsymbol{x}_t\tag{18}$$

我们考虑线性的解，即

$$d\boldsymbol{x} = f_t\boldsymbol{x}dt + g_t d\boldsymbol{w} \tag{19}$$

跟《生成扩散模型漫谈（四）：DDIM = 高观点DDPM》一样，我们写出

记号	含义	采样
$p(\boldsymbol{x}_{t+\Delta t}\vert\boldsymbol{x}_0)$	$\mathcal{N}(\boldsymbol{x}_t;\bar{\alpha}_{t+\Delta t}\boldsymbol{x}_0,\bar{\beta}_{t+\Delta t}^2\boldsymbol{I})$	$\boldsymbol{x}_{t+\Delta t} = \bar{\alpha}_{t+\Delta t}\boldsymbol{x}_0 + \bar{\beta}_{t+\Delta t}\boldsymbol{\varepsilon}$
$p(\boldsymbol{x}_t\vert\boldsymbol{x}_0)$	$\mathcal{N}(\boldsymbol{x}_t;\bar{\alpha}_t\boldsymbol{x}_0,\bar{\beta}_t^2\boldsymbol{I})$	$\boldsymbol{x}_t = \bar{\alpha}_t\boldsymbol{x}_0 + \bar{\beta}_t\boldsymbol{\varepsilon}_1$
$p(\boldsymbol{x}_{t+\Delta t}\vert\boldsymbol{x}_t)$	$\mathcal{N}(\boldsymbol{x}_{t+\Delta t};(1+f_t\Delta t)\boldsymbol{x}_t,g_t^2\Delta t\,\boldsymbol{I})$	$\boldsymbol{x}_{t+\Delta t} = (1+f_t\Delta t)\boldsymbol{x}_t + g_t\sqrt{\Delta}$
$\int p(\boldsymbol{x}_{t+\Delta t}\vert\boldsymbol{x}_t)$ $p(\boldsymbol{x}_t\vert\boldsymbol{x}_0)d\boldsymbol{x}_t$		$\boldsymbol{x}_{t+\Delta t}$ $= (1+f_t\Delta t)\boldsymbol{x}_t + g_t\sqrt{\Delta t}\boldsymbol{\varepsilon}_2$ $= (1+f_t\Delta t)(\bar{\alpha}_t\boldsymbol{x}_0 + \bar{\beta}_t\boldsymbol{\varepsilon}_1) + g_t\sqrt{\Delta t}\boldsymbol{\varepsilon}_2$ $= (1+f_t\Delta t)\bar{\alpha}_t\boldsymbol{x}_0 + ((1+f_t\Delta t)\bar{\beta}_t\boldsymbol{\varepsilon}_1 +$

由此可得

$$\begin{aligned}\bar{\alpha}_{t+\Delta t} =&\, (1+f_t\Delta t)\bar{\alpha}_t \\ \bar{\beta}_{t+\Delta t}^2 =&\, (1+f_t\Delta t)^2\bar{\beta}_t^2 + g_t^2\Delta t\end{aligned} \tag{20}$$

令$\Delta t\to 0$，分别解得

$$f_t = \frac{d}{dt}(\ln\bar{\alpha}_t) = \frac{1}{\bar{\alpha}_t}\frac{d\bar{\alpha}_t}{dt},\quad g_t^2 = \bar{\alpha}_t^2\frac{d}{dt}\left(\frac{\bar{\beta}_t^2}{\bar{\alpha}_t^2}\right) = 2\bar{\alpha}_t\bar{\beta}_t\frac{d}{dt}\left(\frac{\bar{\beta}_t}{\bar{\alpha}_t}\right) \tag{21}$$

取$\bar{\alpha}_t\equiv 1$时，结果就是论文中的VE-SDE（Variance Exploding SDE）；而如果取$\bar{\alpha}_t^2 + \bar{\beta}_t^2 = 1$时，结果就是原论文中的VP-SDE（Variance Preserving SDE）。

至于损失函数，此时我们可以算得

$$\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t|\boldsymbol{x}_0) = -\frac{\boldsymbol{x}_t - \bar{\alpha}_t\boldsymbol{x}_0}{\bar{\beta}_t^2} = -\frac{\boldsymbol{\varepsilon}}{\bar{\beta}_t} \tag{22}$$

第二个等号是因为$\boldsymbol{x}_t = \bar{\alpha}_t\boldsymbol{x}_0 + \bar{\beta}_t\boldsymbol{\varepsilon}$，为了跟以往的结果对齐，我们设$\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t,t) = -\frac{\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{x}_t,t)}{\bar{\beta}_t}$，此时式$(15)$为

$$\frac{1}{\bar{\beta}_t^2}\mathbb{E}_{\boldsymbol{x}_0\sim\tilde{p}(\boldsymbol{x}_0),\boldsymbol{\varepsilon}\sim\mathcal{N}(\boldsymbol{0},\boldsymbol{I})}\left[\left\Vert\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\bar{\alpha}_t\boldsymbol{x}_0 + \bar{\beta}_t\boldsymbol{\varepsilon}, t) - \boldsymbol{\varepsilon}\right\Vert^2\right] \tag{23}$$

忽略系数后就是DDPM的损失函数，而用$-\frac{\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{x}_{t+\Delta t}, t+\Delta t)}{\bar{\beta}_{t+\Delta t}}$替换掉式(9)的$\nabla_{\boldsymbol{x}_{t+\Delta t}}\log p(\boldsymbol{x}_{t+\Delta t})$后，结果与DDPM的采样过程具有相同的一阶近似（意味着$\Delta t\to 0$时两者等价）。

文章小结

本文主要介绍了宋飏博士建立的利用SDE理解扩散模型的一般框架，其中包括以尽可能直观的语言推导了反向SDE、得分匹配等结果，并对方程的求解给出了自己的想法。

如果您需要引用本文，请参考：

苏剑林. (Aug. 03, 2022). 《生成扩散模型漫谈（五）：一般框架之SDE篇 》[Blog post]. Retrieved from https://spaces.ac.cn/archives/9209

```
@online{kexuefm-9209,
        title={生成扩散模型漫谈（五）：一般框架之SDE篇},
        author={苏剑林},
        year={2022},
        month={Aug},
        url={\url{https://spaces.ac.cn/archives/9209}},
}
```