Diffusion models
Chapter1

矢口 真那斗

2024 年 6 月 24 日

目次

第 1 章　生成モデル 2
1.1　生成モデルとは何か 2
1.2　エネルギーベースモデル・分配関数 2
1.3　学習手法 3
1.4　スコア：対数尤度の入力についての勾配 5

第 1 章

生成モデル

1.1 生成モデルとは何か

生成モデルとは、変数 θ でパラメトライズされた確率分布 $q_\theta(\mathbf{x})$ であり、この分布に従ってサンプリングすることで、対象ドメインのデータを生成できるモデルである.
ここでは、生成モデルを訓練データから学習することを考える. 訓練データは iid である N 個のデータからなるデータセット $D = \{\mathbf{x}^{(1)}, \cdots, \mathbf{x}^{(N)}\}$ であり、未知の確率分布 $p(\mathbf{x})$ に従うとする. ただし、$\mathbf{x}^{(i)} \in \mathbb{R}^d$ は d 次元ベクトルである i 番目のデータ、$x_i \in \mathbb{R}$ は $\mathbf{x}$ の i 次元目の成分である.
生成モデルの学習の目標は、目標確率分布 $p(\mathbf{x})$ にできるだけ近い確率分布 $q_\theta(\mathbf{x})$ を見つけることである.

1.2 エネルギーベースモデル・分配関数

データ $\mathbf{x} \in X$ に対して、エネルギー関数 $f(\mathbf{x}; \theta) : \mathbb{R}^d \to \mathbb{R}$ を導入する. また生成モデルの確率分布 $q_\theta(\mathbf{x})$ を、エネルギー関数を用いて次のように定義する.

$$q_\theta(\mathbf{x}) = \frac{\exp(-f_\theta(\mathbf{x}))}{Z(\theta)} \tag{1.1}$$

$$Z(\theta) = \int_{\mathbf{x}' \in X} \exp(-f_\theta(\mathbf{x}'))d\mathbf{x}' \tag{1.2}$$

ここで、$Z(\theta)$ を分配関数とよぶ. このとき、エネルギー $f_\theta(\mathbf{x})$ が小さいほど、データ $\mathbf{x}$ が生成される確率が大きくなることを意味する. これは物理学におけるエネルギーの概念と対応しており、エネルギーベースモデルとよばれる.
エネルギーベースモデルの特徴として、構成性を挙げる. 2 つのエネルギー関数、$f_1(\mathbf{x}), f_2(\mathbf{x})$ と、対応する確率分布 $q_1(\mathbf{x}), q_2(\mathbf{x})$ を考える. このとき、エネルギー関

数の和 $f(\mathbf{x}) = f_1(\mathbf{x}) + f_2(\mathbf{x})$ に対応する確率分布 $q(\mathbf{x})$ は、次のようになる.

$$q(\mathbf{x}) \propto \exp(-f(\mathbf{x})) \tag{1.3}$$
$$= \exp(-f_1(\mathbf{x}) - f_2(\mathbf{x})) \tag{1.4}$$
$$= \exp(-f_1(\mathbf{x}))\exp(-f_2(\mathbf{x})) \tag{1.5}$$
$$= q_1(\mathbf{x})q_2(\mathbf{x}) \tag{1.6}$$

このように、エネルギー関数の和に対応する確率分布は、それぞれの確率分布の積になる. この性質は、生成モデルの構築において、複数のエネルギー関数を組み合わせる際に有用である.

1.3　学習手法

生成モデルの学習手法として、尤度ベースモデルとよばれる手法と、暗黙的生成モデルとよばれる手法の 2 つを紹介する.

1.3.1　尤度ベースモデル

あるデータ $\mathbf{x}$ の生成確率 $q_\theta(\mathbf{x})$ を尤度とよぶ. 訓練データセット $D = \{\mathbf{x}^{(1)}, \cdots, \mathbf{x}^{(N)}\}$ の尤度は、データが i.i.d であるから次のようになる.

$$q_\theta(D) = q_\theta(\mathbf{x}^{(1)}, \cdots, \mathbf{x}^{(N)}) = \prod_{i=1}^{N} q_\theta(\mathbf{x}^{(i)}) \tag{1.7}$$

対数尤度 $L(\theta)$ は、次のようになる.

$$L(\theta) = \log q_\theta(D) = \sum_{i=1}^{N} \log q_\theta(\mathbf{x}^{(i)}) \tag{1.8}$$

対数尤度を最大にするパラメータ $\theta^*_{ML} := \arg\max_\theta L(\theta)$ を求めるのが、最尤推定である.

エネルギーベースモデルの場合の対数尤度を考える. $q_\theta(\mathbf{x}) = \frac{\exp(-f_\theta(\mathbf{x}))}{Z(\theta)}$ より、対数尤度は次のようになる.

$$L(\theta) = \sum_{i=1}^{N} \log q_\theta(\mathbf{x}^{(i)}) \tag{1.9}$$
$$= \sum_{i=1}^{N} \log \frac{\exp(-f_\theta(\mathbf{x}^{(i)}))}{Z(\theta)} \tag{1.10}$$
$$= -\sum_{i=1}^{N} f_\theta(\mathbf{x}^i) - N \log Z(\theta) \tag{1.11}$$
$$= -\sum_{i=1}^{N} f_\theta(\mathbf{x}^i) - N \log \int_{\mathbf{x}' \in X} \exp(-f_\theta(\mathbf{x}'))d\mathbf{x}' \tag{1.12}$$

式 (1.12) より、対数尤度を最大にすることは、訓練データの位置のエネルギーを小さくし (第一項)、一方ですべての位置のエネルギーを大きくすることに対応する (第二項).
"$f_\theta(\mathbf{x}) = \exp(g_\theta(\mathbf{x}))$ と表せる場合を考えてみたい, 鋭くなる、記憶容量大、汎化性低?"
対数尤度を最大化するために、勾配法を用いることを考える.

$$\frac{\partial L(\theta)}{\partial \theta} = -\sum_{i=1}^{N} \frac{\partial f_\theta(\mathbf{x}^{(i)})}{\partial \theta} - N\frac{\partial}{\partial \theta}\log Z(\theta) \tag{1.13}$$

$$= -\sum_{i=1}^{N} \frac{\partial f_\theta(\mathbf{x}^{(i)})}{\partial \theta} - N\frac{1}{Z(\theta)}\int_{\mathbf{x}'\in X} -\exp(-f_\theta(\mathbf{x}'))\frac{\partial f_\theta(\mathbf{x}')}{\partial \theta} d\mathbf{x}' \tag{1.14}$$

$$= -\sum_{i=1}^{N} \frac{\partial f_\theta(\mathbf{x}^{(i)})}{\partial \theta} + N\int_{\mathbf{x}'\in X} q_\theta(\mathbf{x})\frac{\partial f_\theta(\mathbf{x}')}{\partial \theta} d\mathbf{x}' \tag{1.15}$$

$$= -\sum_{i=1}^{N} \frac{\partial f_\theta(\mathbf{x}^{(i)})}{\partial \theta} + N\mathbb{E}_{\mathbf{x}\sim q_\theta(\mathbf{x})}\left[\frac{\partial f_\theta(\mathbf{x})}{\partial \theta}\right] \tag{1.16}$$

ここで第二項は、生成モデルの確率分布に関して期待値を取ったものである。これをシュミレーションで求める際は、モンテカルロサンプリングを行うことが一般的であるが、計算コストが大きい.

尤度の最大化を KL ダイバージェンスの最小化の観点から捉える. 真の分布を $p(\mathbf{x})$, 生成モデルの分布を $q_\theta(\mathbf{x})$ とする. このとき、KL ダイバージェンスは次のように定義される.

$$D_{KL}(p(\mathbf{x})||q_\theta(\mathbf{x})) = \int_{\mathbf{x}\in X} p(\mathbf{x})\log\frac{p(\mathbf{x})}{q_\theta(\mathbf{x})} d\mathbf{x} \tag{1.17}$$

これを変形すると、

$$D_{KL}(p(\mathbf{x})||q_\theta(\mathbf{x})) = \int_{\mathbf{x}\in X} p(\mathbf{x})\log p(\mathbf{x}) d\mathbf{x} - \int_{\mathbf{x}\in X} p(\mathbf{x})\log q_\theta(\mathbf{x}) d\mathbf{x} \tag{1.18}$$

第 1 項は定数である. よって、KL ダイバージェンスの最小化は、第二項 $\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}[\log q_\theta(\mathbf{x})]$ の最大化と等価である.

1.3.2　暗黙的生成モデル

$q_\theta(\mathbf{x})$ と $p(\mathbf{x})$ の KL ダイバージェンスの最小化を考える.

$$\theta^* = \arg\min_\theta D_{KL}(q_\theta(x)||p(x)) \tag{1.19}$$

$$= \arg\min_\theta \mathbb{E}_{\mathbf{x}\sim q_\theta(\mathbf{x})}\left[\log q_\theta(\mathbf{x}) - \log p(\mathbf{x})\right] \tag{1.20}$$

$$= \arg\min_\theta \mathbb{E}_{\mathbf{x}\sim q_\theta(\mathbf{x})}\left[\log q_\theta(\mathbf{x})\right] - \mathbb{E}_{\mathbf{x}\sim q_\theta(\mathbf{x})}\left[\log p(\mathbf{x})\right] \tag{1.21}$$

$$= \arg\max_\theta -\mathbb{E}_{\mathbf{x}\sim q_\theta(\mathbf{x})}\left[\log q_\theta(\mathbf{x})\right] + \mathbb{E}_{\mathbf{x}\sim q_\theta(\mathbf{x})}\left[\log p(\mathbf{x})\right] \tag{1.22}$$

第 1 項はエントロピーの最大化、第二項は尤度の最大化に対応している。

1.4　スコア：対数尤度の入力についての勾配

一般に高次元の確率モデルの分配関数の計算は難しい。また、分配関数の計算が必要ない MCMC 法を持ち知恵も、尤度が高い領域に効率的に到達することは難しい。
任意の入力について微分可能な確率分布上で定義できる関数、スコア関数を以下で定義する.

$$\mathbf{s}(\mathbf{x}) := \nabla_{\mathbf{x}} \log p(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}^d \tag{1.23}$$

生成モデルの確率分布 $q_\theta(\mathbf{x})$ のスコアは、次のようになる.

$$\nabla_{\mathbf{x}} \log q_\theta(\mathbf{x}) = -\nabla_{\mathbf{x}} f_\theta(\mathbf{x}) - \nabla_{\mathbf{x}} \log Z(\theta) \tag{1.24}$$
$$= -\nabla_{\mathbf{x}} f_\theta(\mathbf{x}) \tag{1.25}$$

分配関数の計算が不要であるため、学習効率が良いという利点がある。

1.4.1　ランジュバン・モンテカルロ法

スコアを用いたサンプリング方法として、ランジュバン・モンテカルロ法がある. この方法は、スコアを用いて確率分布 $p(\mathbf{x})$ に従うサンプルを生成する手法である.
ランジュバン・モンテカルロ法は、次の手順でサンプリングを行う.

- 任意の分布 $\pi(\mathbf{x})$ からデータを $\mathbf{x}_0 \sim \pi(\mathbf{x})$ とサンプリングする
- $\mathbf{x}_0$ におけるスコアを計算し、遷移する. この時、正規分布からサンプリングされたノイズを少し加える.
- この遷移を K 回繰り返した結果を、サンプリング結果とする.

1.4.2　スコアマッチング

確率分布のスコアが得られれば、ランジュバン・モンテカルロ法を使って、その確率分布から効率的にサンプリングできる. 確率分布を直接学習する代わりに確率分布のスコアを学習し、スコアを使って生成モデルを実現する手法をスコアベースモデルとよぶ.
次にスコアをニューラルネットワークなどで表したパラメータ θ で特徴づけられたモデル $\mathbf{s}_\theta(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}^d$ を使って学習することを考える.
まず、学習対象のスコアとモデルの出力間の 2 乗誤差が最小となるようなパラメータを求めることを考える. つまり目的関数を以下で定義する.

$$J_{ESM_p}(\theta) = \frac{1}{2} \mathbb{E}_{p(\mathbf{x})} [\| \nabla_{\mathbf{x}} \log p(\mathbf{x}) - \mathbf{s}_\theta(\mathbf{x}) \|^2] \tag{1.26}$$

この目的関数は明示的スコアマッチング (ESM) と呼ばれる. しかし、一般に、学習においては訓練データ $D = \{\mathbf{x}^{(1)}, \cdots, \mathbf{x}^{(N)}\}$ のみが与えられ、スコアは未知である。そのため、

このアプローチは現実的ではない.

1.4.3　暗黙的スコアマッチング

訓練データのみからスコアを学習する手法として、暗黙的スコアマッチング (ISM) と呼ばれる目的関数を以下で定義する.

$$J_{ISM_p}(\theta) = \mathbb{E}_{p(\mathbf{x})}[\frac{1}{2}\|\mathbf{s}_\theta(\mathbf{x})\|^2 + tr(\nabla_{\mathbf{x}}\mathbf{s}_\theta(\mathbf{x}))] \tag{1.27}$$

ここで、第 2 項のトレースは、以下のように表される.

$$tr(\nabla_{\mathbf{x}}\mathbf{s}_\theta(\mathbf{x})) = \sum_{i=1}^{d}\frac{\partial \mathbf{s}_\theta(\mathbf{x})_i}{\partial x_i} = -\sum_{i=1}^{d}\frac{\partial^2 f_\theta(\mathbf{x})}{\partial x_i^2} \tag{1.28}$$

この時、暗黙的スコアマッチングと、明示的スコアマッチングについて定数項 C_1 を用いて以下で表される.

$$J_{ESM_p}(\theta)\ J_{ISM_p}(\theta) + C_1 \tag{1.29}$$

1.4.4　暗黙的スコアマッチングがスコアを推定できることの証明

$$J_{ESM}(\theta) = \frac{1}{2}\mathbb{E}_{p(\mathbf{x})}[\|\nabla_{\mathbf{x}}\mathrm{log}p(\mathbf{x}) - \mathbf{s}_\theta(\mathbf{x})\|^2] \tag{1.30}$$

$$J_{ISM}(\theta) = \mathbb{E}_{p(\mathbf{x})}[\frac{1}{2}\|\mathbf{s}_\theta(\mathbf{x})\|^2 + tr(\nabla_{\mathbf{x}}\mathbf{s}_\theta(\mathbf{x}))] \tag{1.31}$$

以下の 4 つの仮定をおく.

- $p(\mathbf{x})$ が微分可能
- $\mathbb{E}_{p(\mathbf{x})}[\|\nabla_{\mathbf{x}}\mathrm{log}p(\mathbf{x})\|^2]$ が有限
- 任意の θ について $\mathbb{E}_{p(\mathbf{x})}[\|\mathbf{s}_\theta(\mathbf{x})\|^2]$ が有限
- $lim_{\|\mathbf{x}\|\to\infty}[p(\mathbf{x})\mathbf{s}_\theta(\mathbf{x})] = 0$

この時、定数 C_1 を用いて、以下の式が成り立つことを示す.

$$J_{ESM}(\theta) = J_{ISM}(\theta) + C_1 \tag{1.32}$$

$$J_{ESM}(\theta) = \frac{1}{2}\mathbb{E}_{p(\mathbf{x})}[\|\nabla_{\mathbf{x}}\mathrm{log}p(\mathbf{x}) - \mathbf{s}_\theta(\mathbf{x})\|^2] \tag{1.33}$$

$$= \int_{\mathbf{x}\in\mathbb{R}^d} p(\mathbf{x})\left[\frac{1}{2}\|\nabla_{\mathbf{x}}\mathrm{log}p(\mathbf{x})\|^2 + \frac{1}{2}\|\mathbf{s}_\theta(\mathbf{x})\|^2 - \nabla_{\mathbf{x}}\mathrm{log}p(\mathbf{x})^T\mathbf{s}_\theta(\mathbf{x})\right]d\mathbf{x} \tag{1.34}$$

$$\tag{1.35}$$

第 3 項について、

$$-\int_{\mathbf{x}\in\mathbb{R}^d} p(\mathbf{x})\nabla_{\mathbf{x}}\log p(\mathbf{x})^T \mathbf{s}_\theta(\mathbf{x})d\mathbf{x} = -\sum_i \int_{\mathbf{x}\in\mathbb{R}^d} p(\mathbf{x})(\nabla_{\mathbf{x}}\log p(\mathbf{x}))_i \mathbf{s}_\theta(\mathbf{x})_i d\mathbf{x} \quad (1.36)$$

$$= -\sum_i \int_{\mathbf{x}\in\mathbb{R}^d} p(\mathbf{x})\frac{\partial \log p(\mathbf{x})}{\partial x_i}\mathbf{s}_\theta(\mathbf{x})_i d\mathbf{x} \quad (1.37)$$

$$= -\sum_i \int_{\mathbf{x}\in\mathbb{R}^d} \frac{\partial p(\mathbf{x})}{\partial x_i}\mathbf{s}_\theta(\mathbf{x})_i d\mathbf{x} \quad (1.38)$$

ここで、成分 i ごとに、以下が成り立つことを用いる.

$$-\int_{\mathbf{x}\in\mathbb{R}^d} \frac{\partial p(\mathbf{x})}{\partial x_i}\mathbf{s}_\theta(\mathbf{x})_i d\mathbf{x} = \int_{\mathbf{x}\in\mathbb{R}^d} \frac{\partial \mathbf{s}_\theta(\mathbf{x})_i}{\partial x_i} p(\mathbf{x})\mathbf{x} \quad (1.39)$$

これを用いると、

$$-\sum_i \int_{\mathbf{x}\in\mathbb{R}^d} \frac{\partial p(\mathbf{x})}{\partial x_i}\mathbf{s}_\theta(\mathbf{x})_i d\mathbf{x} = \sum_i \int_{\mathbf{x}\in\mathbb{R}^d} \frac{\partial \mathbf{s}_\theta(\mathbf{x})_i}{\partial x_i} p(\mathbf{x})\mathbf{x} \quad (1.40)$$

$$= \mathbb{E}_{p(\mathbf{x})}[tr(\nabla_{\mathbf{x}}\mathbf{s}_\theta(\mathbf{x})] \quad (1.41)$$

よって題意は示された.

1.4.5　デノイジングスコアマッチング

暗黙的スコアマッチングを用いることで、目標分布のスコアが未知でも学習ができる. 一方で暗黙的スコアマッチングの欠点が 2 つ存在する. 一つは、$\mathbb{E}_{p(\mathbf{x})}[tr(\nabla_{\mathbf{x}}\mathbf{s}_\theta(\mathbf{x}))]$ を求める計算量が大きいことである. もう一つは、過学習が起こりやすいことである. 有限のデータから暗黙的スコアマッチングにより学習すると、訓練データの位置で対数尤度の 1 次微分が 0, 2 次微分が負の無限大を取るのが最適である. この時、各訓練データの位置で確率が正の無限大となるようなディラックのデルタ関数の混合分布が最適な分布となる. このため、過学習を防ぐ正則化を加える必要がある.

ここではデノイジングスコアマッチングを使ってこれら 2 つの問題が解決されることを見る.

データにノイズを加えたデータ $\tilde{\mathbf{x}}$ を考える. この過程は、以下の式で表される.

$$p_\sigma(\tilde{\mathbf{x}}|\mathbf{x}) = N(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2 I) = \frac{1}{(2\pi\sigma^2)^{d/2}}\exp\left(-\frac{1}{2\sigma^2}\|\tilde{\mathbf{x}} - \mathbf{x}\|^2\right) \quad (1.42)$$

この時、データ分布 $p(\mathbf{x})$ の各点に摂動が加わった後に得られる摂動後分布は、次のようになる.

$$p_\sigma(\tilde{\mathbf{x}}) = \int_{\mathbf{x}\in\mathbb{R}^d} p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})p(\mathbf{x})d\mathbf{x} \quad (1.43)$$

この時、摂動後分布での明示的スコアマッチング、および暗黙的スコアマッチングの目的関数は次のようになる.

$$J_{ESM_{p_\sigma}}(\theta) = \frac{1}{2}\mathbb{E}_{p_\sigma(\tilde{\mathbf{x}})}[\|\nabla_{\tilde{\mathbf{x}}}\log p_\sigma(\tilde{\mathbf{x}}) - \mathbf{s}_\theta(\tilde{\mathbf{x}},\sigma)\|^2] \tag{1.44}$$

$$J_{ISM_{p_\sigma}}(\theta) = \mathbb{E}_{p_\sigma(\tilde{\mathbf{x}})}[\frac{1}{2}\|\mathbf{s}_\theta(\tilde{\mathbf{x}},\sigma)\|^2 + tr(\nabla_{\tilde{\mathbf{x}}}\mathbf{s}_\theta(\tilde{\mathbf{x}},\sigma))] \tag{1.45}$$

これらの目的関数を最小化することで、過学習の問題は解決できる. 一方で計算量の問題は解決していない.
デノイジングスコアマッチングでは、直接スコアを目標に学習するのではなく、摂動時の条件付き確率に関するスコアを目標に学習する.

$$J_{DSM_{p_\sigma}}(\theta) = \frac{1}{2}\mathbb{E}_{p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})p(\mathbf{x})}[\|\nabla_{\tilde{\mathbf{x}}}\log p_\sigma(\tilde{\mathbf{x}}|\mathbf{x}) - \mathbf{s}_\theta(\tilde{\mathbf{x}},\sigma)\|^2] \tag{1.46}$$

ここで、条件付き確率のスコアは次のように解析的に求めることができる.

$$\nabla_{\tilde{\mathbf{x}}}\log p_\sigma(\tilde{\mathbf{x}}|\mathbf{x}) = \nabla_{\tilde{\mathbf{x}}}\log(\frac{1}{(2\pi)^{d/2}\sigma^d})exp(-\frac{1}{2\sigma^2}\|\tilde{\mathbf{x}}-\mathbf{x}\|^2) \tag{1.47}$$

$$= \nabla_{\tilde{\mathbf{x}}}\log\frac{1}{(2\pi)^{d/2}\sigma^d} + \nabla_{\tilde{\mathbf{x}}}(-\frac{1}{2\sigma^2}\|\tilde{\mathbf{x}}-\mathbf{x}\|^2) \tag{1.48}$$

$$= 0 - \frac{1}{\sigma^2}(\tilde{\mathbf{x}}-\mathbf{x}) \tag{1.49}$$

$$= -\frac{1}{\sigma^2}\epsilon \tag{1.50}$$

したがって、デノイジングスコアマッチング関数を書き直すと、

$$J_{DSM_{p_\sigma}}(\theta) = \frac{1}{2}\mathbb{E}_{\epsilon\sim N(0,\sigma^2 I),\mathbf{x}\sim p(\mathbf{x})}[\|-\frac{1}{\sigma^2}\epsilon - \mathbf{s}_\theta(\mathbf{x}+\epsilon,\sigma)\|^2] \tag{1.51}$$

このデノイジングスコアマッチングは、摂動後分布での明示的スコアマッチングと、定数 C を用いて次の関係で表される.

$$J_{ESM_{p_\sigma}}(\theta) = J_{DSM_{p_\sigma}}(\theta) + C \tag{1.52}$$

1.4.6　デノイジングスコアマッチングがスコアを推定できることの証明

デノイジングスコアマッチングが明示的スコアマッチングと定数を除いて一致することを証明する.

$$J_{ESM_{p_\sigma}}(\theta) = \frac{1}{2}\mathbb{E}_{p_\sigma(\tilde{\mathbf{x}})}[\|\nabla_{\tilde{\mathbf{x}}}\log p_\sigma(\tilde{\mathbf{x}}) - \mathbf{s}_\theta(\tilde{\mathbf{x}},\sigma)\|^2] \tag{1.53}$$

$$= \frac{1}{2}\mathbb{E}_{p_\sigma(\tilde{\mathbf{x}})}[\|\mathbf{s}_\theta(\tilde{\mathbf{x}},\sigma)\|^2] - \mathbb{E}_{p_\sigma(\tilde{\mathbf{x}})}[< \mathbf{s}_\theta(\tilde{\mathbf{x}},\sigma), \nabla_{\tilde{\mathbf{x}}} log p_\sigma(\tilde{\mathbf{x}}) >] + C_2 \tag{1.54}$$

$$= \frac{1}{2}\mathbb{E}_{p_\sigma(\tilde{\mathbf{x}})}[\|\mathbf{s}_\theta(\tilde{\mathbf{x}},\sigma)\|^2] - S(\theta) + C_2 \tag{1.55}$$

一方、

$$J_{DSM_{p_\sigma}}(\theta) = \frac{1}{2}\mathbb{E}_{p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})p(\mathbf{x})}[\|\nabla_{\tilde{\mathbf{x}}} log p_\sigma(\tilde{\mathbf{x}}) - \mathbf{s}_\theta(\tilde{\mathbf{x}}, \sigma)\|^2] \tag{1.56}$$

$$= \frac{1}{2}\mathbb{E}_{p_\sigma(\tilde{\mathbf{x}})}[\|\mathbf{s}_\theta(\tilde{\mathbf{x}}, \sigma)\|^2] - \mathbb{E}_{p_\sigma(\tilde{\mathbf{x}},\mathbf{x})}[< \mathbf{s}_\theta(\tilde{\mathbf{x}}, \sigma), \nabla_{\tilde{\mathbf{x}}} log p_\sigma(\tilde{\mathbf{x}}|\mathbf{x}) >] + C_3 \tag{1.57}$$

よって、第 2 項と $S(\theta)$ が等しいことを示せば良い. (P.27 参照)