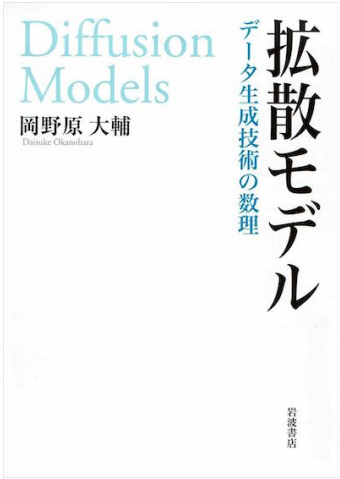
Preferred Networks

Workshop OT 2023 2023/03/17

拡散モデルとその周辺



Preferred Networks 岡野原 大輔
@hillbig

本講演中の図の一部は岩波書店「拡散モデル」（岡野原 著）から引用しています

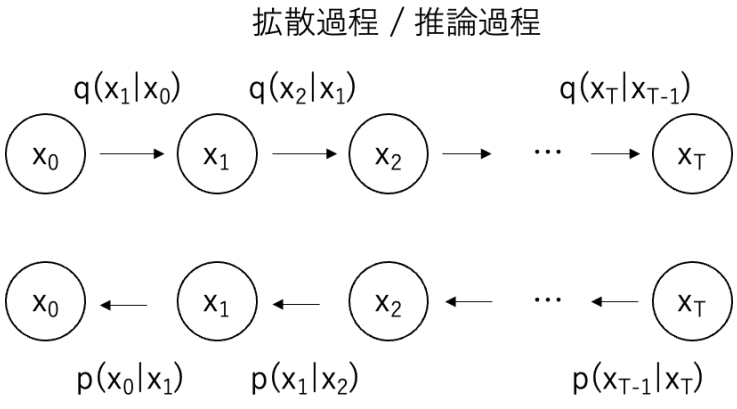
アジェンダ

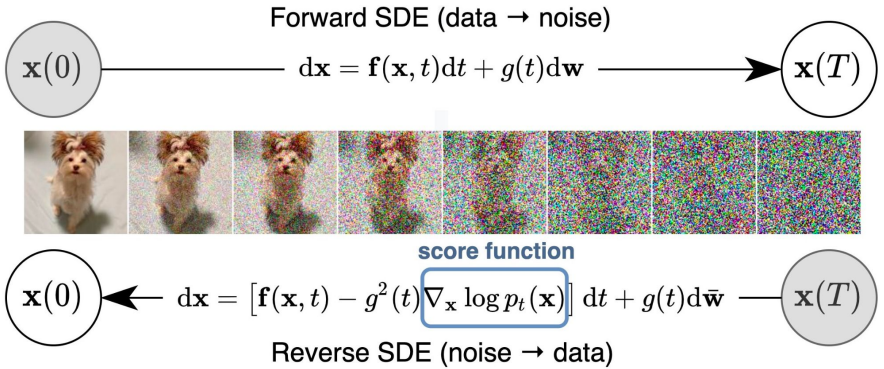
- 拡散モデルとは
- デノイジングスコアマッチング
- 確率微分方程式 - 確率ODEフロー
- フローマッチング
- 今後の展望

拡散モデルとは

拡散モデル [SohI-Dickstein+ 2015] [Song+ 2019] [Ho+ 2020]

- 非平衡熱力学を源流に持つ、深層生成モデルの一種
- データにノイズを徐々に加えていく拡散過程を逆向きに辿る逆拡散過程（生成過程）によって生成モデルを定義する
- データを破壊することで生成方法を学習する





https://yang-song.net/blog/2021/score/

拡散モデルの広がり

- 多くのタスクで従来性能を凌駕する性能を達成
  - 画像、音声、点群、化合物、動画
  - 編集：補間、超解像、Zero-shot編集
  - 密度推定、非可逆圧縮、敵対的摂動頑健性向上、最適化
- テキスト条件付画像生成では既に1億枚超の画像が生成されている

テキスト条件付動画生成の例
[Ho+ 2022]

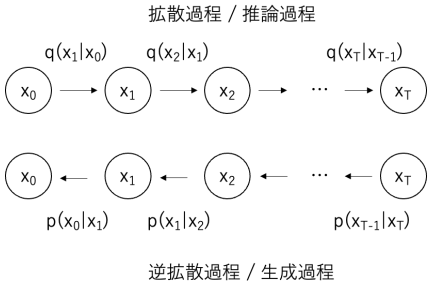
midjourney v5の出力例 Yuki Homma @y__homm https://twitter.com/y__homm/status/1636186478899494912

拡散モデルは複数の確率層からなるVAE

- 拡散過程（固定の推論）
- 逆拡散過程（生成）
- データの対数尤度の変分下限（ELBO）最大化で学習

$$q(\mathbf{x}_{1:T}|\mathbf{x}_0) := \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})$$

$$p_\theta(\mathbf{x}_{0:T}) := p(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$$



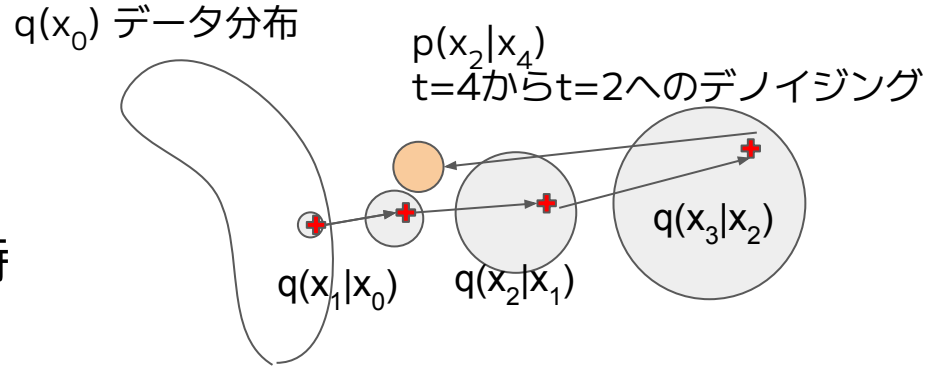
$$\log p_\theta(\mathbf{x}_0) = \log \int p(\mathbf{x}_{0:T}) d\mathbf{x}_{1:T}$$

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

データを拡散して破壊した時に、元に復元できる経路を求める

物理の言葉でいえば、事前分布から目標分布へ変換する経路の中で発生する散逸（自由エネルギー減少）が最小の経路を求める

デノイジング



拡散過程が次のように与えられた時

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\alpha_t \mathbf{x}_0, \sigma_t^2 \mathbf{I})$$

最適な逆拡散過程の1ステップ(s<t)は次のように求められる

$$p(\mathbf{x}_s|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_s; \mu(\mathbf{x}_t, s, t), \tilde{\sigma}^2(s,t)\mathbf{I})$$

$$\mu(\mathbf{x}_t, s, t)$$

$$= \frac{\alpha_{t|s}\sigma_s^2}{\sigma_t^2}\mathbf{x}_t + \frac{\alpha_s \sigma_{t|s}^2}{\sigma_t^2}\hat{\mathbf{x}}_\theta(\mathbf{x}_t; t)$$

今のデータに推定した**デノイジング結果**を足す

または

$$= \frac{1}{\alpha_{t|s}}\mathbf{x}_t - \frac{\sigma_{t|s}^2}{\alpha_{t|s}\sigma_t}\hat{\epsilon}_\theta(\mathbf{x}_t; t)$$

今のデータから推定した**ノイズ**を引く

または

$$= \frac{1}{\alpha_{t|s}}\mathbf{x}_t + \frac{\sigma_{t|s}^2}{\alpha_{t|s}}\mathbf{s}_\theta(\mathbf{x}_t; t)$$

今のデータを推定した**スコア**に従って遷移する

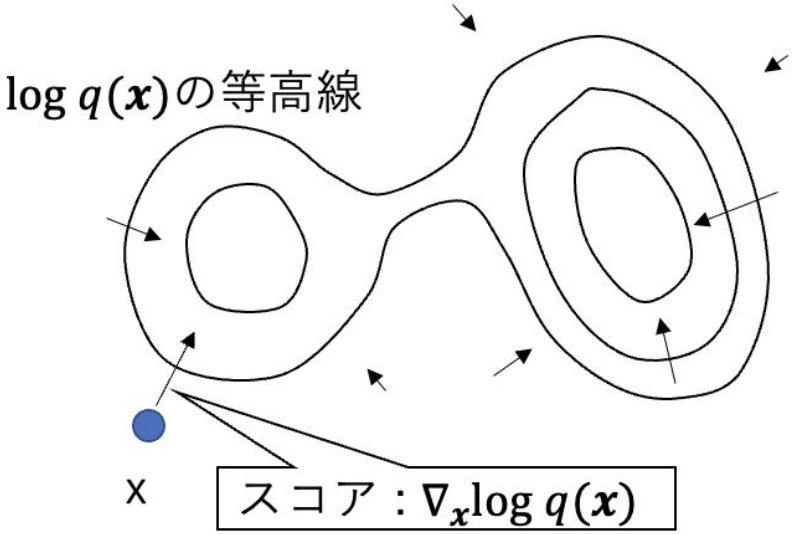
スコア* = 対数尤度の勾配 = エネルギーの負の勾配

*注意：情報幾何などの文脈ではパラメータについての勾配だがここでは入力についての勾配

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

$$q_\theta(\mathbf{x}) = \exp(-f_\theta(\mathbf{x}))/Z(\theta)$$

$$\begin{aligned}
&\nabla_{\mathbf{x}} \log q_\theta(\mathbf{x}) \\
&= -\nabla_{\mathbf{x}} f_\theta(\mathbf{x}) - \underbrace{\nabla_{\mathbf{x}} \log Z(\theta)}_{=0} \\
&= -\nabla_{\mathbf{x}} f_\theta(\mathbf{x})
\end{aligned}$$



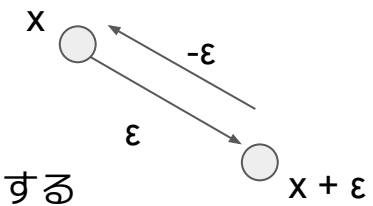
スコアには分配関数が現れず、局所的な情報だけで決定される

→ モデル化（学習）しやすい、合成しやすい、対称性を導入しやすい

デノイジングスコアマッチング [Vincent+ 2011]

以降、簡略化のため拡散過程の時刻tの代わりに撹乱強度σを使う

$$p_\sigma(\tilde{\mathbf{x}}|\mathbf{x}) = \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2\mathbf{I})$$



デノイジングスコアマッチング目的関数を考える

(3) 加えたノイズを予測する

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

(1) ノイズを用意

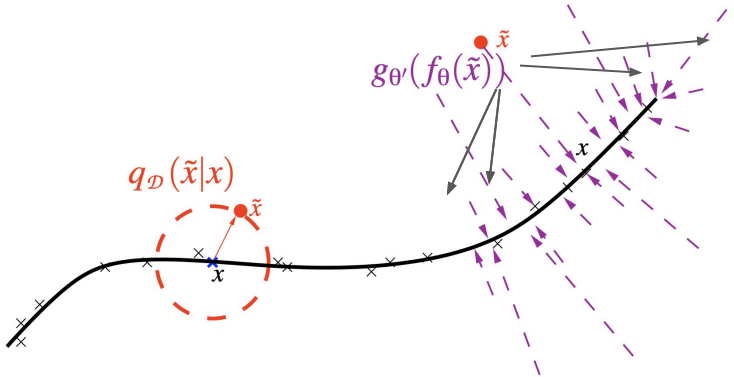
(2) データにノイズを載せる

この最適化問題の最適解$s^*_\theta(x, \sigma)$はスコア$\nabla \log p_\sigma(x)$と一致する

デノイジングスコアマッチングの直感的な意味

ノイズを加えると確率が低い領域へ飛び出す。様々な方向へのデノイジングの平均は確率が高い方向への垂線となる

フォッカー・プランク方程式中にスコアがでてくるのと仕組みは同じであり、エントロピーの変化率からスコアがでてくる



いろいろ方向へ向けてのデノイジングは打ち消し合い垂直方向のみが残る

図は[Vincent 2010]

拡散モデルを使った学習と推論（生成）

学習

データに様々な強度のノイズをのせ、デノイジングできるよう学習する

推論（生成）

完全なノイズからデータをサンプリングし、それをデノイジング強度を下げながらデノイジングするのを繰り返す

局所解に陥らないよう各強度の撹乱後分布でランジュバンモンテカルロを使ってサンプリングするスコアベースドモデル [Song+ 2019]と拡散モデルは目的関数の係数などを除いて一致する [Ho+ 2020]

確率微分方程式での定式化 [Song+ 201]

ノイズを加える過程のステップ数を無限化した場合の拡散過程は
次の確率微分方程式（ランジュバン方程式）で表される

$$\mathrm{d}\mathbf{x} = f(t)\mathbf{x}\mathrm{d}t + g(t)\mathrm{d}\mathbf{w}$$

ブラウン運動

この逆拡散過程の確率微分方程式は次の通り与えられる [Anderson 1982]

$$\mathrm{d}\mathbf{x} = [f(t)\mathbf{x} - g(t)^2\nabla_{\mathbf{x}}\log p_t(\mathbf{x})]\mathrm{d}t + g(t)\mathrm{d}\bar{\mathbf{w}}$$

各時刻のスコアさえ求まれば、この方程式に従ってデータを生成できる

前向き確率微分方程式

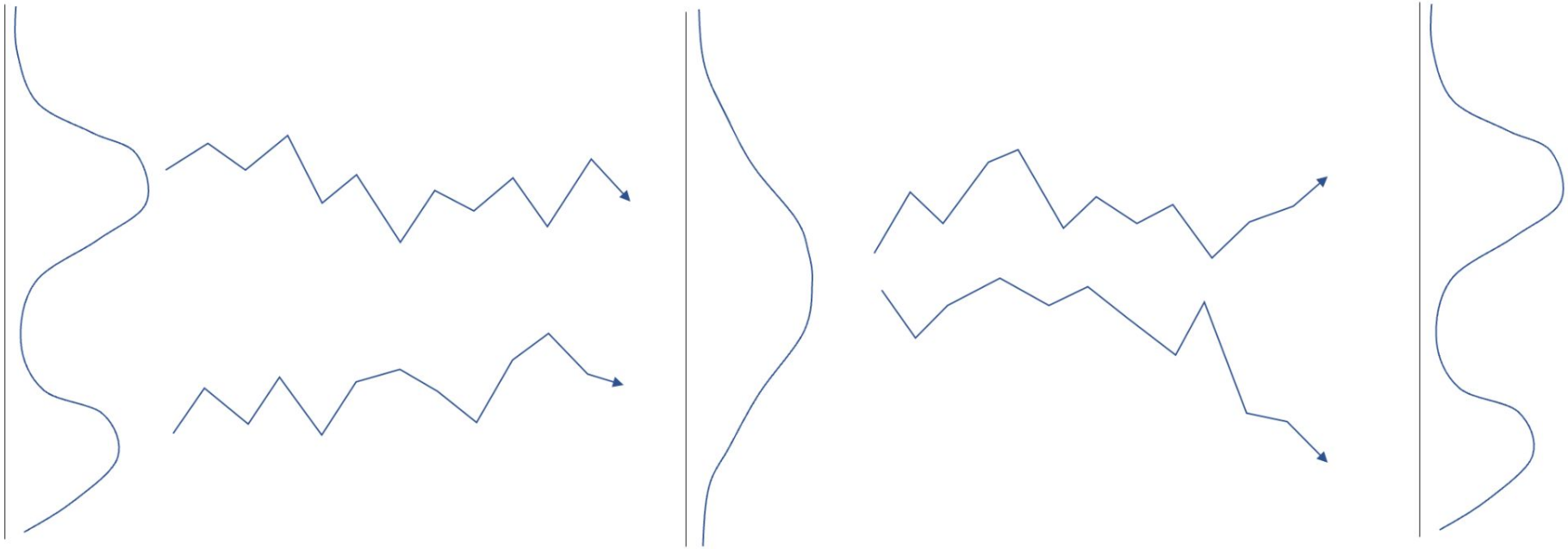
逆向き確率微分方程式

[SDE]

x(0) → x(T) → x(0)

$$dx = f(x,t)dt + g(t)dw$$

$$dx = [f(x,t) - g^2(t)\nabla_x \log p_t(x)]\,dt + g(t)d\bar{w}$$



前向き確率微分方程式でデータ分布は破壊され事前分布に変換され、
逆向き確率微分方程式で、事前分布からデータ分布に変換する。

確率フローODE：常微分方程式での定式化

先程と同じ確率分布を与えるノイズを含まない常微分方程式（Neural ODE）は次のように与えられる [Song+ 2021]

$$\mathrm{d}\mathbf{x} = [\mathbf{f}(\mathbf{x}, t) - \tfrac{1}{2} g^2(t) \nabla_{\mathbf{x}} \log p_t(\mathbf{x})]\mathrm{d}t$$

事前分布とデータ分布間のデータの1対1対応を与える
実用上は次の拡散係数σ(t)のみに依存する式がよく使われる[Karras+ 2022]

$$\mathrm{d}\mathbf{x} = -\sigma'(t)\sigma(t) \nabla_{\mathbf{x}} \log p(\mathbf{x}; \sigma(t))\mathrm{d}t$$

前向き常微分方程式

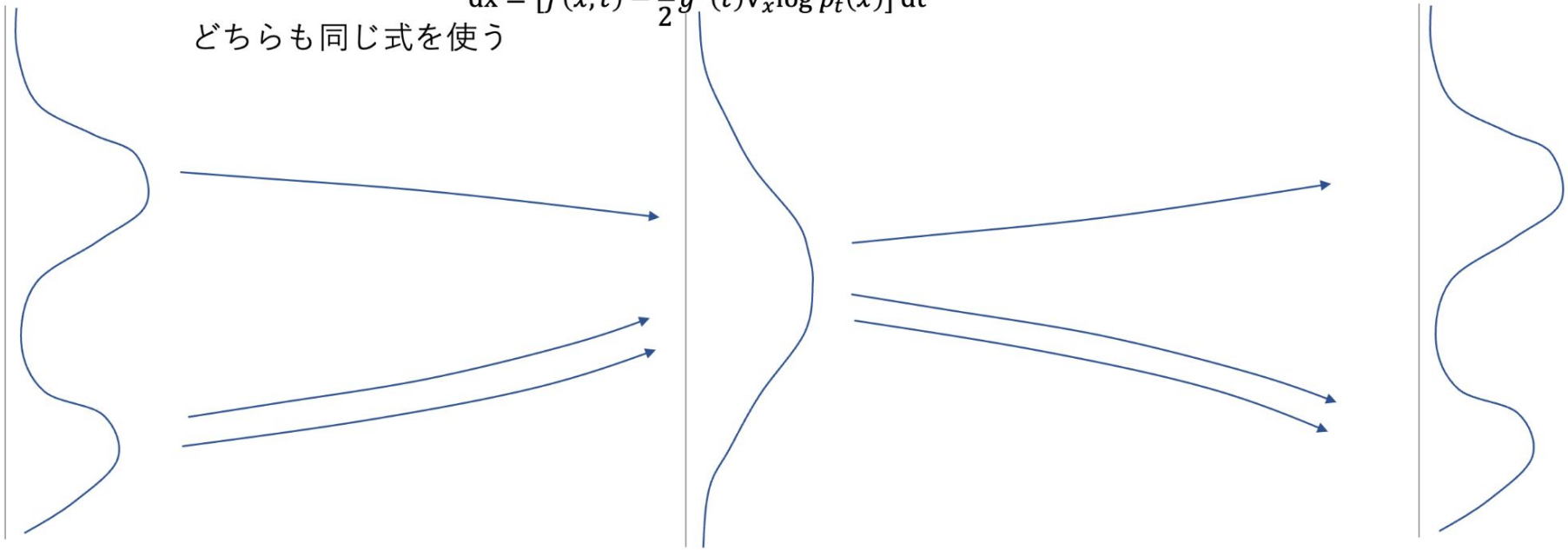
逆向き常微分方程式

[ODE]

x(0) → x(T) → x(0)

$$\mathrm{dx} = [f(x,t) - \frac{1}{2} g^2(t) \nabla_x \log p_t(x)]\,\mathrm{dt}$$

どちらも同じ式を使う



確率微分方程式は常微分方程式で表すことができ、データから事前分布中の点への変換は可逆変換で表すことができる。

なぜ拡散モデルが優れているか (1/4)
一つの最適化問題で安定して学習できる

- VAE ：生成モデルと認識モデルを同時に学習する必要
  - 認識モデルの学習が一般に難しい

- GAN：二つのネットワークを競合して学習する
  - うまく競合させる必要があり、失敗することも多い
  - 逆KL最適化でモード崩壊が起きやすい

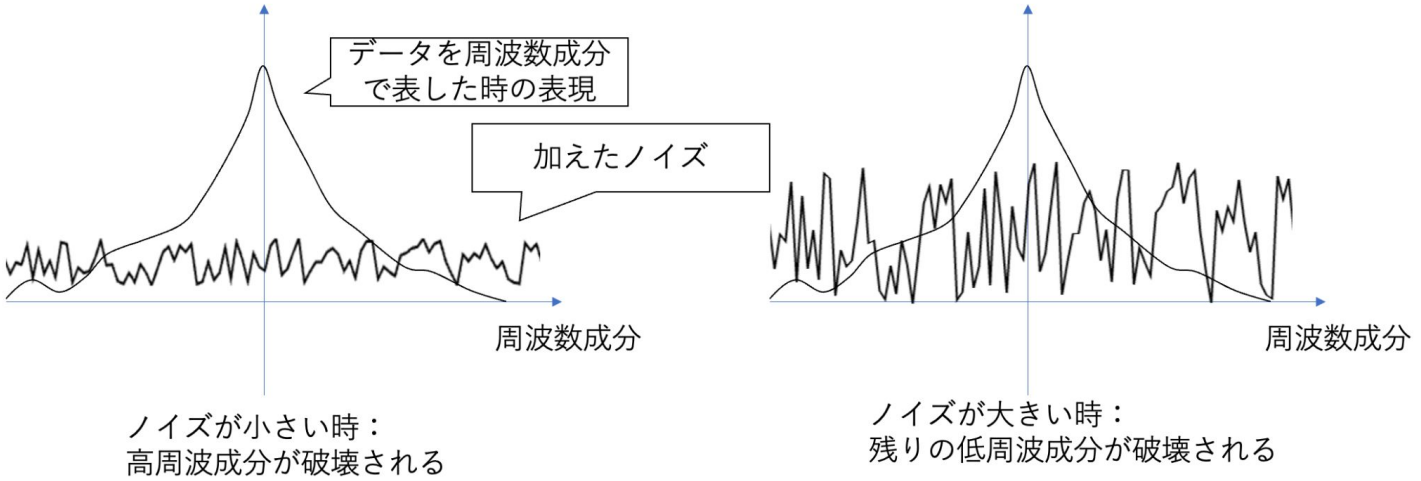
拡散モデルは固定で事後分布崩壊が起きない認識モデル（拡散過程）を使って生成モデルのみ学習するVAE

安定的に学習できる → 大きなデータで長時間学習できる

なぜ拡散モデルが優れているか (2/4)

複雑な生成過程を簡単な部分生成問題に自動的に分解する

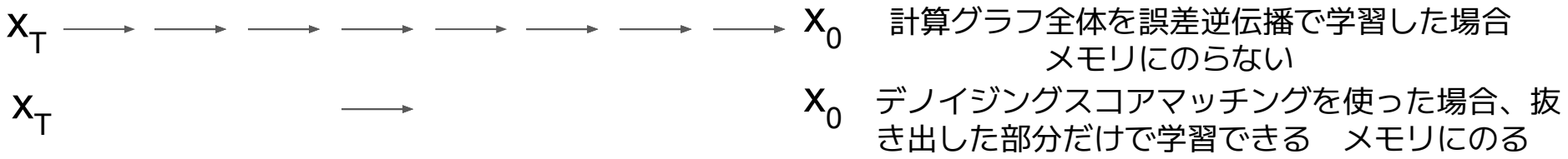
- 拡散過程はノイズを徐々に強くしていくため、データの詳細な部分から全体に向けて徐々に破壊する
- 生成は逆に全体構造から詳細に向けて生成する



なぜ拡散モデルが優れているか (3/4)

非常に長い生成過程を各ステップ独立に学習できる

- 拡散モデルの生成は多くの確率層を使った計算グラフで表される
  - 生成過程が1000ステップの拡散モデルは1000層の確率層からなる生成モデルとみなせる
  - 各デノイジングは巨大なNNを使って実現する
- 計算グラフ全体がメモリに乗らないため誤差逆伝搬法で学習することはメモリ容量的に不可能
- デノイジングスコアマッチングは、計算グラフの途中の一部分を抜き出し、そこだけ学習できる。巨大な計算過程を学習可能に



なぜ拡散モデルが優れているか (4/4)
摂動前後のデータの局所的な変化をモデル化

- 分配関数を直接扱う必要がなく、入力空間から入力空間へ変換できる関数だけモデル化すれば良い
  - 他全体は気にせず局所だけモデル化すればよい

- 逆拡散は拡散と同じ関数形で近似できる [Feller 1949]

- 生成時の入力操作に対する対称性を簡単に導入できる
  - 事前分布が操作に対し不変、変換が操作に対し同変な関数を用いた場合、生成確率は操作に対し不変
  - 例：自己回転、並進移動（SE(3)）操作に対し生成確率は不変なモデル [Xu+ 2022]

拡散モデルの条件付生成 [Ho+ 2022]

$$\| - \frac{1}{\sigma^2}\epsilon - s_\theta(\mathbf{x} + \epsilon, c, \sigma)\|^2$$

- 条件cを入れたデノイジングスコアマッチングで推定
  - 条件無しもcに特別な値（0ベクトル等）で学習
- 生成時は条件付スコアを使って生成
- 複数の条件付確率が各条件付確率の積で表される場合、そのスコアは和の形で分解され、学習しやすい

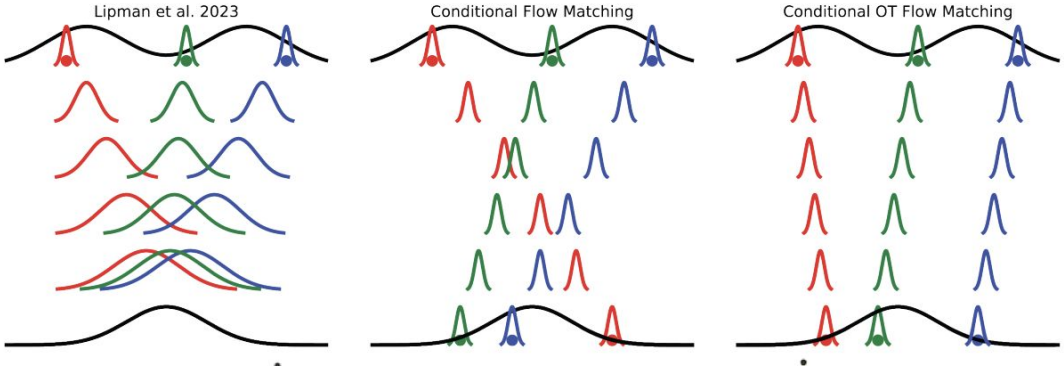
$$p(\mathbf{x}|c_1, c_2) \propto p(\mathbf{x}|c_1)p(x|c_2)$$

$$s(\mathbf{x}|c_1, c_2) = s(\mathbf{x}|c_1) + s(\mathbf{x}|c_2)$$

テキスト条件づけなど複雑な条件付学習も内部はこのような分解がおきていると考えられる

フローマッチング [Lipman+ ICLR 2023] [Tong+ arXiv]

- データ毎のデータ点から事前分布への最適輸送をデータ分布で”周辺化”することでデータ分布から事前分布への最適輸送を求める
  デノイジングスコアマッチングとほぼ同様の証明を行う
- 拡散モデルと同じ枠組みにのるが、フローマッチングが実用上、拡散モデルより優れている部分も多い



今後の展望

今後の展望 (1/3)

- より優れた拡散過程 or 認識過程
  - 拡散モデルは実際のデータの生成過程は推定していない
    - 例えばStyleGANが獲得したような視点変化は得られない
    - 条件付によって生成分布を分解することのみできる
  - 拡散過程が生成過程を定義し、学習効率/汎化を決めている
    - 適当な変換も拡散過程で使える [Bansal+ 2022]
  - 生成過程は別の仕組みで推定し、潜在変数上の分布のみモデル化
    - (潜在空間/部分空間/自己符号化器)拡散モデル
- ノイズスケジュールの最適化
  - データ生成過程の複雑度にあわせて最適化する必要があり、実はとても重要

今後の展望 (2/3)

- 非平衡熱力学や物理との接点を明らかにする
  - 拡散モデルのコミュニティに未発見の様々な知見が既に非平衡熱力学や物理の世界で得られている [伊藤 私信]
- 拡散モデルが今後成功しそうな分野問題への適用
  - 生成過程や対称性が分かっているドメイン
    - 拡散過程や対称性などで強い事前知識を入れる
  - 最適化
    - 多峰性がある最適化問題での汎用最適化ヒューリスティックスを獲得できる可能性
  - 離散データ / 言語モデル
    - 現在のTransformer（非）自己回帰モデルは離散変数のデノイジングの一種とみなすことができる [Lezama+ 2023]

今後の展望 (3/3)

- 脳の学習との接点を明らかにする
  - 拡散モデルの学習（デノイジングスコアマッチング）、推論は脳内の計算機構で実現可能性が高い
    - 各ニューロンの発火は離散化ノイズが加わる。その上で周囲も使って自分の正確な値（活性値）を予測する
      → デノイジングスコアマッチング目的関数が実現
  - 任意の制約、条件の上で推論を自由自在にできる
    - エネルギーベースドモデルと同じことができる

→ 脳の計算処理はデノイジング/スコア推定ではないかという仮説

- 実現可能性で残されたギャップ：
  - 1ステップのデノイジングを誤差逆伝搬法無しで学習できるか？

文献

[Sohl-Dickstein+ 2015] “Deep Unsupervised learning using nonequilibrium thermodynamics”, ICML 2015
[Song+ 2019] “Generative Modeling by Estimating Gradients of the Data Distribution”, NeurIPS 2019
[Ho+ 2020] “Denosing Diffusion Probabilistic Mdoels”, NeurIPS 2020
[Ho+ 2022] “High Definition Video Generation with Diffusion Models”, arXiv:2210.02303
[Vincent+ 2010] “Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion”, JMLR 2010
[Vincent+ 2011] “A Connection Between Score Matching and Denoising AutoEncoder”, Neural Computation 2011

[Song+ 2021] “Score-Based Generative Modeling through Stochastic Differential Equations”, ICLR 2021
[Karras+ 2022] “Elucidating the Design Space of Diffusion-Based Generative Models”, NeurIPS 2022
[Feller 1949] “On the theory of stochastic processes, with particular reference to applications”, In Proc. of Bekerly Symposium on Mathematical Statistics and Probability
[Xu+ 2022] “GeoDiff: a Geometric Diffusion Mdodel for Molecular Conformation Generation”, ICLR 2022
[Ho+ 2022] “Classifier-Free Diffusion Guidance”, arXiv:2207.12598
[Bansal+ 2022] “Cold Diffusion: Inverting Arbitrary Image Transformers Without Noise”, arXiv:2208.09392
[Lezama+ 2023] “Discrete Predictor-Corrector Diffusion Models for Image Synthesis, ICLR 2023