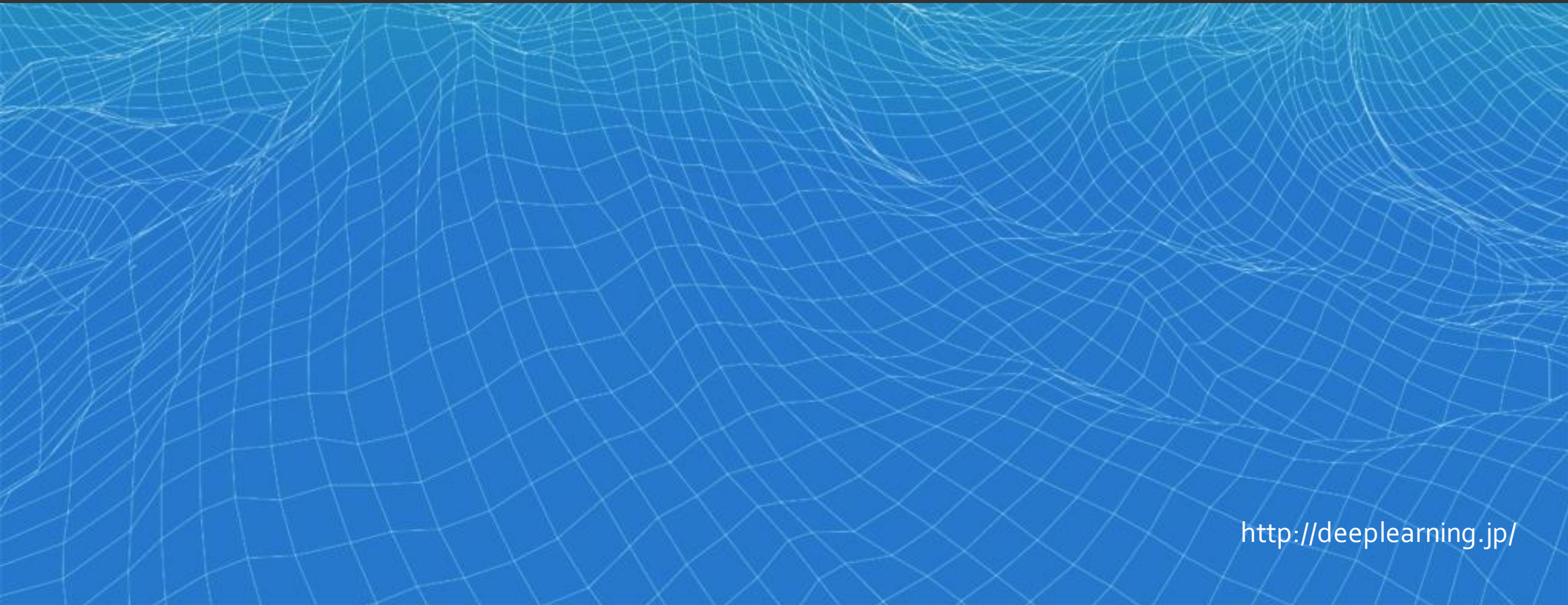
DEEP LEARNING JP
[DL Papers]

Score-based Generative Modeling Secretly Minimizes the Wasserstein Distance

Miyake Daiki, Matsuo Lab, M1



http://deeplearning.jp/

書誌情報

著者: Dohyun Kwon, Ying Fan, Kangwook Lee
(University of Wisconsin-Madison)

NeurIPS2022に採択

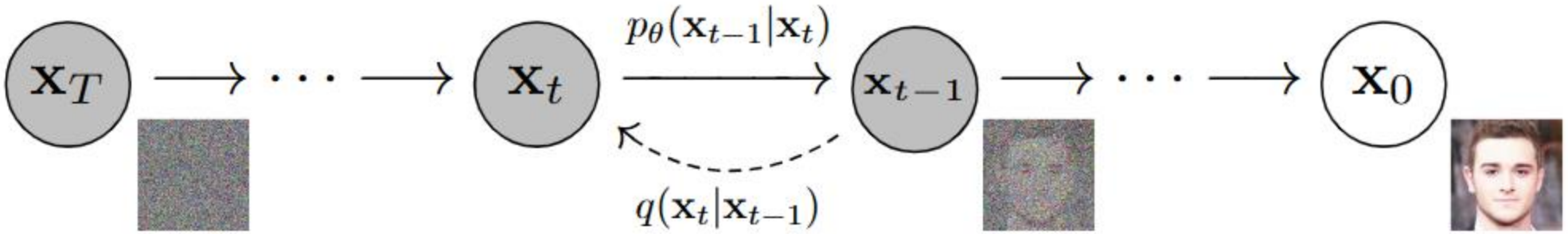
arXiv: https://arxiv.org/abs/2212.06359
NeurIPS2022: https://neurips.cc/virtual/2022/poster/53873

背景: Diffusion Models

- Diffusion Modelは，データにノイズをかける拡散過程(0→T)と，ノイズを外していく逆拡散過程(T→0)をもつ

- モデルは以下の損失関数により画像にかけられたノイズを予測する

$$\mathbb{E}_{x_t}\left[\|\epsilon_\theta(x_t, t) - \epsilon\|_2^2\right]$$



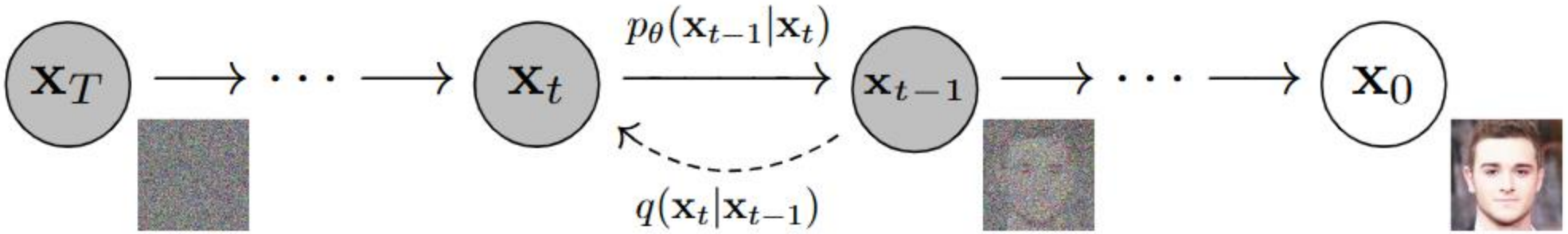
[Ho et al. 2020]

背景: Diffusion Models

- ノイズの予測は，**スコア関数**を予測していることに対応する

$$\mathbb{E}_{x_t}\left[\|s_\theta(x_t,t)-\nabla\log p_t(x_t|x_0)\|_2^2\right]=\mathbb{E}_{x_t}\left[\frac{1}{\sigma_t^2}\|\epsilon_\theta(x_t,t)-\epsilon\|_2^2\right]$$

- この損失関数の最小化は，データ分布とモデル分布とのKL divergenceの上限の最小化に対応する



[Ho et al. 2020]

背景: Diffusion Models

- 拡散過程，逆拡散過程は以下の確率微分方程式で表せる

拡散過程 $$dx = f(x,t)dt + g(t)d\mathbf{w}$$

逆拡散過程 $$dx = (f(x,t) - g(t)^2 \nabla \log p)dt + g(t)d\bar{\mathbf{w}}$$

背景: Wasserstein距離

- KL divergenceを最小化するためには，2つの分布のサポート(確率が0でない領域)が被っている必要がある

$$D_{KL}(q\|p) = \int q(x) \log \frac{q(x)}{p(x)} dx$$

q(x)が0ならp(x)は無視される

p(x)が0なら発散する

- **Wasserstein距離**は以下の最小化問題の解として定義される

$$\min_{\pi(x,y)} \int\int \pi(x,y)\|x-y\|_2^2 dxdy \quad \text{s.t.} \int \pi(x,y)dy = p(x), \int \pi(x,y)dx = q(y)$$

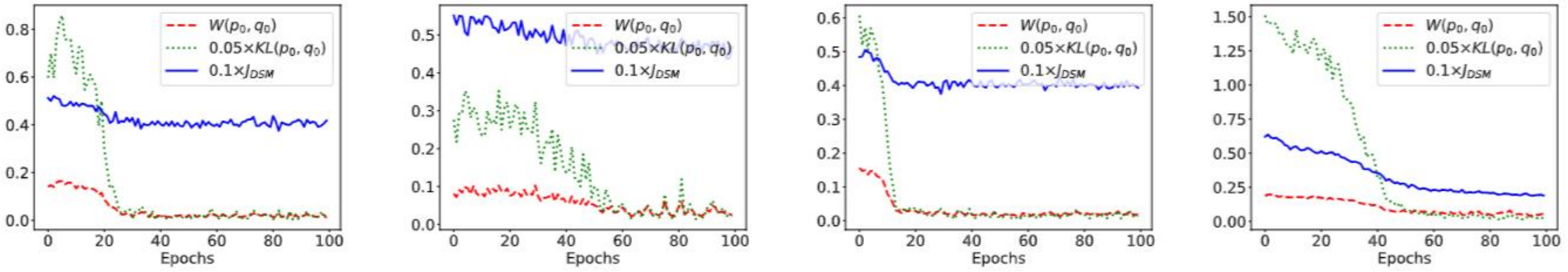
周辺分布がp, q

- Wasserstein距離は2つの分布のサポートが被っていなくても距離として機能する(一般にf-divergenceとIntegral Probability Metricについても同じことがいえる)

導入

- Diffusion modelの学習はKLの上限の最小化に対応することが知られていた

- 実験をしてみると，学習が進むにつれてWasserstein距離も小さくなっていることが分かった
  →Diffusion modelは実はWasserstein距離も最小化しているのでは？



(a) 1 cluster in 1D　(b) 2 clusters in 1D　(c) 1 cluster in 2D　(d) 4 clusters in 2D

準備

- 8個の仮定をおく（詳細は論文の 3.1 Assumptions を参照）

- (A1) $f(x,t)$ が x に対して $L_f(t)$ -リプシッツ連続，すなわち，任意の x, y で

$$\|f(x,t) - f(y,t)\| \leq L_f(t)\|x - y\|$$

を満たす

- (A2) $s_\theta(x,t)$ が x に対して $L_s(t)$ -片側リプシッツ連続，すなわち，任意の x, y で

$$(s_\theta(x,t) - s_\theta(y,t)) \cdot (x - y) \leq L_s(t)\|x - y\|^2$$

を満たす

主要な定理

- データ分布とモデル分布(t=0)でのWasserstein距離の上限を，Diffusion modelの損失関数を含む形で導出できる

学習によって最小化される

$$W_2(p_0, q_0) \leq \int_0^T g(t)^2 I(t) \mathbb{E}_{p_t(x)} \left[\| \nabla \log p_t(x) - s_\theta(x, t) \|^2 \right]^{\frac{1}{2}} dt + I(T) W_2(p_T, q_T)$$

$$I(t) = \exp\left(\int_0^t \left(L_f(r) + L_s(r) g(r)^2 \right) dr \right)$$

異なる損失関数による上限

- 以下の関係式が成り立つ

$$\mathbb{E}_{x_t}\left[\|s_\theta(x_t,t)-\nabla\log p_t(x_t)\|_2^2\right]=\mathbb{E}_{x_t}\left[\|s_\theta(x_t,t)-\nabla\log p_t(x_t|x_0)\|_2^2\right]+\text{Const.}$$

計算できない　　計算できる

- 左辺は実際には計算できないため，代わりに右辺を最小化する

- 前述の上限は左辺を使ったものだったが，右辺を使った別の上限を求めることができる

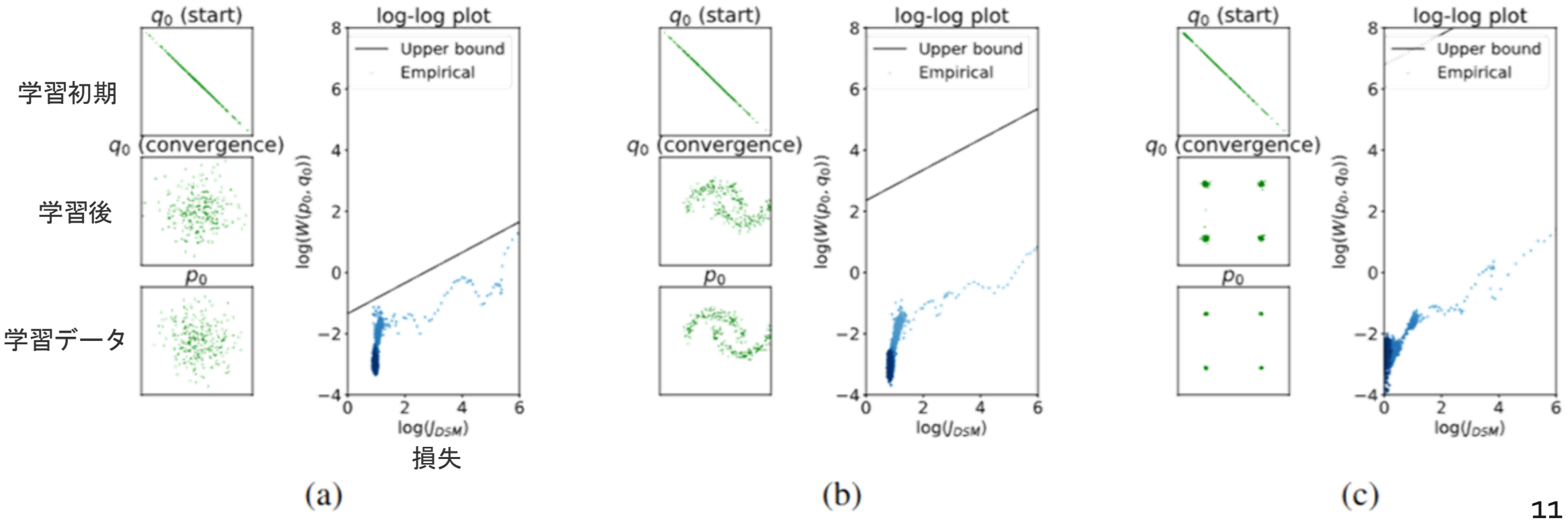
$$W_2(p_0,q_0)\le\sqrt{2\left(\int_0^T g(t)^2I(t)^2dt\right)\mathbb{E}_{p_t(x)}\left[\frac{1}{2}\|\nabla\log p_t(x|x_0)-s_\theta(x,t)\|^2\right]}+I(T)W_2(p_T,q_T)$$

$$I(t)=\exp\left(\int_0^t\left(L_f(r)+L_s(r)g(r)^2\right)dr\right)$$

実験

- 2次元データを4層MLPで学習

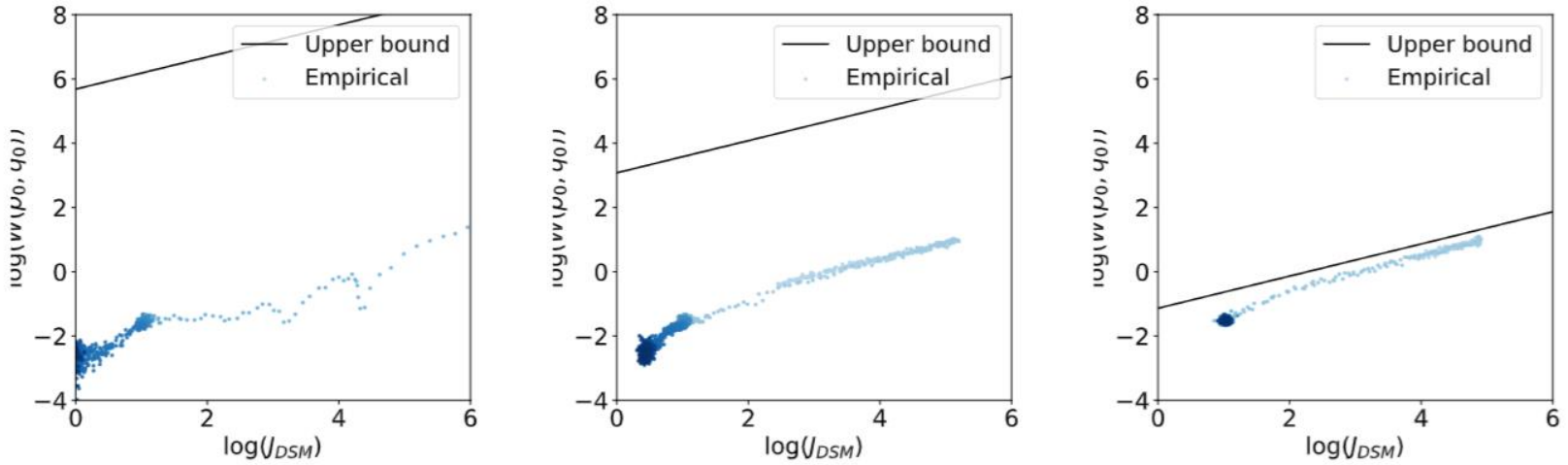
- 学習が進む(=損失が小さくなる)につれて，実際のWasserstein距離(青点)も小さくなっている



実験2

- $L_s(t)$ の値を小さくすることで，Wasserstein距離の値と上限の値とのギャップを小さくできる
- Spectral NormalizationやWeight clippingによって，モデルのリプシッツ定数を小さく抑えられる

- 実際にはギャップは小さくなるものの，損失が小さくなりきらず，Wasserstein距離も大きくなってしまう

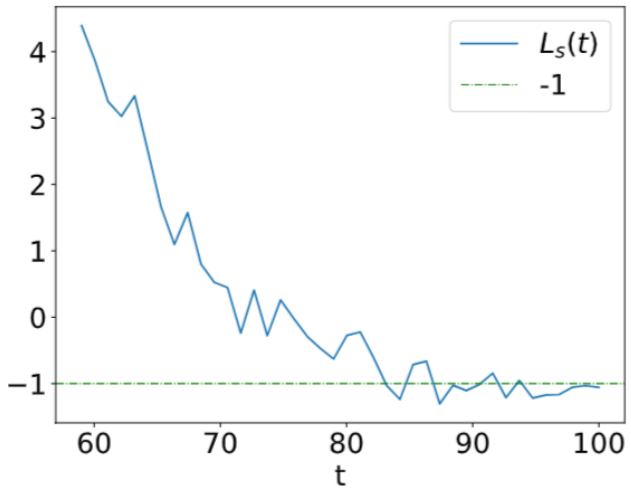


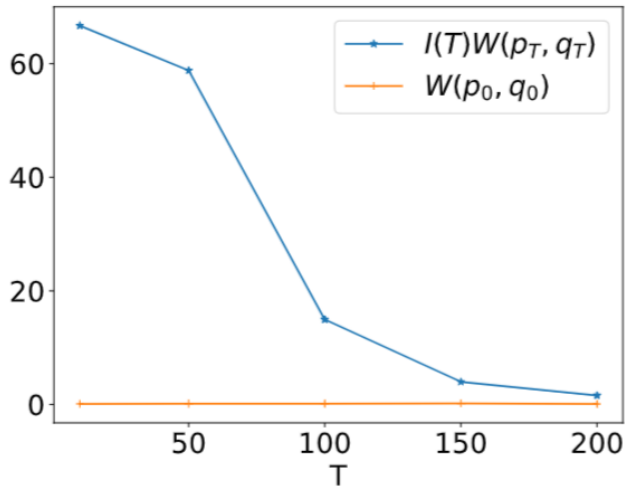
(a) Vanilla (b) Spectral normalization (c) Weight clipping

実験3

- DDPMの設定のもとでは，$L_s(t)$ は t が大きくなるにつれて-1に収束することが示せる

- T が大きくなるにつれて，$W(p_T, q_T)$ も小さくなる
  - p_T が標準正規分布に近づくため
  - 上の話と合わせると，$L_s(t)$ を被積分関数に含む $I(t)$ も小さくなるため，上限のバイアスも小さくなっていく





議論

- この上限のもとでは，たとえ損失が0になったとしても，Wasserstein距離が0になることは保証されない
  - どんな分布に収束するのかわからない

- 一般的なモデルで片側リプシッツ定数 $L_s(t)$ を推定するのはNP困難
  - 正確な上限を求めることは一般的には不可能
  - (両側)リプシッツ定数であれば求められるが，より緩い上限になってしまう

結論

- Diffusion modelの損失の最小化がWasserstein距離の上限の最小化に対応する
- リプシッツ定数を小さくするような工夫を加えることで，上限と実際の値とのギャップを小さくすることができる