Stein's Method for Modern Machine Learning

From Gradient Estimation to Generative Modeling

Jiaxin Shi

Google DeepMind

2024/3/14 @ OT-Berlin

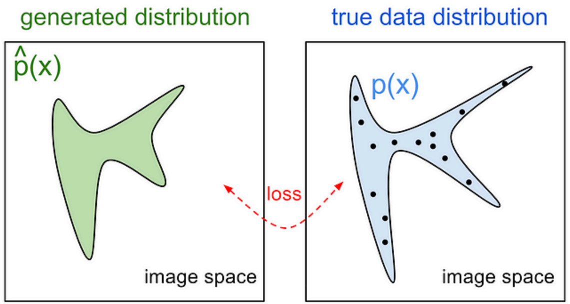
jiaxins.io

Outline

- Stein’s method: Foundations
- Stein’s method and machine learning
    - Sampling
    - Gradient estimation
    - Score-based modeling

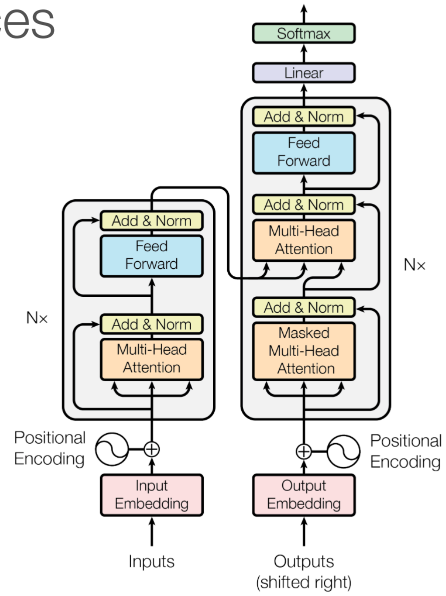
Divergences between Probability Distributions

- How well does my model fit the data?
- Parameter estimation by minimizing divergences
- Sampling as optimization



https://openai.com/blog/generative-models/

GANs, Diffusion models



Transformers

Integral Probability Metrics (IPM)

$$d_{\mathscr{H}}(q, p) = \sup_{h \in \mathscr{H}} |\mathbb{E}_q[h(X)] - \mathbb{E}_p[h(Y)]|$$

- When $\mathscr{H}$ is sufficient large, convergence in $d_{\mathscr{H}}(q_n, p)$ implies q_n weakly converges to p
- Examples: Total variation distance, Wasserstein distance
- **Problem**: Often p is our model and integration under p is intractable
- **Idea**: Only consider functions with $\mathbb{E}_p[h(Y)] = 0$

Stein's Method

- Identify an operator $\mathcal{T}$ that generates mean-zero functions under target distribution p.

$$\mathbb{E}_p[(\mathcal{T} g)(X)] = 0 \text{ for all } g \in \mathcal{G}$$

- Define the Stein discrepancy:

$$\mathcal{S}(q, \mathcal{T}, \mathcal{G}) \triangleq \sup_{g \in \mathcal{G}} \mathbb{E}_q[(\mathcal{T} g)(X)] - \mathbb{E}_p[(\mathcal{T} g)(X)]$$

- Show that the Stein discrepancy is lower bounded by an IPM. For example, if for any $h \in \mathcal{H}$, a solution $g \in \mathcal{G}$ exists for the equation $h(x) - \mathbb{E}_p[h(Y)] = (\mathcal{T} g)(x)$, then $d_{\mathcal{H}}(q, p) \leq \mathcal{S}(q, \mathcal{T}, \mathcal{G})$.

[Stein, 1972]

Identifying a Stein Operator

Stein's Lemma

If p is a standard normal distribution, then

$$\mathbb{E}_p[g'(X) - Xg(X)] = 0 \text{ for all } g \in C_b^1$$

The corresponding Stein operator: $\mathcal{T}(g) = g'(x) - xg(x)$

Identifying a Stein Operator

Barbour's generalization via stochastic processes

- The (infinitesimal) generator A of a stochastic process $(X_t)_{t\geq 0}$ is defined as

$$(Af)(x) = \lim_{t\to 0} \frac{\mathbb{E}[f(X_t)\,|\,X_0 = x] - f(x)}{t}.$$

- The generator of a stochastic process with stationary distribution p satisfies $\mathbb{E}_p[(Af)(X)] = 0$.

[Barbour, 1988 & 1990]

Langevin Stein Operator

- Langevin diffusion on $\mathbb{R}^d$: $dX_t = \nabla \log p(X_t)dt + \sqrt{2}dW_t$
- Generator:

$$(Af)(x) = \nabla \log p(x)^\top \nabla f(x) + \nabla \cdot \nabla f(x)$$

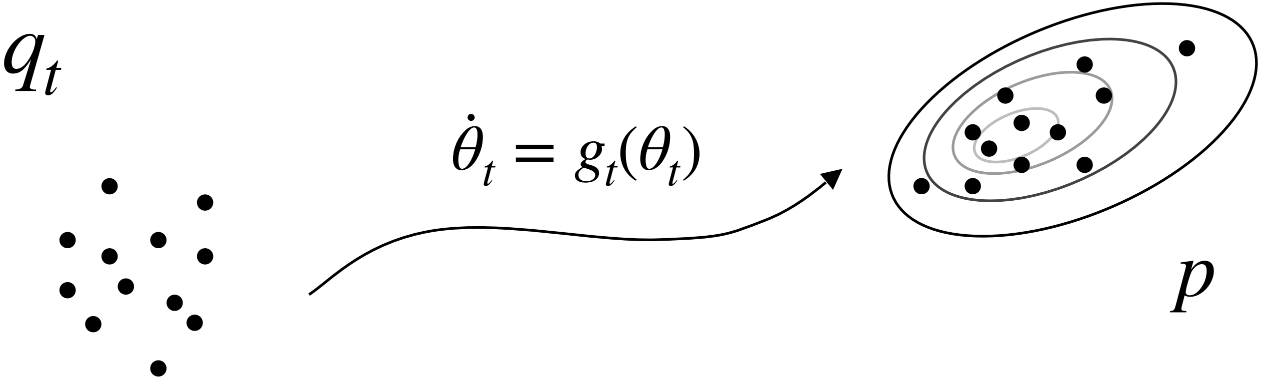
- Convenient form with a vector-valued function $g : \mathbb{R}^d \to \mathbb{R}^d$:

$$(\mathcal{T}_p g)(x) = \nabla \log p(x)^\top g(x) + \nabla \cdot g(x)$$

- Depends on p only through $\nabla \log p$, computable even for unnormalized p

[Gorham & Mackey, 2015]

Stein Operators and Sampling



Find the direction that most quickly decreases the KL divergence to p

$$\frac{d}{dt}\mathrm{KL}(q_t \| p) = -\mathbb{E}_{q_t}[(\mathcal{T}_p g_t)(X)]$$

[Liu & Wang, 2016]

Wasserstein Gradient Flow and SVGD

$$\inf_{g_t \in \mathcal{G}} \frac{d}{dt} \mathrm{KL}(q_t \| p) = - \sup_{g_t \in \mathcal{G}} \mathbb{E}_{q_t}[(\mathcal{T}_p g_t)(X)]$$

- $\mathcal{G} = \mathcal{L}^2(q_t)$: Wasserstein Gradient Flow

$$g_t^* \propto \nabla \log p - \nabla \log q_t,$$

  Same density evolution as Langevin diffusion

- $\mathcal{G}$ = RKHS of kernel K: Stein Variational Gradient Descent [Liu & Wang, 2016]

$$g_t^* \propto \mathbb{E}_{q_t}[K(\cdot, X) \nabla \log p(X) + \nabla_X \cdot K(\cdot, X)]$$

Convergence Analysis of SVGD

- Korba, A., Salim, A., Arbel, M., Luise, G., & Gretton, A. A non-asymptotic analysis for Stein variational gradient descent. *NeurIPS* (2020).

- Chewi, S., Le Gouic, T., Lu, C., Maunu, T., & Rigollet, P. SVGD as a kernelized Wasserstein gradient flow of the chi-squared divergence. *NeurIPS* (2020).

- Shi, J., & Mackey, L. A finite-particle convergence rate for Stein variational gradient descent. *NeurIPS* (2023).

  Convergence rate for discrete-time, finite-particle SVGD

Stein’s Method and Gradient Estimation

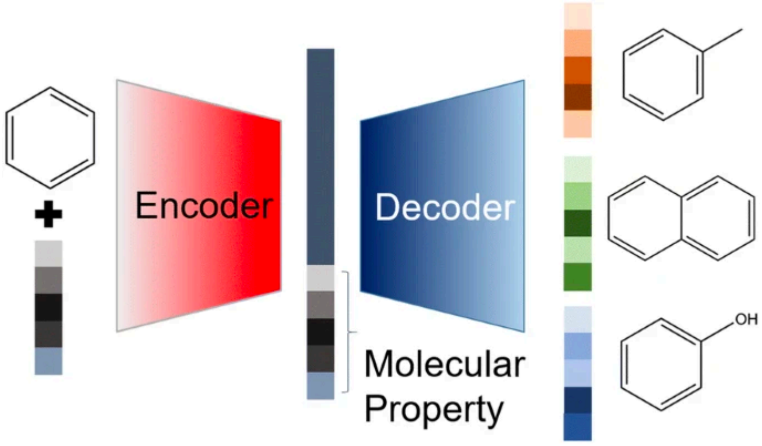
The Gradient Estimation Problem

A common problem in training generative models and reinforcement learning

$$\nabla_\eta \mathbb{E}_{q_\eta}[f(X)]$$

Encoder

Decoding error



[Lim et al., 2018]

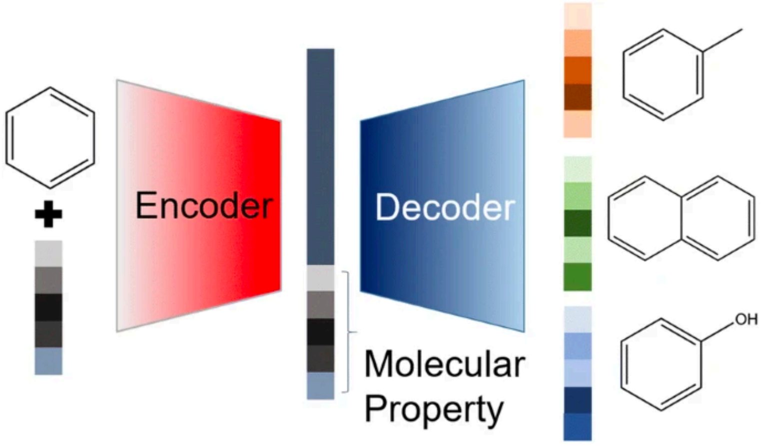
The Gradient Estimation Problem

A common problem in training generative models and reinforcement learning

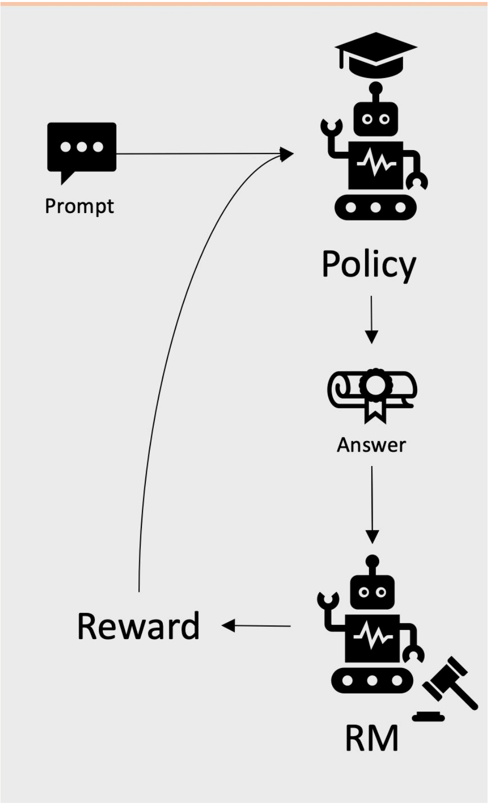
$$\nabla_\eta \mathbb{E}_{q_\eta}[f(X)]$$

Policy

Reward Model



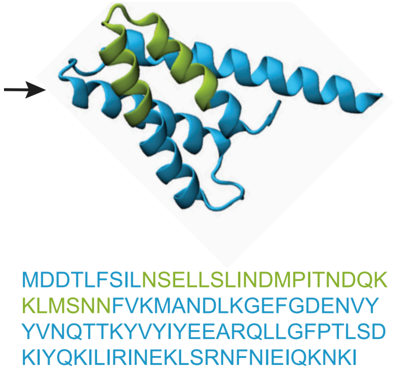
[Lim et al., 2018]



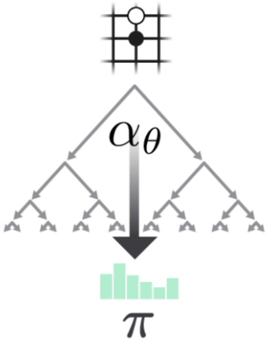
https://aws.amazon.com/de/what-is/reinforcement-learning-from-human-feedback/

Discrete Gradient Estimation

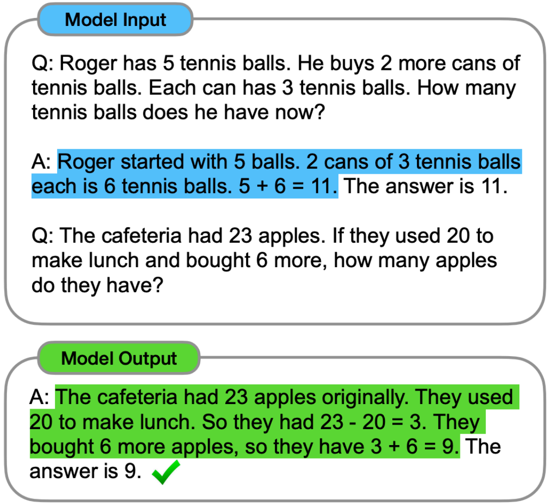
- Discrete data, states, and actions



[Alamdari et al., 2023]



[Silver et al., 2017]



[Wei et al., 2022]

- Computing exact gradients is often intractable

Intractable sum over 2^d configurations

$$\nabla_\eta \mathbb{E}_{q_\eta}[f(X)] = \nabla_\eta \sum_{x\in\{0,1\}^d} q_\eta(x) f(x)$$

d-dimensional binary vector

Complex, nonlinear function

Gradient Estimation and Variance Reduction

$$\hat{g}_1 = \frac{1}{K}\sum_{k=1}^{K} f(x_k)\,\nabla_\eta \log q_\eta(x_k) \qquad \text{(REINFORCE)}$$

High variance!

Control Variates

$$\hat{g}_2 = \frac{1}{K}\sum_{k=1}^{K} [f(x_k)\,\nabla_\eta \log q_\eta(x_k) \; + \; cv(x_k)] - \mathbb{E}_{q_\eta}[cv(X)]$$

- Strong correlation is required for effective variance reduction
- **Fundamental tradeoff:** cv needs to be very *flexible* but still have *analytic expectation* under q_η.

A: Stein Operator

$$\hat{g}_2 = \frac{1}{K}\sum_{k=1}^{K} [f(x_k)\,\nabla_\eta \log q_\eta(x_k) \; + \; (Ah)(x_k)] - \underbrace{\mathbb{E}_{q_\eta}[(Ah)(X)]}_{=0}$$

Discrete Stein Operators

How: Apply Barbour's idea to discrete-state Markov chains.

$$\mathbb{E}_q[((K - I)h)(X)] = 0 \quad \xrightarrow{\text{cont. time}} \quad \mathbb{E}_q[(Ah)(X)] = 0$$

K: transfer operator

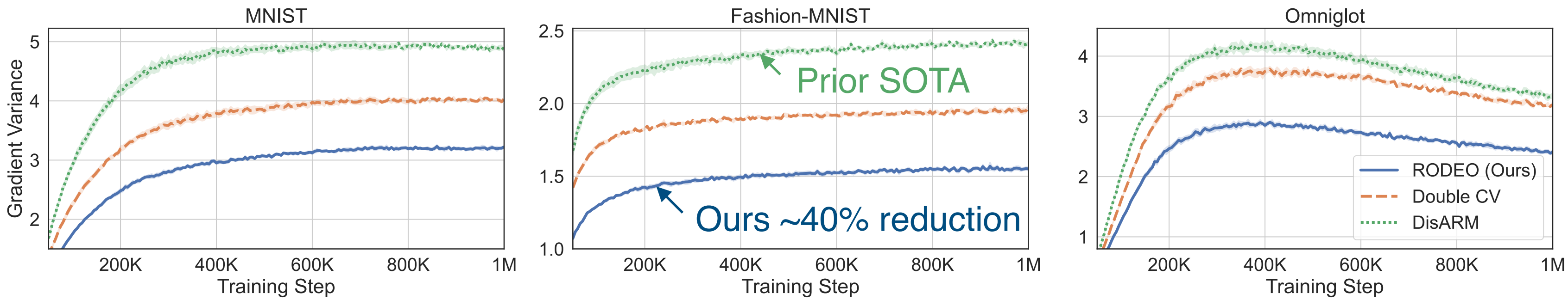
A: generator

Stein Operator	$(Ah)(x)$
Gibbs (4)	$\frac{1}{d}\sum_{i=1}^{d}\sum_{y_{-i}=x_{-i}} q(y_i \mid x_{-i})h(y) - h(x)$
MPF (6)	$\sum_{y\in\mathcal{N}_x, y\neq x}\sqrt{q(y)/q(x)}(h(y) - h(x))$
Barker (6)	$\sum_{y\in\mathcal{N}_x, y\neq x}\frac{q(y)}{q(x)+q(y)}(h(y) - h(x))$
Difference (8)	$\frac{1}{d}\sum_{i=1}^{d} h(\mathtt{dec}_i(x)) - \frac{q(\mathtt{inc}_i(x))}{q(x)}h(x)$

Shi, Zhou, Hwang, Titsias & Mackey. Gradient estimation with discrete Stein operators, NeurIPS 2022 Outstanding Paper.

Experiments: Training Binary Latent VAEs

		Bernoulli Likelihoods			Gaussian Likelihoods		
		MNIST	Fashion-MNIST	Omniglot	MNIST	Fashion-MNIST	Omniglot
$K=2$	DisARM	-102.75 ± 0.08	-237.68 ± 0.13	-116.50 ± 0.04	668.03 ± 0.61	182.65 ± 0.47	446.61 ± 1.16
	Double CV	-102.14 ± 0.06	-237.55 ± 0.16	-116.39 ± 0.10	676.87 ± 1.18	186.35 ± 0.64	447.65 ± 0.87
	RODEO (Ours)	$\mathbf{-101.89 \pm 0.17}$	$\mathbf{-237.44 \pm 0.09}$	$\mathbf{-115.93 \pm 0.06}$	$\mathbf{681.95 \pm 0.37}$	$\mathbf{191.81 \pm 0.67}$	$\mathbf{454.74 \pm 1.11}$
$K=3$	ARMS	-100.84 ± 0.14	-237.05 ± 0.12	-115.21 ± 0.07	683.55 ± 1.01	193.07 ± 0.34	457.98 ± 1.03
	Double CV	-100.94 ± 0.09	-237.40 ± 0.11	-115.06 ± 0.12	686.48 ± 0.68	193.93 ± 0.20	457.44 ± 0.79
	RODEO (Ours)	$\mathbf{-100.46 \pm 0.13}$	$\mathbf{-236.88 \pm 0.12}$	$\mathbf{-115.01 \pm 0.05}$	$\mathbf{692.37 \pm 0.39}$	$\mathbf{196.56 \pm 0.42}$	461.87 ± 0.90
RELAX (3 evals)		-101.99 ± 0.04	-237.74 ± 0.12	-115.70 ± 0.08	688.58 ± 0.52	196.38 ± 0.66	$\mathbf{462.23 \pm 0.63}$



Shi, Zhou, Hwang, Titsias & Mackey. Gradient estimation with discrete Stein operators, NeurIPS 2022 Outstanding Paper.

Stein’s Method and Score-Based Modeling

Stein Discrepancy as a Learning Rule

?

Model fitting: $$\min_{\theta} |\mathbb{E}_q[h(x)^\top \nabla_x \log p_\theta(x) + \nabla \cdot h(x)]|$$

Data distribution

Model distribution

Score Matching

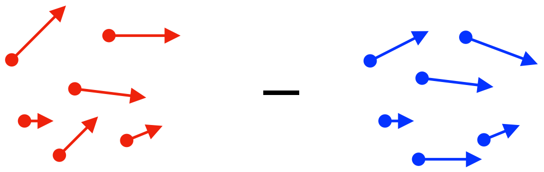
[Hyvärinen, 2005]

Model fitting: $$\min_{\theta} \sup_{h \in L^2(q)} \left| \mathbb{E}_q[h(x)^\top \nabla_x \log p_\theta(x) + \nabla \cdot h(x)] \right|$$

Data distribution

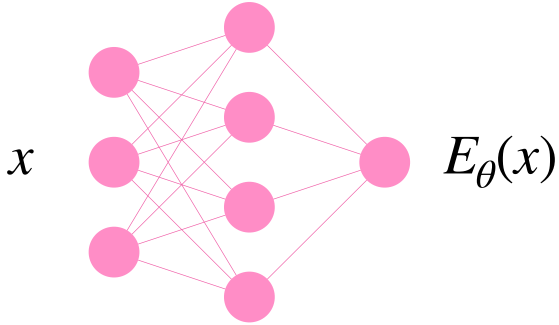
Model distribution

$$\rightarrow \min_{\theta} \mathbb{E}_{q_{\text{data}}}[\|\nabla \log p_\theta(x) - \nabla \log q_{\text{data}}(x)\|^2]$$



Training Energy-Based Models

$$p_\theta(x) = \frac{e^{-E_\theta(x)}}{Z_\theta}$$



Sliced Score Matching

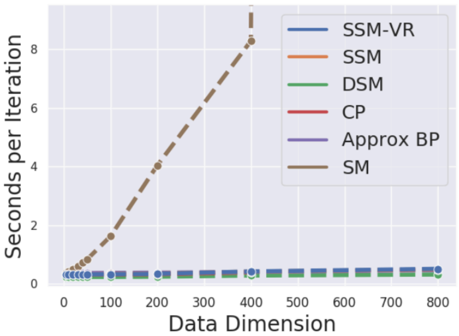
[Song*, Garg*, S & Ermon, UAI'19]

Key insight: The score does not depend on normalizing constant Z_θ

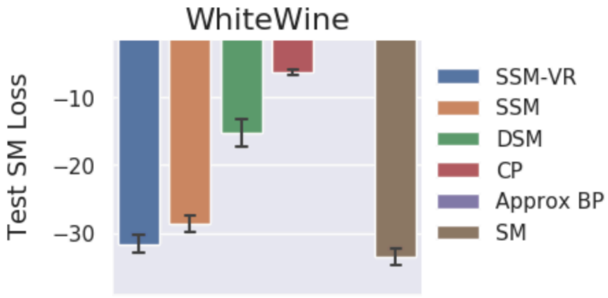
$$\nabla_x \log p_\theta(x) = -\nabla E_\theta(x) + \cancel{\nabla_x \log Z_\theta}$$

- Score Matching is more suitable for training such models than maximum likelihood!

Time



Performance



Song*, Garg*, Shi & Ermon. Sliced score matching: A scalable approach to density and score estimation. UAI 2019

Score-Based Modeling

Idea: Model the score $s := \nabla \log p$ instead of the density

Advantages:

1. less computation than energy-based modeling
2. enable more flexible models

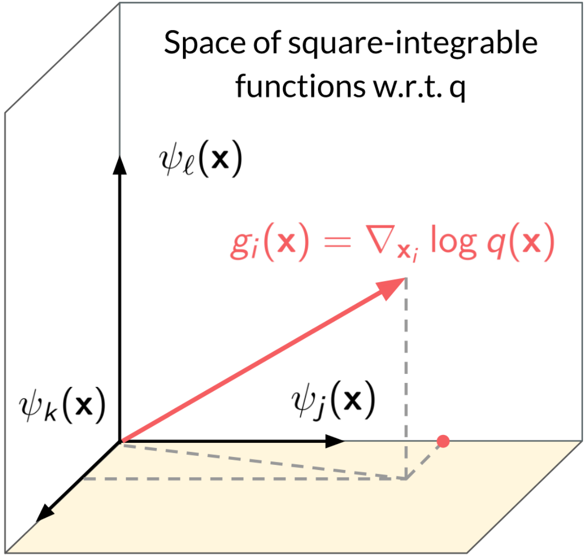
Nonparametric Score Model

$$\min_{s \in \mathscr{H}} \mathbb{E}_{q_{\text{data}}} \|s(x) - \nabla \log q_{\text{data}}(x)\|^2 + \frac{\lambda}{2}\|s\|^2_{\mathscr{H}}$$

The spectral estimator (Shi et al., 18) is a special case.

Zhou, Shi & Zhu. Nonparametric score estimators. ICML 2020

A Spectral Method for Score Estimation



$$\nabla_{\mathbf{x}} \log q(\mathbf{x}) = -\sum_{j \geq 1} \mathbb{E}_q \left[\nabla \psi_j(\mathbf{x}) \right] \psi_j(\mathbf{x})$$

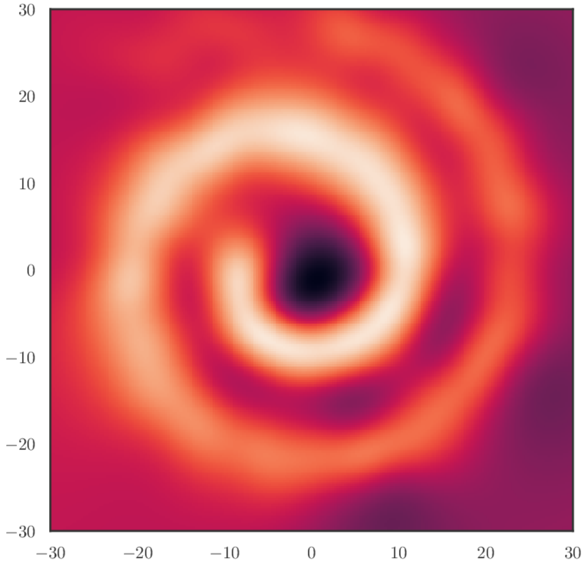
density gradients (score) eigenfunction

$$\langle \nabla \log q, \psi_j \rangle_{L^2(q)} = -\mathbb{E}_q[\nabla \psi_j(x)]$$

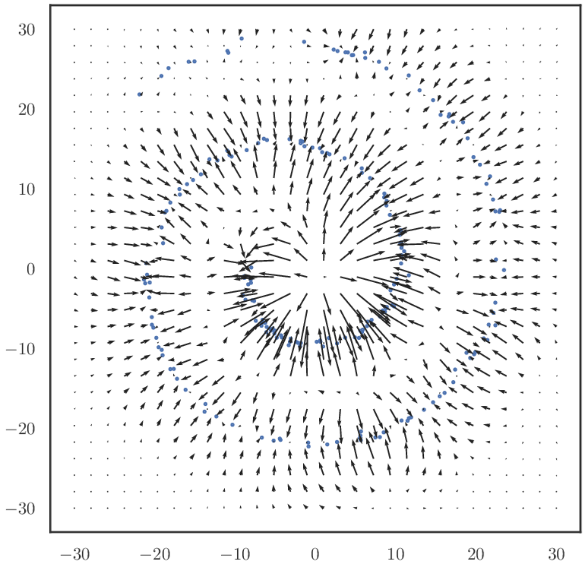
$$\mathbb{E}_{\mathbf{x}' \sim q}[k(\mathbf{x}, \mathbf{x}')\psi_j(\mathbf{x}')] = \lambda_j \psi_j(\mathbf{x})$$

Shi, Sun & Zhu. A spectral approach to gradient estimation for implicit distributions. ICML 2018

A Spectral Method for Score Estimation



$q(\mathbf{x})$ (unknown)



$\{\mathbf{x}^j\}_{j=1}^M \overset{\text{i.i.d.}}{\sim} q \;\rightarrow\; \nabla_{\mathbf{x}} \log q(\mathbf{x})$

Score function

Shi, Sun & Zhu. A spectral approach to gradient estimation for implicit distributions. ICML 2018
Zhou, Shi & Zhu. Nonparametric score estimators. ICML 2020

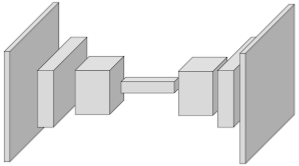
Score-Based Modeling

Idea: Model the score $s := \nabla \log p$ instead of the density

Advantages:

1. less computation than energy-based modeling
2. enable more flexible models

x



$s_\theta(x) \approx \nabla \log q_{\text{data}}(x)$

Nonparametric Score Model

$$\min_{s \in \mathcal{H}} \mathbb{E}_{q_{\text{data}}} \|s(x) - \nabla \log q_{\text{data}}(x)\|^2 + \frac{\lambda}{2} \|s\|^2_{\mathcal{H}}$$

The spectral estimator (Shi et al., 18) is a special case.

Score Network

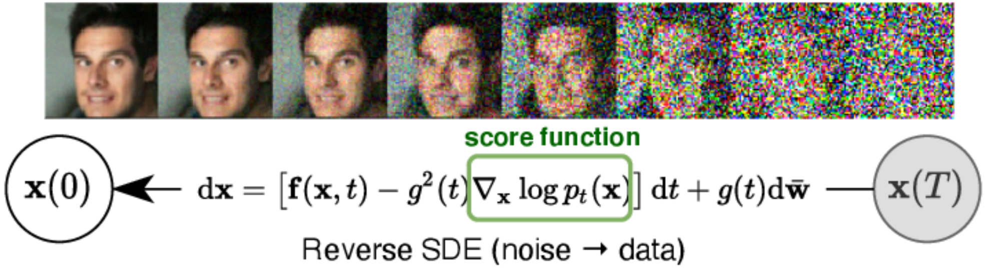
Use neural networks to model score, trained by sliced score matching

$$\min_{\theta} \mathbb{E}_{q_{\text{data}}} \|s_\theta(x) - \nabla \log q_{\text{data}}(x)\|^2$$

Song*, Garg*, Shi & Ermon. Sliced score matching: A scalable approach to density and score estimation. UAI 2019
Zhou, Shi & Zhu. Nonparametric score estimators. ICML 2020

From Score Networks to Diffusion Models

Updates produced by score networks transform noise to data



[Song et al., ICLR'20]

Images created by OpenAI's DALLE-2.
DALLE-2 is based on diffusion models.

Open Problems

- Improving finite-particle rates of SVGD
- Approximately solving the Stein equation for improved gradient estimation
- Lower bounding the discrete Stein discrepancy
- Learning the features in nonparametric score models
- Finding the “right” discrete correspondence of the score matching objective

Joint work with Lester Mackey, Yuhao Zhou, Jessica Hwang, Michalis K. Titsias, Shengyang Sun, Jun Zhu, Yang Song, Sahaj Garg, Stefano Ermon

Main References

- Shi & Mackey. A finite-particle convergence rate for Stein variational gradient descent. NeurIPS 2023.
- Shi, Zhou, Hwang, Titsias, & Mackey. Gradient estimation with discrete Stein operators. NeurIPS 2022
- Titsias & Shi. Double control variates for gradient estimation in discrete latent-variable models. AISTATS 2022
- Shi, Sun, & Zhu. A spectral approach to gradient estimation for implicit distributions. ICML 2018
- Song, Garg, Shi, & Ermon. Sliced score matching: A scalable approach to density and score estimation. UAI 2019
- Zhou, Shi, Zhu. Nonparametric score estimators. ICML 2020

References

- Stein, C. (1972). A bound for the error in the normal approximation to the distribution of a sum of dependent random variables.
- Barbour, A. D. (1988). Stein's method and Poisson process convergence. *Journal of Applied Probability*, *25*(A), 175-184.
- Barbour, A. D. (1990). Stein's method for diffusion approximations. *Probability theory and related fields*, *84*(3), 297-322.
- Gorham, J., & Mackey, L. (2015). Measuring sample quality with Stein's method. *Advances in Neural Information Processing Systems*, *28*.
- Liu, Q., & Wang, D. (2016). Stein variational gradient descent: A general purpose bayesian inference algorithm. *Advances in Neural Information Processing Systems*, *29*.

References

- Lim, J., Ryu, S., Kim, J. W., & Kim, W. Y. (2018). Molecular generative model based on conditional variational autoencoder for de novo molecular design. Journal of Cheminformatics, 10, 1-9.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., ... & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. Advances in Neural Information Processing Systems, 35, 24824-24837.
- Alamdari, S., Thakkar, N., van den Berg, R., Lu, A. X., Fusi, N., Amini, A. P., & Yang, K. K. (2023). Protein generation with evolutionary diffusion: sequence is all you need. bioRxiv, 2023-09.
- Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., ... & Hassabis, D. (2017). Mastering the game of go without human knowledge. Nature, 550(7676), 354-359.
- Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., & Poole, B. (2020). Score-based generative modeling through stochastic differential equations. arXiv preprint arXiv:2011.13456.