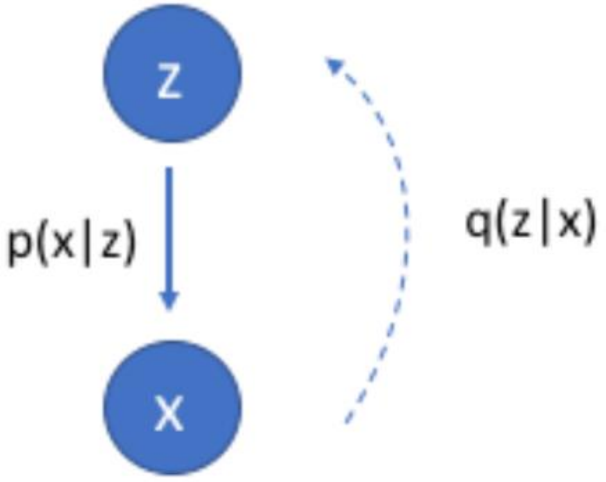
Nonparametric Score Estimators

Yuhao Zhou, Jiaxin Shi, Jun Zhu

Tsinghua University

Gradients of Intractable Log-densities

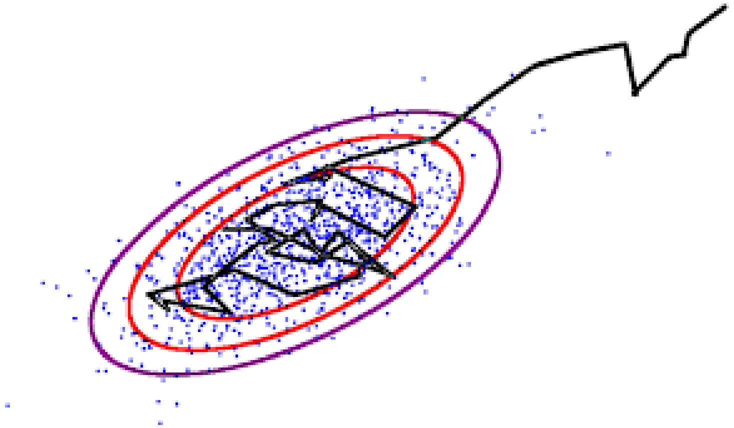
Where do they appear?



Generative Models

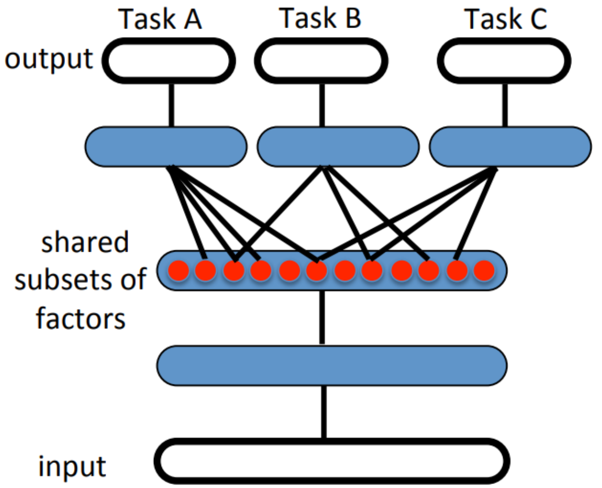
(KL-divergence, entropy)

[Tolstikhin et al., 2017, Song et al., 2019]



Gradient-free MCMC

[Strathmann et al., 2015]



Representation Learning

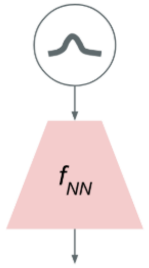
(mutual information)

[Wen et al., 2020]

The Score Estimation Problem

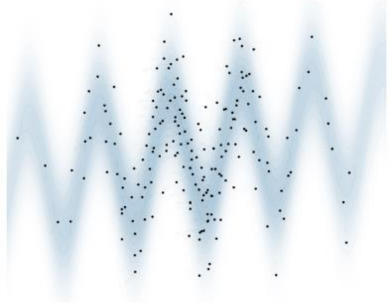
Estimating gradients of log-densities from samples

- Assume that $p(\mathbf{x})$ is **intractable**
  - But **easy to get samples**.
- Given i.i.d. samples $\mathbf{x}^1, \cdots, \mathbf{x}^M \sim p(\mathbf{x})$.
- Estimate $\nabla_{\mathbf{x}} \log p(\mathbf{x})$ using these samples.

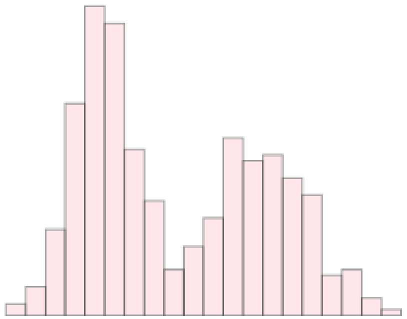


$\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}),\ \mathbf{x} = f_{NN}(\mathbf{z})$

Distributions generated by a **non-invertible** transformation

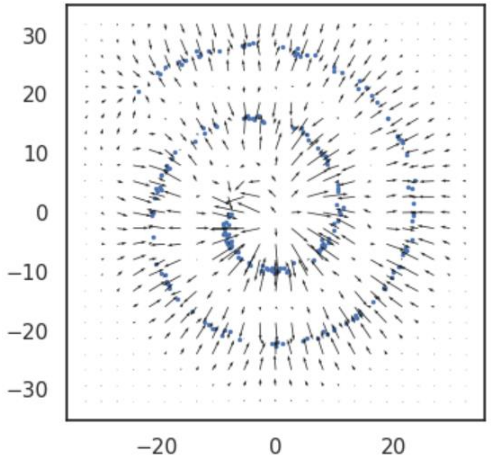


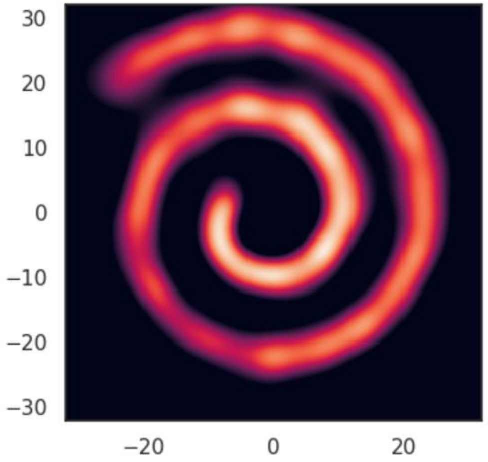
Distributions directly specified by **a set of particles**



$\longrightarrow$ $s_p \coloneqq \nabla_{\mathbf{x}} \log p(\mathbf{x})$

The score function





The Score Estimation Problem

Our contributions

- A unified framework of nonparametric score estimators
  - Based on vector-valued regression + regularization
  - Unifying KEF (Sriperumbudur et al., 2017), Stein (Li & Turner, 2018), SSGE (Shi et al., 2018)
- A unified analysis of the convergence results
  - Recover, improve, or establish the convergence of existing estimators
- Iterative-based curl-free score estimators
  - Reduce the computational complexity depending on dimensions

Score Estimation via Regression

An ideal case

- Given i.i.d. samples $\mathbf{x}^1, \cdots, \mathbf{x}^M \sim p(\mathbf{x})$
- Suppose we know the ground truth $s_p := \nabla \log p$ at sample points
- We can minimize the empirical mean-square error (MSE) in a vector-valued RKHS

$$\hat{s}_{p,\lambda} = \arg\min_{s \in \mathcal{H}_\mathcal{K}} \frac{1}{M} \sum_{m=1}^{M} \| s(\mathbf{x}^m) - s_p(\mathbf{x}^m) \|_2^2 + \frac{\lambda}{2} \| s \|_{\mathcal{H}_\mathcal{K}}^2.$$

- The solution is $\hat{s}_{p,\lambda} = (\hat{L}_\mathcal{K} + \lambda I)^{-1} \hat{L}_\mathcal{K} s_p$ → Empirical estimate of $\mathbb{E}_\mathbf{x}[\mathcal{K}(\mathbf{x}, \cdot) s_p(\mathbf{x})]$

Score Estimation via Regression

Towards the real case

- We can use integration by parts to reformulate the unknown term

$$L_{\mathcal{K}} s_p(\mathbf{y}) = \mathbb{E}_{\mathbf{x}}\left[\mathcal{K}(\mathbf{x},\mathbf{y}) s_p(\mathbf{x})\right] = -\mathbb{E}_{\mathbf{x}}[\mathrm{div}_{\mathbf{x}} \mathcal{K}(\mathbf{x},\mathbf{y})^{\mathrm{T}}]$$

The empirical estimate is known!

- Now, we obtain our estimator

$$\hat{s}_{p,\lambda} = -\left(\hat{L}_{\mathcal{K}} + \lambda I\right)^{-1} \hat{\zeta}, \quad \text{where } \hat{\zeta} = \frac{1}{M} \sum_{i=1}^{M} \mathrm{div}_{\mathbf{x}^m} \mathcal{K}(\mathbf{x}^m, \mathbf{y})^{\mathrm{T}}$$

Score Estimation via Regression

General regularization schemes

- The (Tikhonov) regularization term in the loss approximates the inverse

$$\frac{1}{M}\sum_{m=1}^{M}\|s(\mathbf{x}^m) - s_p(\mathbf{x}^m)\|_2^2 + \boxed{\frac{\lambda}{2}\|s\|_{\mathcal{H}_K}^2}. \quad \Rightarrow \quad \hat{s}_{p,\lambda} = -\boxed{\left(\hat{L}_{\mathcal{K}} + \lambda I\right)^{-1}}\hat{\zeta}$$

Approximation of $\hat{L}_{\mathcal{K}}$

- We can consider the general regularization

$$\hat{s}_{p,\lambda}^{g} = -\boxed{g_\lambda(\hat{L}_K)}\hat{\zeta}$$

- For example, $g_\lambda(\sigma) = (\sigma + \lambda)^{-1}$ is the Tikhonov regularization

Regularization Schemes

$$g_\lambda(\sigma) = (\sigma + \lambda)^{-1}$$

Tikhonov Regularization

$$g_\lambda(\sigma) = \begin{cases} \sigma^{-1} & \sigma > \lambda, \\ 0 & \sigma \leq \lambda. \end{cases}$$

Spectral-Cutoff Regularization

$$g_\lambda(\sigma) = \mathbf{1}_{\{\sigma > 0\}}(\lambda + \sigma)^{-1}$$

Truncated Tikhonov

$$g_\lambda(\sigma) = \mathrm{poly}(\sigma)$$

Iterative-based Regularization

Hypothesis Spaces

Diagonal matrix-valued kernels

- How to choose the matrix-valued kernel?
- Consider a scalar-valued kernel $k: \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, we can use the diagonal kernel

$$\mathcal{K}(\mathbf{x}, \mathbf{y}) \;=\; k(\mathbf{x}, \mathbf{y})\mathbf{I}_d$$

- This corresponds to a product RKHS $\mathcal{H}_k^d := \bigotimes_{i=1}^d \mathcal{H}_k$
  - All output dimensions of the function are independent (like SSGE, Stein)
  - This assumption may not hold for the score function
- The computation cost is low, i.e., $O(M^3)$

Hypothesis Spaces

Curl-free matrix-valued kernels

- We want elements in the RKHS to be a gradient of some functions.
- Consider a scalar-valued RBF kernel $k: \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ with $k(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x} - \mathbf{y})$
- we can construct the curl-free kernel

$$\mathcal{K}_{\mathrm{cf}}(\mathbf{x}, \mathbf{y}) := -\nabla^2 \phi(\mathbf{x} - \mathbf{y})$$

  - Each function in this RKHS is a linear combination of columns of such kernel
  - The j-th column of it is $-\nabla(\partial_j \phi)$, which is a gradient field!
- The computation cost is high, i.e., $O(M^3 d^3)$

[Fuselier Jr, 2007; Macedo & Castro, 2010]

Convergence Rates

The regularization qualification

- The qualification of the regularization g_λ is the maximal ν such that

$$\sup_{0<\sigma\leqslant\kappa^2} |1-g_\lambda(\sigma)\sigma|\sigma^\nu \leqslant \gamma_\nu\lambda^\nu$$

- It turns out that when $s_p = L_{\mathcal{K}}^r f_0$ for some $f_0 \in \mathcal{H}_{\mathcal{K}}$ and $r \in [0, \nu]$,

$$\|\hat{s}_{p,\lambda}^g - s_p\|_{\mathcal{H}_{\mathcal{K}}} = O_p\left(M^{-\frac{r}{2r+2}}\right)$$

- The number r reflects the “smoothness” of the score
- The maximal convergence rate depends on the regularization qualification
  - The spectral cutoff regularization, and iterative regularization are theoretically better!

Different Kernel Score Estimators

Unified in our framework

Algorithm	Kernel	Regularizer	Complexity	Rate (original)	Rate (this work)
SSGE [1]	Diagonal	Spectral-Cutoff	$O(M^3)$	$\leq 1/8$	$[1/4, 1/2]$
Stein [2]	Diagonal	Truncated Tikhonov	$O(M^3)$	None	$[0, 1/4]$*
KEF [3]	Curl-Free	Tikhonov	$O(M^3 d^3)$	$[1/4, 1/3]$	$[1/4, 1/3]$
NKEF [4]	Curl-Free	Truncated Tikhonov	$O(MN^2 d^3)$	$[1/4, 1/3]$	$[1/4, 1/3]$

- M is the sample size, d is the dimension, $N \approx \sqrt{M} \log M$
- The rate of Stein uses the sup-norm, others are L^2-norm

[1] Shi, J., Sun, S., and Zhu, J. A spectral approach to gradient estimation for implicit distributions. ICML 2018.
[2] Li, Y. and Turner, R. E. Gradient estimators for implicit models. ICLR 2018.
[3] Sriperumbudur, B., Fukumizu, K., Gretton, A., Hyvarinen, A., and Kumar, R. Density estimation in infinite dimensional exponential families. JMLR 2017.
[4] Sutherland, D., Strathmann, H., Arbel, M., and Gretton, A. Efficient and principled score estimation with nystr¨om kernel exponential families. AISTATS 2018.

Iterative Curl-free Estimators

Reduce the complexity on the dimension

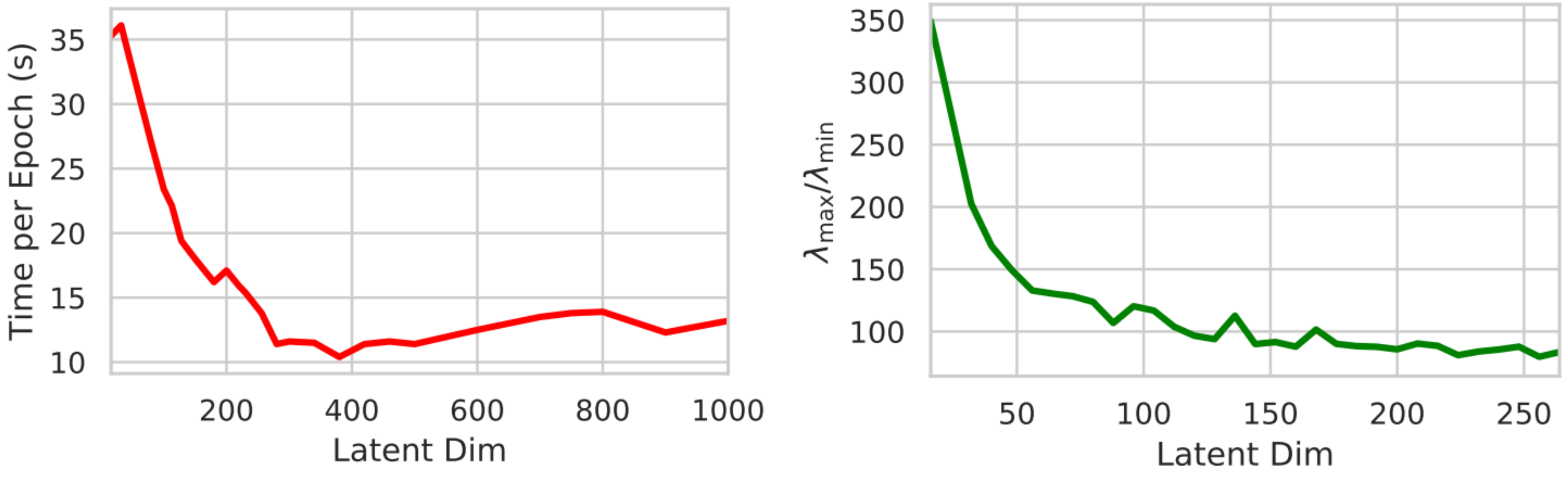
- We proposed two iterative methods that only require matrix-vector multiplication
  - Iterative regularization methods (ν-method)
  - Using Conjugate Gradients to solve linear systems
- When using curl-free kernels, induced by $k(\mathbf{x}, \mathbf{y}) = \phi(||\mathbf{x} - \mathbf{y}||)$

$$\mathcal{K}_{\mathrm{cf}}(\mathbf{x}, \mathbf{y}) = \left(\frac{\phi'}{r^3} - \frac{\phi''}{r^2} \right) \mathbf{r}\mathbf{r}^\top - \frac{\phi'}{r}\mathbf{I}$$

  - This is rank-one matrix + identity matrix
  - Reduce the complexity of matrix-vector multiplication from $O(M^2 d^2)$ to $O(M^2 d)$

Iterative Curl-free Estimators

Spectral decay of kernel matrices



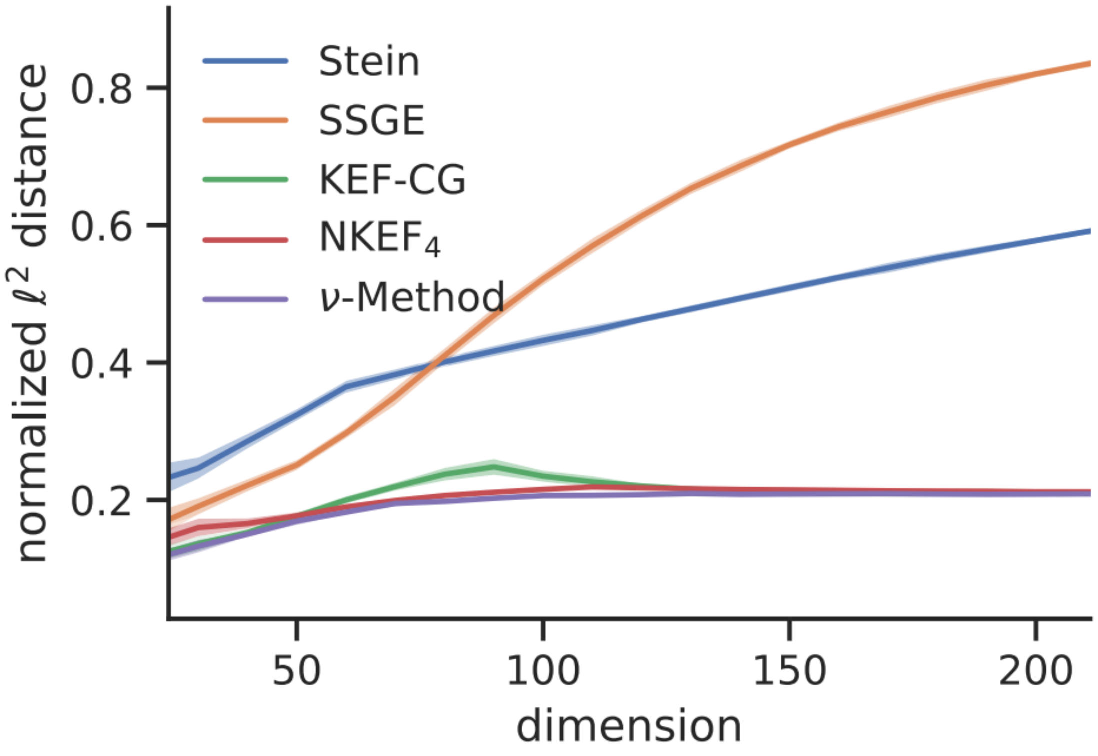
(a) Computational Cost

(b) $\lambda_{max}/\lambda_{min}$

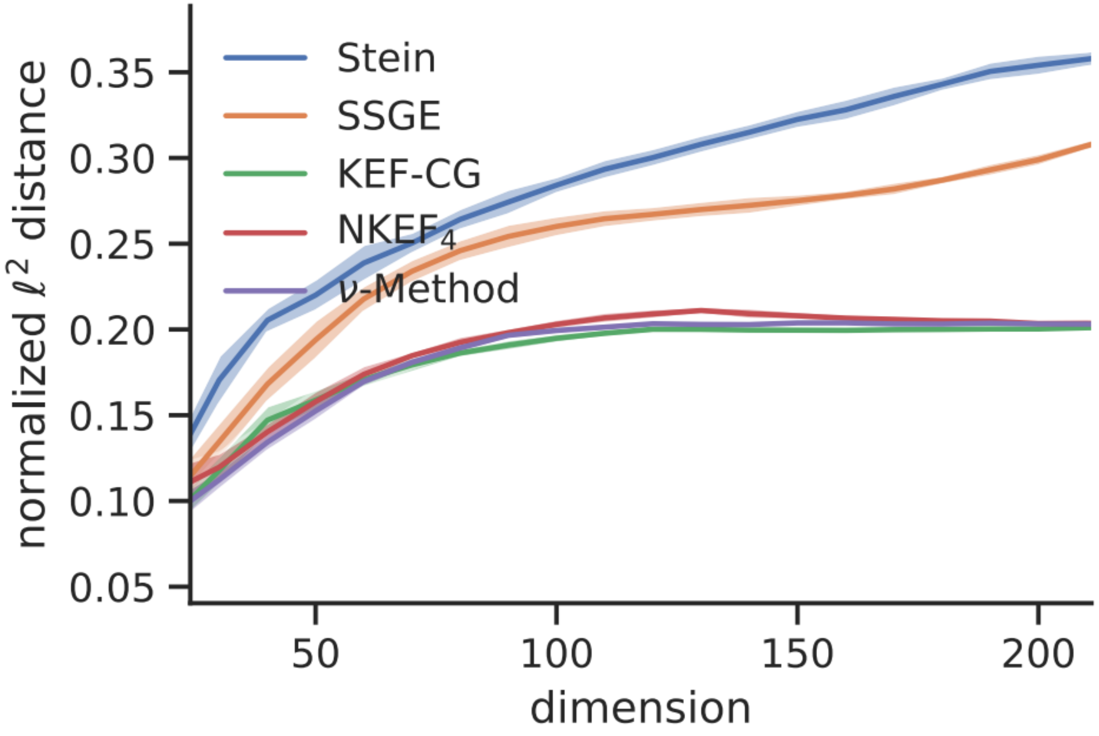
Figure 2. (a) Computational costs of KEF-CG for $\lambda = 10^{-5}$ on MNIST; (b) The ratio of the maximum and the minimum eigenvalues of kernel matrices.

Toy Experiments

A grid distribution



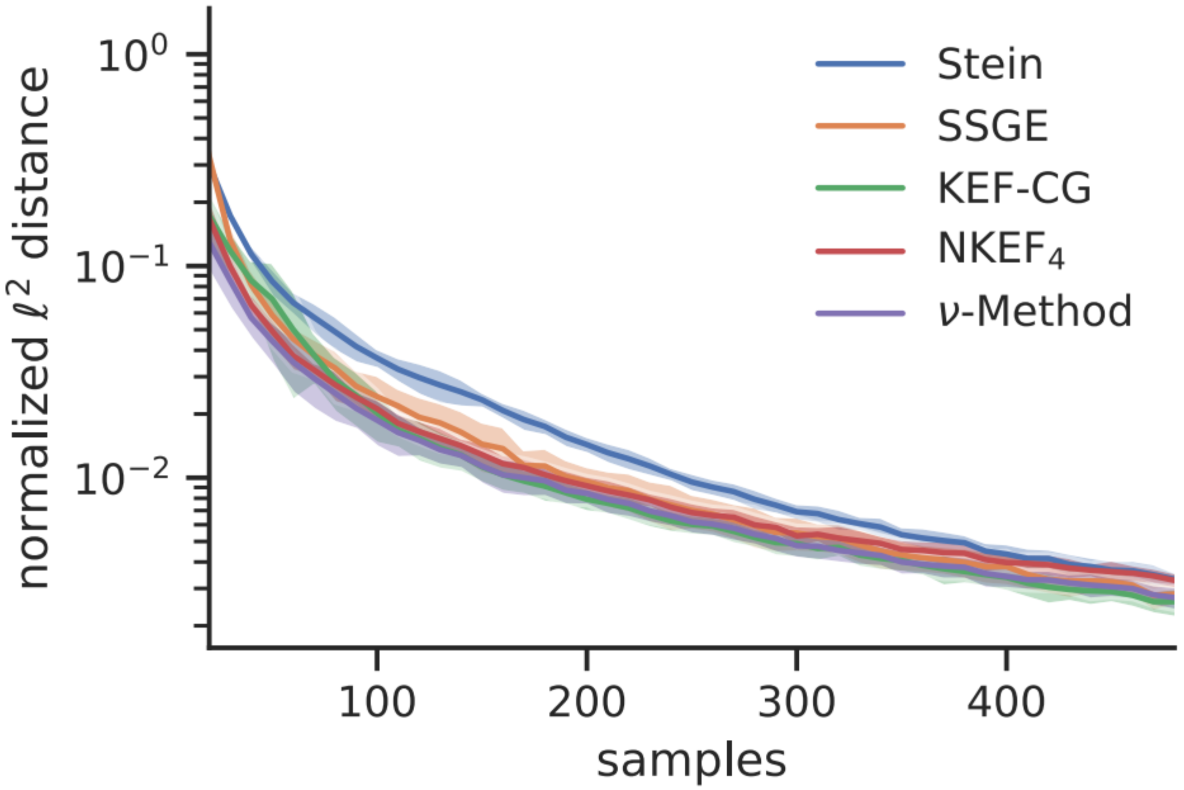
(a) $M = 128$



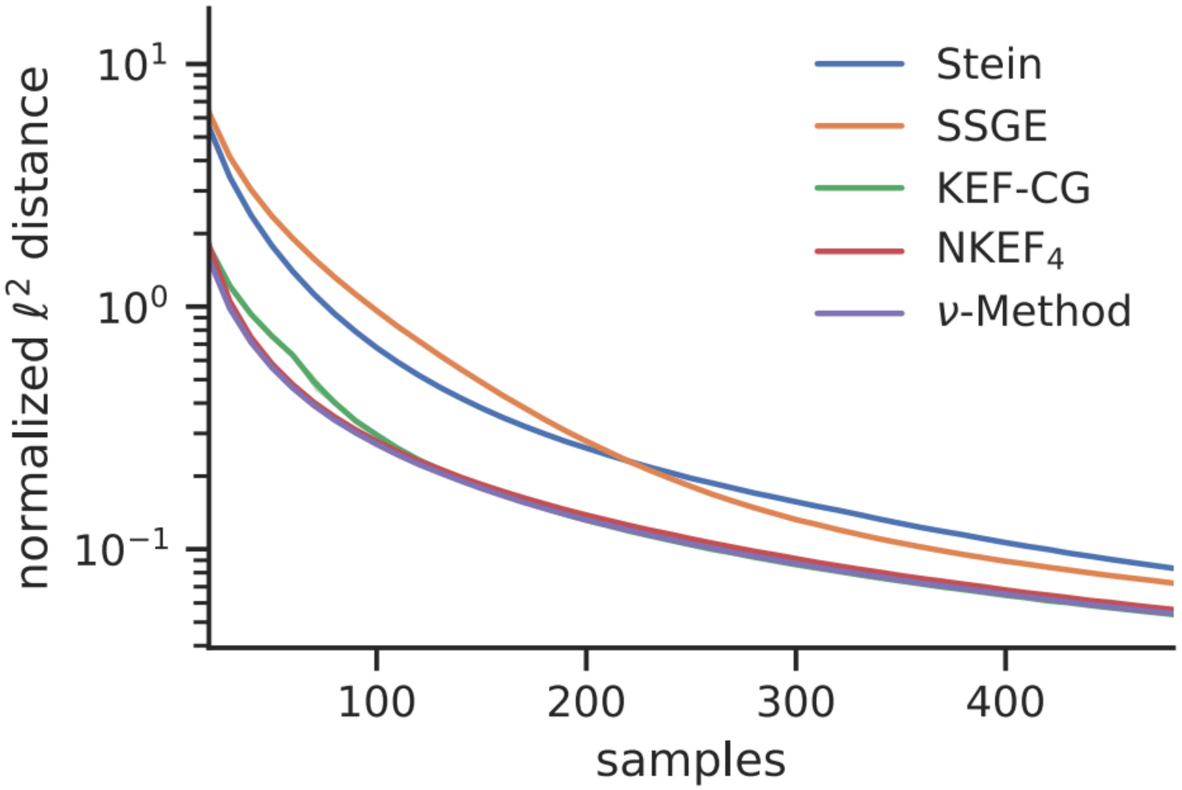
(b) $M = 512$

Toy Experiments

A grid distribution



(c) $d = 16$



(d) $d = 128$

Wasserstein Autoencoders

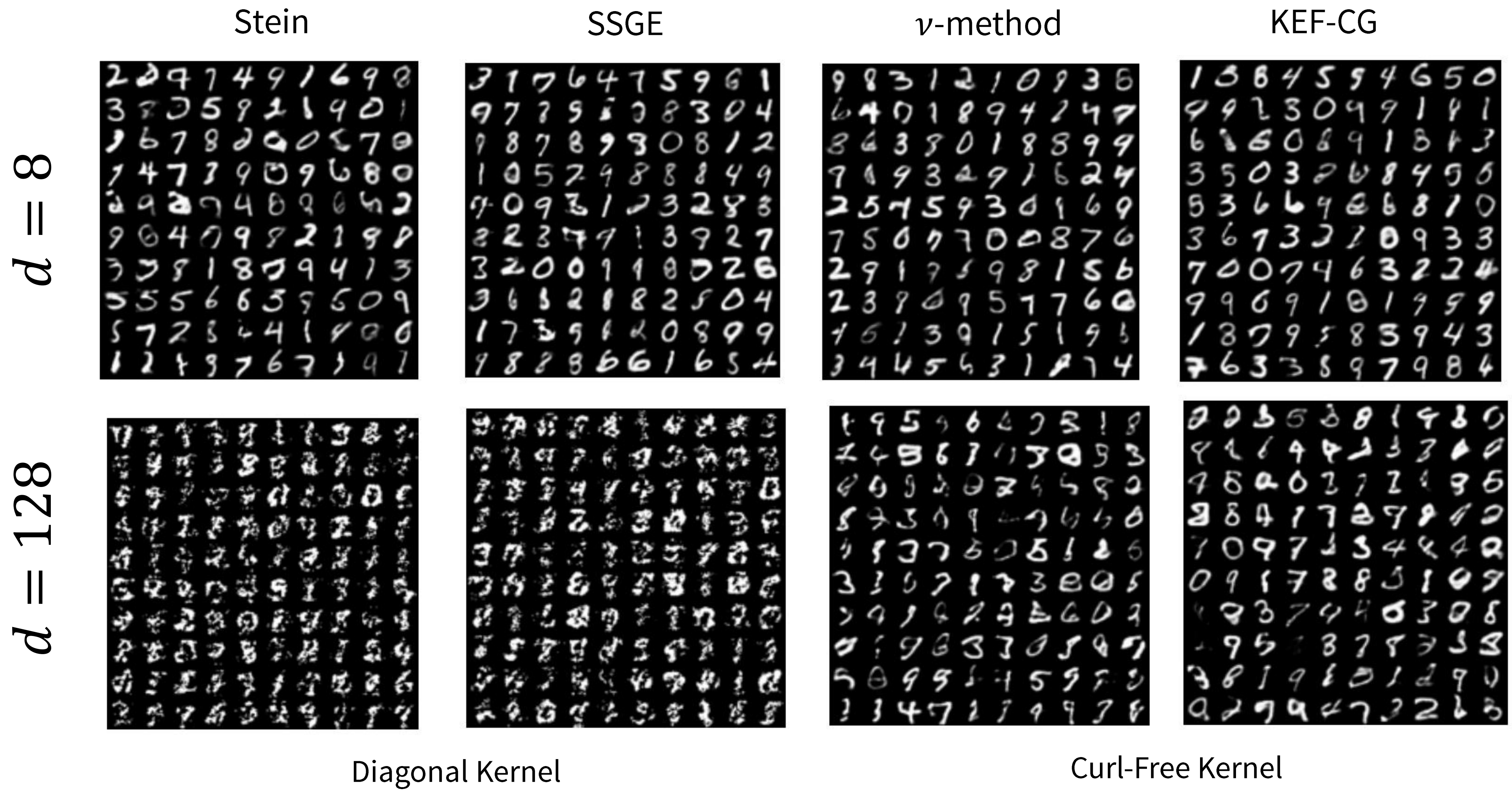
On MNIST

Table 3. Negative log-likelihoods on the MNIST dataset and per epoch time on 128 latent dimension.

	LATENT DIM	8	32	64	128	TIME	
Existing	STEIN	97.15 ± 0.14	92.10 ± 0.07	101.60 ± 0.44	114.41 ± 0.25	4.2s	Diagonal Kernel
	SSGE	97.24 ± 0.07	92.24 ± 0.17	101.92 ± 0.08	114.57 ± 0.23	9.2s	
	KEF	97.07 ± 0.03	90.93 ± 0.23	91.58 ± 0.03	92.40 ± 0.34	201.1s	Curl-free Kernel
	NKEF_2	97.71 ± 0.24	92.29 ± 0.41	92.82 ± 0.18	94.14 ± 0.69	36.4s	
	NKEF_4	97.59 ± 0.15	91.19 ± 0.08	91.80 ± 0.12	92.94 ± 0.58	97.5s	
	NKEF_8	97.23 ± 0.06	90.86 ± 0.09	92.39 ± 1.32	92.49 ± 0.41	301.2s	
Ours	KEF-CG	97.39 ± 0.22	90.77 ± 0.12	92.66 ± 0.67	$\mathbf{92.05} \pm 0.06$	13.7s	Curl-free Kernel
	ν-METHOD	97.28 ± 0.17	90.94 ± 0.02	$\mathbf{91.48} \pm 0.09$	92.10 ± 0.06	78.1s	
	SSM → Parametric	$\mathbf{96.98} \pm 0.27$	$\mathbf{89.06} \pm 0.01$	93.06 ± 0.68	96.92 ± 0.08	6.0s	Neural Network

Wasserstein Autoencoders

On MNIST



Wasserstein Autoencoders

On CelebA

Table 4. Fréchet Inception Distances on the CelebA dataset and per epoch time on 128 latent dimension.

LATENT DIM	8	32	64	128	TIME
STEIN	73.85 ± 1.39	58.29 ± 0.46	57.54 ± 0.57	76.31 ± 1.33	164.4s
SSGE	72.49 ± 1.09	58.01 ± 0.60	58.39 ± 1.00	76.85 ± 1.12	172.2s
NKEF_2	75.12 ± 1.55	53.92 ± 0.29	51.16 ± 0.30	55.17 ± 0.43	244.7s
NKEF_4	73.15 ± 0.77	54.54 ± 1.02	50.76 ± 0.19	53.70 ± 0.10	412.5s
KEF-CG	72.92 ± 0.60	54.32 ± 0.31	50.44 ± 0.20	$\mathbf{50.66} \pm 0.89$	166.2s
ν-METHOD	72.02 ± 1.22	52.86 ± 0.20	$\mathbf{50.16} \pm 0.23$	52.80 ± 0.43	220.9s
SSM	$\mathbf{69.72} \pm 0.25$	$\mathbf{49.93} \pm 0.74$	72.68 ± 1.75	94.07 ± 3.57	163.3s

Wasserstein Autoencoders

On CelebA

A Library of Kernel Score Estimators

An example

- Kernel score estimators contains too many formulas
- We provide a library of them in https://github.com/miskcoo/kscore
- A simple example using Tikhonov regularization + curl-free kernels

Change these for other methods

```
estimator = Tikhonov(lam=0.0001, kernel=kernels.CurlFreeIMQ)
estimator.fit(samples, kernel_hyperparams=kernel_width)
estimator.compute_gradients(x)
```

References

- Bauer, F., Pereverzev, S., and Rosasco, L. On regularization algorithms in learning theory. Journal of complexity, 23 (1):52–72, 2007.
- Canu, S. and Smola, A. Kernel methods and the exponential family. Neurocomputing, 69(7-9):714–720, 2006.
- Engl, H. W., Hanke, M., and Neubauer, A. Regularization of inverse problems, volume 375. Springer Science & Business Media, 1996.
- Fukumizu, K. Exponential manifold by reproducing kernel hilbert spaces. Algebraic and Geometric mothods in statistics, pp. 291–306, 2009.
- Hyvarinen, A. Estimation of non-normalized statistical models by score matching. Journal of Machine Learning Research, 6(Apr):695–709, 2005.
- Li, Y. and Turner, R. E. Gradient estimators for implicit models. In International Conference on Learning Representations, 2018.
- Macedo, I. and Castro, R. Learning divergence-free and curl-free vector fields with matrix-valued kernels. IMPA, 2010.
- Shi, J., Sun, S., and Zhu, J. A spectral approach to gradient estimation for implicit distributions. In International Conference on Machine Learning, pp. 4651–4660, 2018.

References

- Smale, S. and Zhou, D.-X. Learning theory estimates via integral operators and their approximations. Constructive approximation, 26(2):153–172, 2007.
- Song, Y. and Ermon, S. Generative modeling by estimating gradients of the data distribution. In Advances in Neural Information Processing Systems, pp. 11895–11907, 2019.
- Song, Y., Garg, S., Shi, J., and Ermon, S. Sliced score matching: A scalable approach to density and score estimation. arXiv preprint arXiv:1905.07088, 2019.
- Sriperumbudur, B., Fukumizu, K., Gretton, A., Hyv¨arinen, A., and Kumar, R. Density estimation in infinite dimensional exponential families. The Journal of Machine Learning Research, 18(1):1830–1888, 2017.
- Sutherland, D., Strathmann, H., Arbel, M., and Gretton, A. Efficient and principled score estimation with nystr¨om kernel exponential families. In International Conference on Artificial Intelligence and Statistics, pp. 652–660, 2018.
- Tolstikhin, I., Bousquet, O., Gelly, S., and Schoelkopf, B. Wasserstein auto-encoders. arXiv preprint arXiv:1711.01558, 2017.
- Wen, L., Zhou, Y., He, L., Zhou, M., and Xu, Z. Mutual information gradient estimation for representation learning. In International Conference on Learning Representations, 2020.
- Williams, C. K. and Seeger, M. Using the nystr¨om method to speed up kernel machines. In Advances in Neural Information Processing Systems, pp. 682–688, 2001.