Normalizing Flows for data with (and without) Translational and Rotational Symmetry

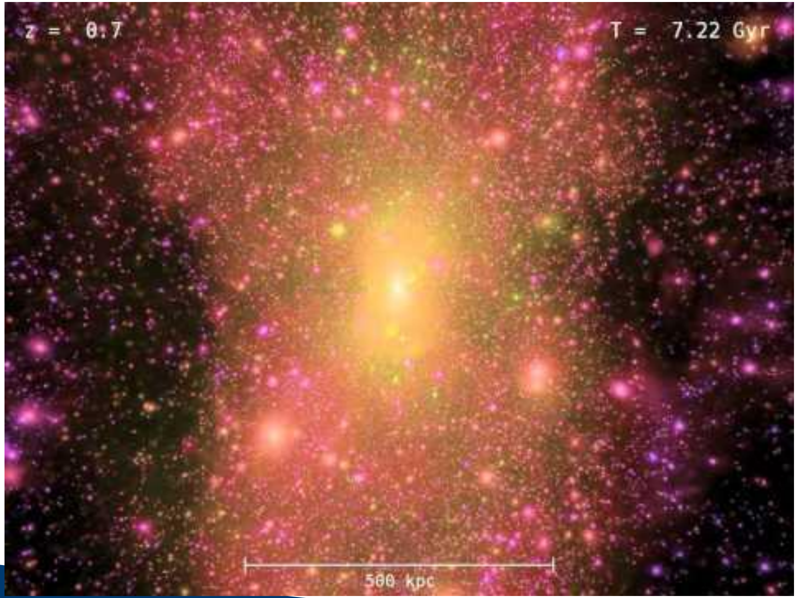
Biwei Dai
Advisor: Uroš Seljak
March 17, 2021

Berkeley
UNIVERSITY OF CALIFORNIA

Background

- Numerical simulations are crucial for accurately predicting cosmological observations and understanding the systematic effects in the current and future sky surveys, but they are computationally expensive.

http://www.sdss3.org/press/dr9.php



https://www.youtube.com/watch?v=xfgDoExbu_Q

UNIVERSITY OF CALIFORNIA

Motivations

- Deep generative models have been used in modeling structure formation
  - Predict tSZ signal, HI, dark matter annihilation feedback, super resolution, etc.

- Why not GANs?
  - Pro: Realistic looking samples
  - Con: Mode collapse. Low-dimensional manifold assumption leads to information loss. Potentially introduce systematics in data analysis.

Full dimensional models like **NF**s

- Physical constraints: translational and rotational symmetry
  - What people do now: let the model learn the symmetry by augmenting the training data. Performance not guaranteed. Increase the training work.

Enforce the symmetry in the generative model

Berkeley
UNIVERSITY OF CALIFORNIA

Normalizing Flows

- Bijective mapping f between data x and latent variable z ($z = f(x)$, $z \sim \pi(z)$)
  - Sample: $x = f^{-1}(z)$ ($z \sim \pi(z)$)
  - Evaluate density: $p(x) = \pi(f(x))\, |\det(df/dx)|$
  - $f = f_1 \circ f_2 \circ \ldots \circ f_n$ usually parametrizes with NNs. f should be easy to invert, Jacobian determinant $|\det(df/dx)|$ should be easy to evaluate.
  - Training objective: $\langle \log p(x) \rangle$

Berkeley
UNIVERSITY OF CALIFORNIA

NFs with translational and rotational symmetry

1D function. Model with **cubic splines**

- Convolutional neural networks in Fourier space:
  - Convolution: $\int T(\mathbf{r}-\mathbf{r}')x(\mathbf{r}')d\mathbf{r}' \xrightarrow{\text{FT}} T(\mathbf{k})x(\mathbf{k}) \xrightarrow{\text{rotational symmetry}} T(k)x(\mathbf{k})$
  - Nonlinearity: any monotonic, differentiable functions Ψ

1D function. Model with **rational quadratic splines**, which is continuously differentiable and analytically invertible (Gregory & Delbourgo, 1986)

- Each layer: $\mathbf{f_i(x) = \Psi_i\,(F^{-1}\,T_i(k)\,F\,x)}$
  - Translational and rotational invariant
  - Inverse transform: $f^{-1}_i(x) = F^{-1}\,T_i^{-1}(k)\,F\,\Psi_i^{-1}(x)$
  - Jacobian determinant: $\det(df_i/dx) = d\Psi_i/dx\,\mathbf{\Pi}_k T_i(k)$
- Deep model: $f = f_1 \circ f_2 \circ \ldots \circ f_n$

Berkeley
UNIVERSITY OF CALIFORNIA

NFs with translational and rotational symmetry

- Enforcing translational and rotational symmetry
  - No data augmentation.
  - Fewer parameters: $N_{layer}*(3*N_{\Psi\_knot}+2*N_{T\_knot}) \sim$ O(1000) for d ~ 10^4
  - Faster training.
  - Less overfitting. Better generalization.

- Normalizing Flow structure:
  - Efficient sampling and density estimation.
  - logp training. Better coverage of the data.

Berkeley
UNIVERSITY OF CALIFORNIA

Conditional NFs

- Build generative models conditional on physical parameters y (cosmological parameters, baryon physics parameters, redshift, noise, calibration parameters, etc.)
    - $f(x) = f_y(x)$: $T(x) = T_y(x)$, $\boldsymbol{\Psi}(x) = \boldsymbol{\Psi}_y(x)$
    - Model the conditional dependence: $y \xrightarrow{\text{NN}}$ the knots of T_y and $\boldsymbol{\Psi}_y$
    - Training objective: log-likelihood log p(x|y) $\longrightarrow$ **posterior distribution!**

with non-gaussian information

- $$X \xrightarrow[T_1,\ \boldsymbol{\Psi}_1]{\Psi_1 (F^{-1} T_1(k) F X)} X_1 \xrightarrow[T_2,\ \boldsymbol{\Psi}_2]{\Psi_2 (F^{-1} T_2(k) F X_1)} X_2 \xrightarrow[T_3,\ \boldsymbol{\Psi}_3]{\Psi_3 (F^{-1} T_3(k) F X_2)} \ldots\ldots \xrightarrow[T_N,\ \boldsymbol{\Psi}_N]{\Psi_N (F^{-1} T_N(k) F X_{N-1})} Z$$

$$y \xrightarrow{\text{NN}} T_1, \boldsymbol{\Psi}_1 \qquad y \xrightarrow{\text{NN}} T_2, \boldsymbol{\Psi}_2 \qquad y \xrightarrow{\text{NN}} T_3, \boldsymbol{\Psi}_3 \qquad y \xrightarrow{\text{NN}} T_N, \boldsymbol{\Psi}_N$$

Berkeley
UNIVERSITY OF CALIFORNIA

Experiments

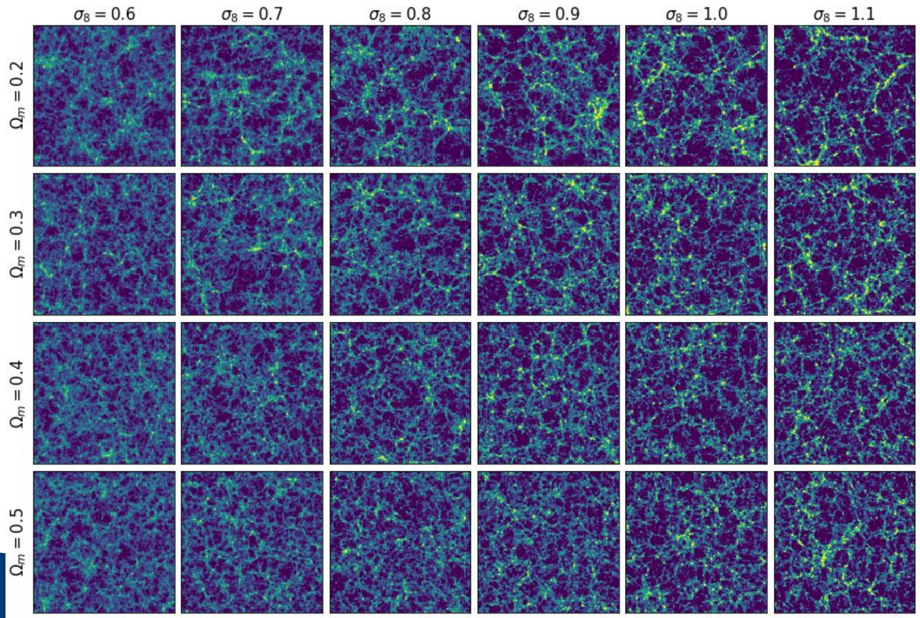
- Data: 2D dark matter overdensity field at redshift 0
  - 512 h^{-1}Mpc Box, 128 h^{-1}Mpc slice, 128 x 128 mesh.
  - Vary cosmology parameters $\Omega_m \in [0.2, 0.5]$ and $\sigma_8 \in [0.5, 1.1]$ (conditional variable y).
  - Generated by N-body solver FastPM.

- Model: convolutional rotation/translation invariant normalizing flow
  - $N_{layer} = 5$, $N_{\Psi\_knot} = 8$, $N_{T\_knot} = 128$
  - Conditional NN: MLP with 2 hidden layers of $N_{neuron} = 512$

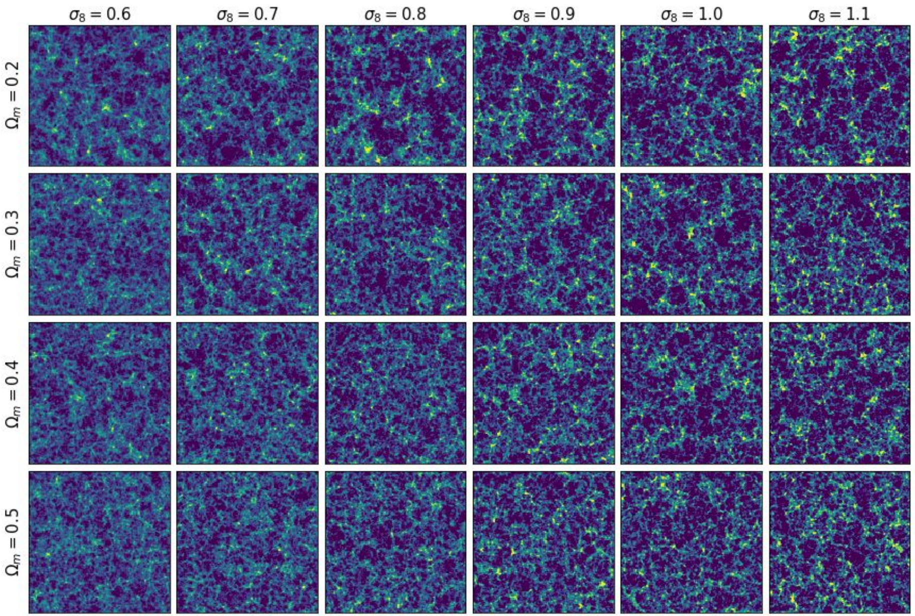
Berkeley
UNIVERSITY OF CALIFORNIA

Preliminary Results -- Samples

- data:



- sample:

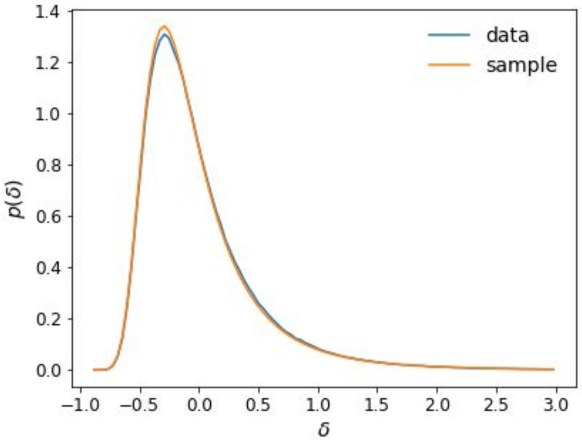


more clustering

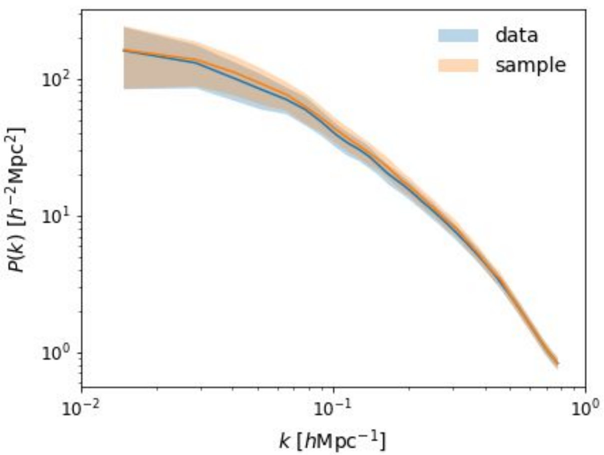
Berkeley UNIVERSITY OF CALIFORNIA

Preliminary Results -- Samples

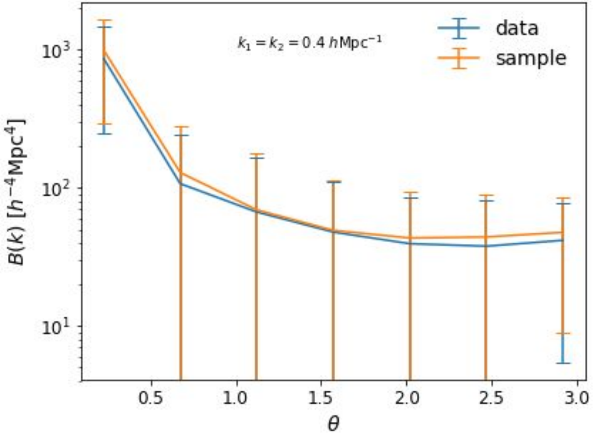
- 1-D probability distribution function



- power spectrum:



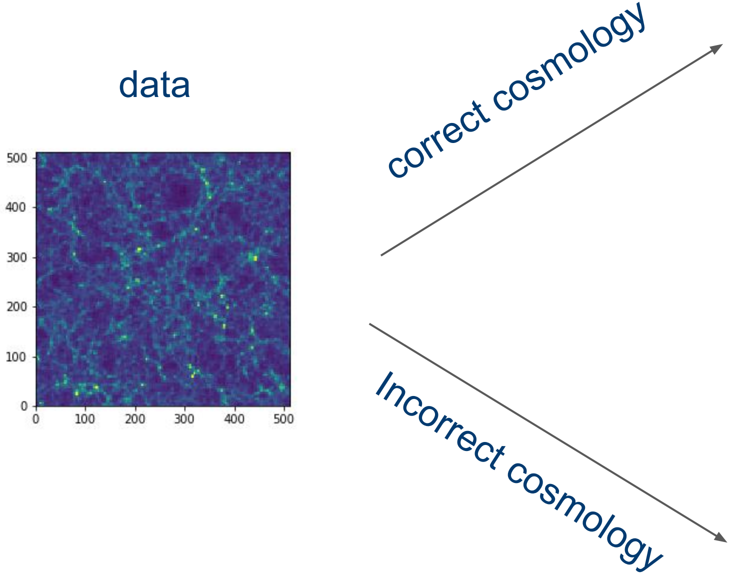
- bispectrum:

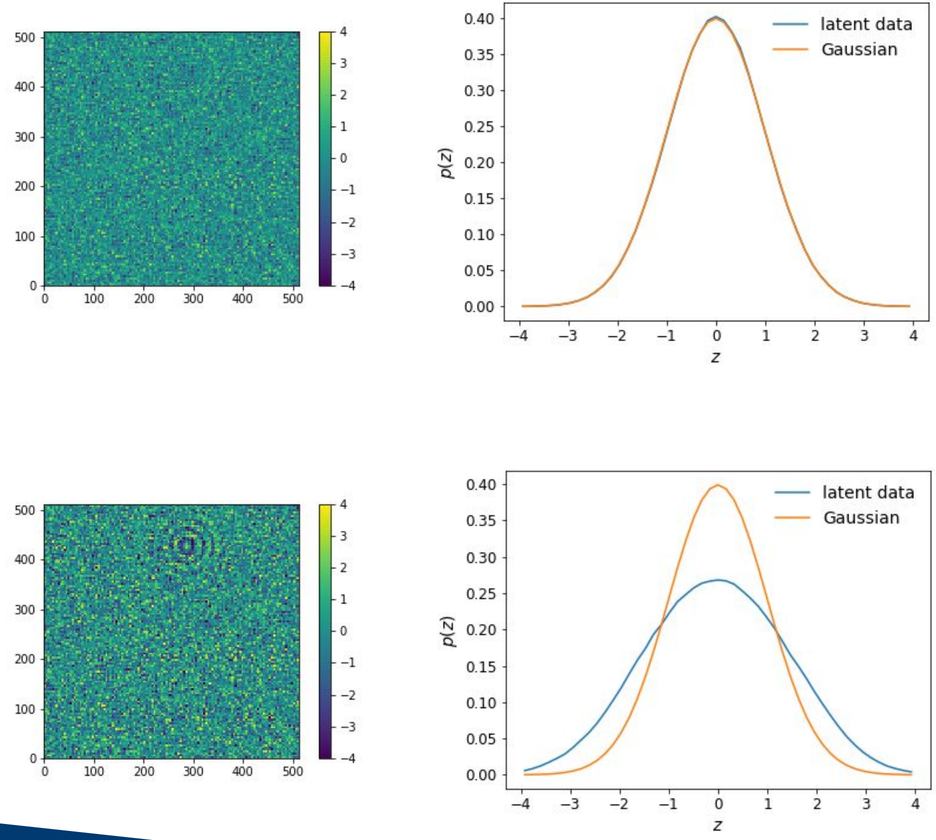


Measured over 10000 samples

Berkeley
UNIVERSITY OF CALIFORNIA

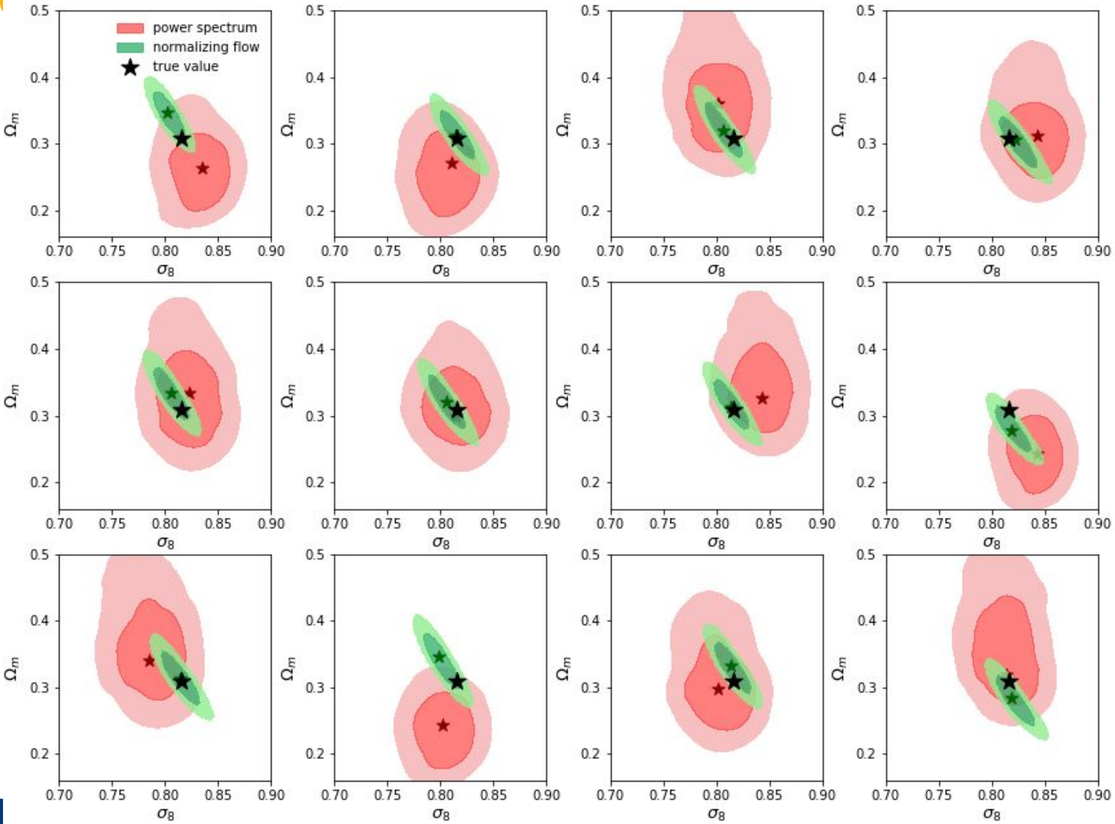
Preliminary Results -- Latent variables





Preliminary Results – Posterior analysis

- This approach learns likelihood p(x|y) directly: full posterior distribution via p(y|x)=p(x|y)p(y)/p(x)
- Aims to capture all the information of the data ⟶ **optimal posterior analysis**
- This is very different from Likelihood Free Inference or ABC, which use suboptimal summary statistics S and build their p(y|S) or p(S|y)
- Figure of merit (inverse of the area of the 68% confidence region):
  - power spectrum: ~ 176
  - normalizing flow: ~ 1065



Discussion

- Current and future works:
  - Apply the model to other (noisy) observables (weak lensing, galaxies, tSZ, etc.). Adding translational / rotational invariant noise is easy (train with noise).
  - Use this model to learn the relation / mapping between baryon and DM.
  - Explore training with $p(y|x)$.

- Challenges: the translational and rotational symmetry is broken
  - Redshift space distortion: $T(k) \longrightarrow T(k_r, k_z)$
  - Survey window function: adding non convolutional layers (e.g. add SINF)

Berkeley
UNIVERSITY OF CALIFORNIA

Conclusions

- We introduce a NF model that explicitly satisfies the **translational and rotational symmetry**. These symmetries reduce the degrees of freedom and enable the model to scale well to high dimensions.

- Compared to other generative models like GANs, this model is **bijective (no information loss), is easier and faster to train, has better coverage of the data, and allows exact density (likelihood) evaluation p(x|y)**.

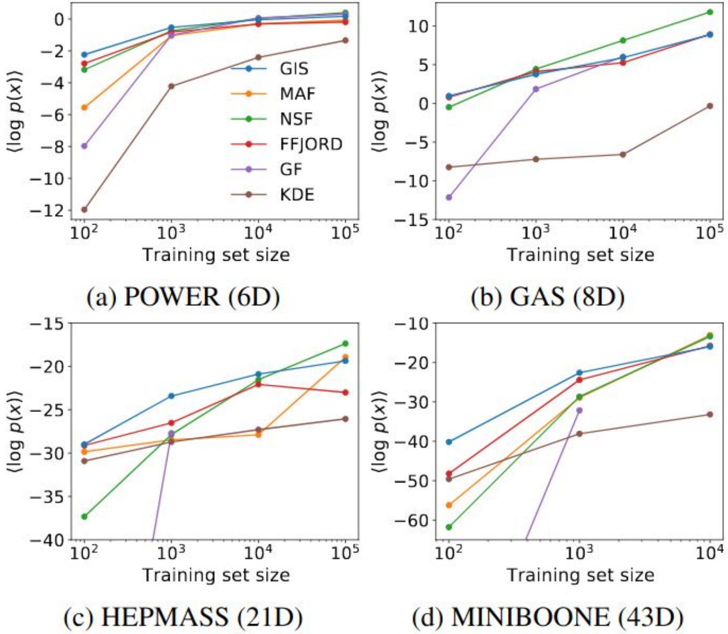
- We apply the model for generating DM density field conditional on Ω_m and σ_8. We show it is able to generate **realistic samples** that agree well with the data on various summary statistics, and the data is visually mapped to a Gaussian. We show that its likelihood gives **much tighter constraints on cosmological parameters** than standard power spectrum analysis. If NF maps to a perfect Gaussian then p(x|y) enables **optimal posterior analysis** of cosmological data.

Sliced Iterative Normalizing Flows (SINF)

- This is for data without translational/rotational invariance
- Iteratively building the NF based on optimal transport on 1D slices
- GIS trains in Gaussian space: achieves **the best density estimation results on small training sets**.
- One line code:

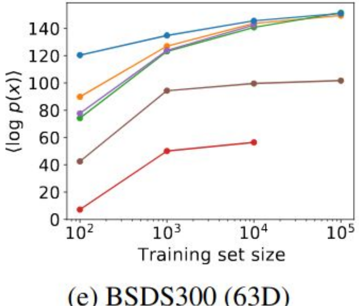
```
model = GIS(data_train, data_validate)

logp_test = model.evaluate_density(data_test)
sample, logp_sample = model.sample(nsample=100)
```



(a) POWER (6D) (b) GAS (8D)
(c) HEPMASS (21D) (d) MINIBOONE (43D)

2007.00674
https://github.com/biweidai/SIG_GIS



(e) BSDS300 (63D)

Sliced Iterative Normalizing Flows (SINF)

- (SIG) trains in data space: allows directly optimizing the distribution of samples

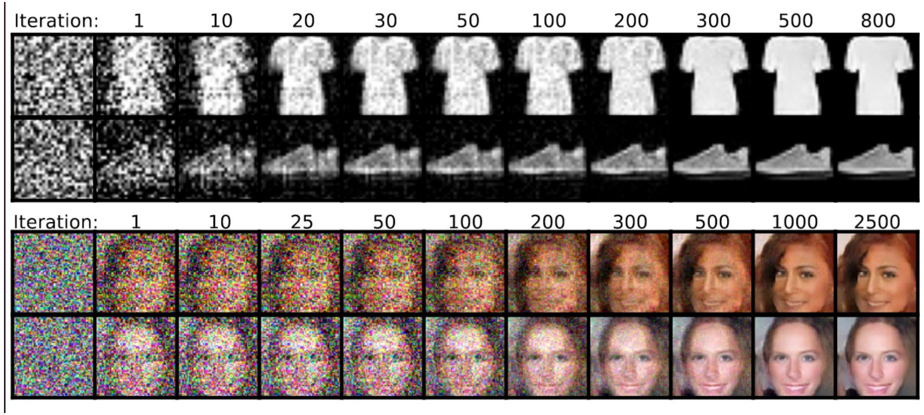


Table 2. FID scores on different datasets (lower is better). The errors are generally smaller than the differences.

	Method	MNIST	Fashion	CIFAR-10	CelebA
iterative	SWF	225.1	207.6	-	-
	SIG ($T = 1$) (this work)	**4.5**	**13.7**	66.5	37.3
adversarial training	Flow-GAN (ADV)	155.6	216.9	71.1	-
	WGAN	6.7	21.5	**55.2**	41.3
	WGAN GP	20.3	24.5	55.8	**30.0**
	Best default GAN	~ 10	~ 32	~ 70	~ 48
AE based	SWAE	29.8	74.3	141.9	53.9
	CWAE	23.6	57.1	120.0	49.7
	PAE	-	28.0	-	49.2
	two-stage VAE	12.6	29.3	96.1	44.4
	Best default VAE	16.6	43.6	-	53.3

2007.00674
https://github.com/biweidai/SIG_GIS