Neural Inverse Transform Sampler

Henry Li[1], Yuval Kluger[1]

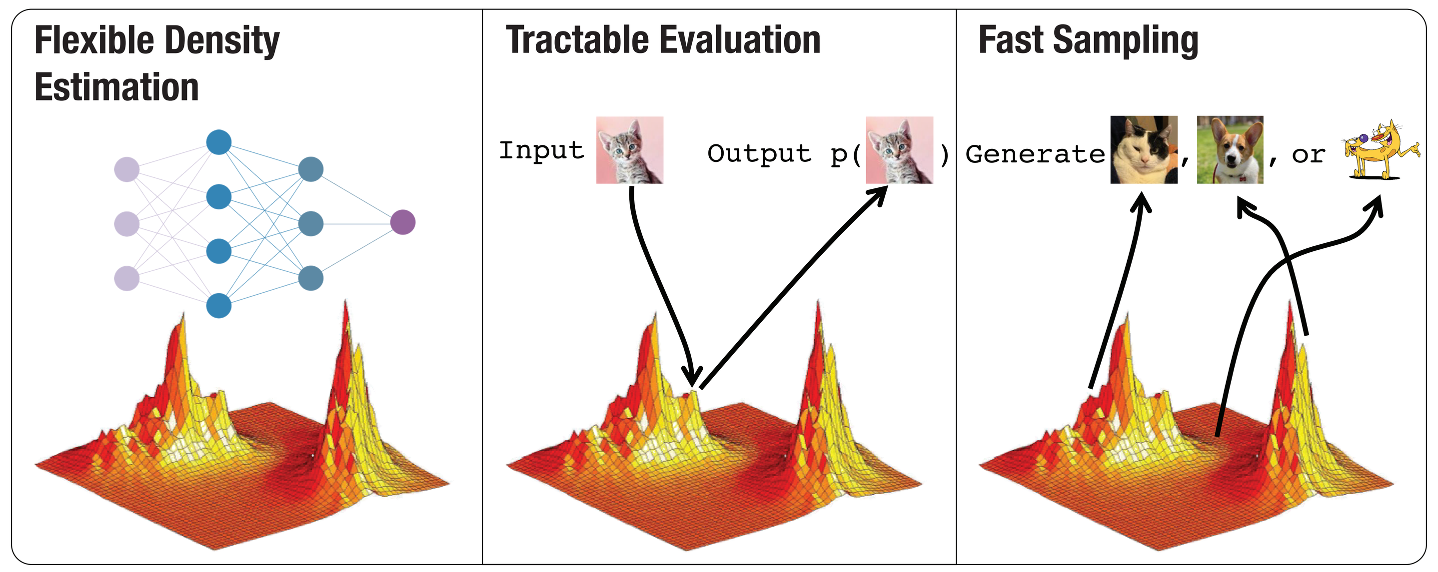
[1]Applied Mathematics Program, Yale University

ICML International Conference On Machine Learning

Yale

Desiderata

We consider the following attributes in an ideal density estimator:



Problem

A fundamental obstacle in density estimation is the trade-off between tractability and flexibility of the density function $p(x)$. For example...

Energy Based Models	Gaussian Mixture Models	Normalizing Flows
$p(x) = e^{-f_\theta(x)}/Z_{f_\theta}$	$p(x) = \sum_{i=1}^{k} \alpha_i p_{\mathcal{N}(\mu_i,\sigma_i)}(x)$	$p(x) = \mu(f_\theta^{-1}(x))\lvert J_{f_\theta}(x)\rvert^{-1}$
can have arbitrarily powerful f_θ, but require estimation of the normalizing constant Z_{f_θ}, which usually requires numerical integration.	have analytically computable normalizing constants, but few degrees of freedom.	sidestep the normalizing constant entirely, but require judicious choice of f_θ so that f_θ^{-1} and $\lvert J_{f_\theta}(x)\rvert^{-1}$ are tractable.

Solution

What if we can compute the normalizing constant analytically, for arbitrary f_θ? Recall the Fundamental Theorem of Calculus: If there exists F_θ such that

$$\frac{dF_\theta}{dx} = f_\theta \text{ for all } x \in [A, B],$$

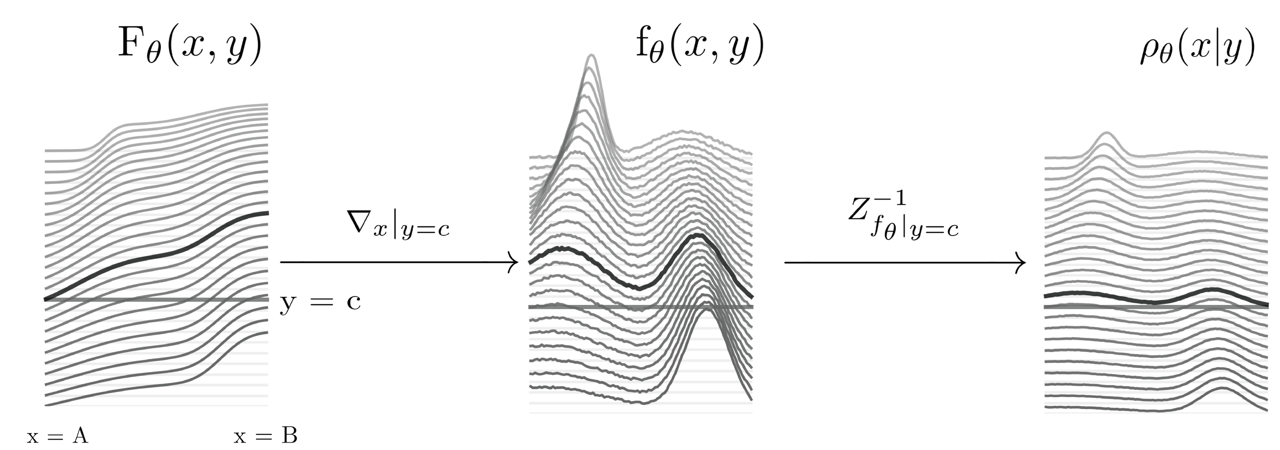
then

$$\int_A^B f_\theta(x)dx = F_\theta(B) - F_\theta(A).$$

This basic strategy can be extended to higher dimensions via the Gradient Theorem. Therefore, by representing F_θ as a a neural network, the above condition is always fulfilled, and so we retain the flexibility of an arbitrarily powerful f_θ while retaining the tractability of Z_{f_θ}.

Our Method

We call the resulting network a Probabilistically Normalized Network (PNN), which can model arbitrary continuous, compactly supported conditional densities $\rho_\theta(x|y)$.



Above: An illustration of a two-dimensional PNN. To compute $\rho_\theta(x|y)$, we first differentiate w.r.t. x while holding y constant. Then, we divide by $F_\theta|_{y=c}$ evaluated at the boundaries $x = A$ and $x = B$.

By decomposing n-dimensional densities autoregressively via the probabilistic chain rule, we can model arbitrary densities:

$$\rho_\theta(x_1, x_2, \dots, x_n) = \prod_{i=1}^{n} \rho_\theta(x_i|x_{<i}).$$

Since the normalized network F_θ/Z_{f_θ} represents the cumulative distribution function of each conditional density, we can easily invert it via bisection search, and sample from each conditional density via the Inverse Transform Method:

1. sample $z \sim \text{Uniform}[0, 1]$,
2. compute $x = (F_\theta/Z_{f_\theta})^{-1}(z)$,

where x is now distributed as the desired density.

NITS is a Universal Density Estimator

The resulting estimator can universally approximate any continuous autoregressive random variable with compact support:

Corollary 1. *Let $\rho(x)$ be a general joint density for a d-dimensional autoregressive random variable, i.e. takes on the form*

$$\rho(x) = \rho(x_d|x_{d-1}, \dots, x_1) \dots \rho(x_1).$$

Then there exists a set of PNNs $\{F_{\theta_i}\}_{i=1}^d$ that induce a ρ_θ such that for any $\epsilon > 0$,

$$||\rho_\theta(x) - \rho(x)||_1 < \epsilon.$$

Empirical Results

NITS achieves state-of-the-art performance on density estimation tasks on tabular data, among normalizing flow-based density estimators.

Model	POWER	GAS	HEPMASS	MINIBOONE	BSDS300
MAF	0.30 ± 0.01	9.59 ± 0.02	-17.39 ± 0.02	-11.68 ± 0.44	156.36 ± 0.28
TAN	0.48 ± 0.01	11.19 ± 0.02	-15.12 ± 0.02	-11.01 ± 0.48	157.03 ± 0.07
NAF	0.62 ± 0.02	11.91 ± 0.13	-15.09 ± 0.40	$\mathbf{-8.86 \pm 0.15}$	157.73 ± 0.04
B-NAF	0.61 ± 0.01	12.06 ± 0.02	-14.71 ± 0.02	-8.95 ± 0.07	157.36 ± 0.03
FFJORD	0.46 ± 0.01	8.59 ± 0.12	-14.92 ± 0.08	-10.43 ± 0.04	157.40 ± 0.19
SOS	0.60 ± 0.01	11.99 ± 0.41	-15.15 ± 0.10	-8.90 ± 0.11	157.48 ± 0.41
NSF	$\mathbf{0.66 \pm 0.01}$	13.09 ± 0.02	-14.01 ± 0.03	-9.22 ± 0.48	157.31 ± 0.28
RealNVP	0.17 ± 0.01	8.33 ± 0.14	-18.71 ± 0.02	-13.84 ± 0.52	153.28 ± 1.78
MADE MoG	0.40 ± 0.01	8.47 ± 0.02	-15.15 ± 0.02	-12.27 ± 0.47	153.71 ± 0.28
NITS-MLP (ours)	$\mathbf{0.66 \pm 0.01}$	$\mathbf{13.20 \pm 0.01}$	$\mathbf{-12.93 \pm 0.02}$	-10.85 ± 0.02	155.91 ± 0.21
NITS-CONV (ours)	-	-	-	-	$\mathbf{163.35 \pm 0.22}$

NITS also performs favorably in a generative modeling setting with images, when compared against normalizing flow-based models and autoregressive models.

Model	CIFAR-10
Pixel CNN	3.14
Gated Pixel CNN	3.03
Row Pixel RNN	3.00
Pixel CNN++	2.92
Image Transformer	2.90
PixelSNAIL	2.85
DISCRETE NITS-CONV (ours)	2.94
RealNVP	3.49
Glow	3.35
Flow++	3.08
NITS-CONV (ours)	2.97

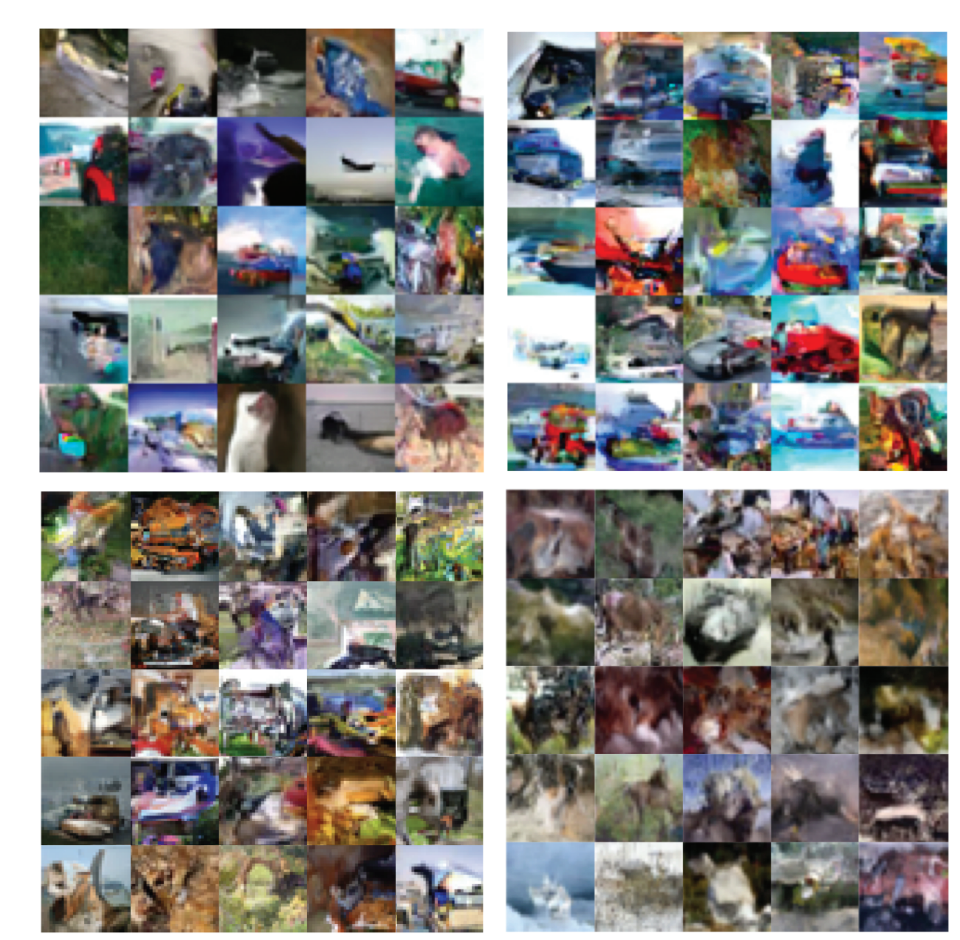


Figure 2. Randomly generated images from DISCRETE NITS-CONV (top left) and NITS-CONV (top right). Compare with competing discretized and continuous density models, Pixel CNN (bottom left) and Flow++ (bottom right), respectively.