Residual Flows

for Invertible Generative Modeling

Ricky T. Q. Chen, Jens Behrmann,
David Duvenaud, Jörn-Henrik Jacobsen

Universität Bremen
UNIVERSITY OF TORONTO
VECTOR INSTITUTE

Invertible Residual Networks (i-ResNet)

It can be shown that residual blocks

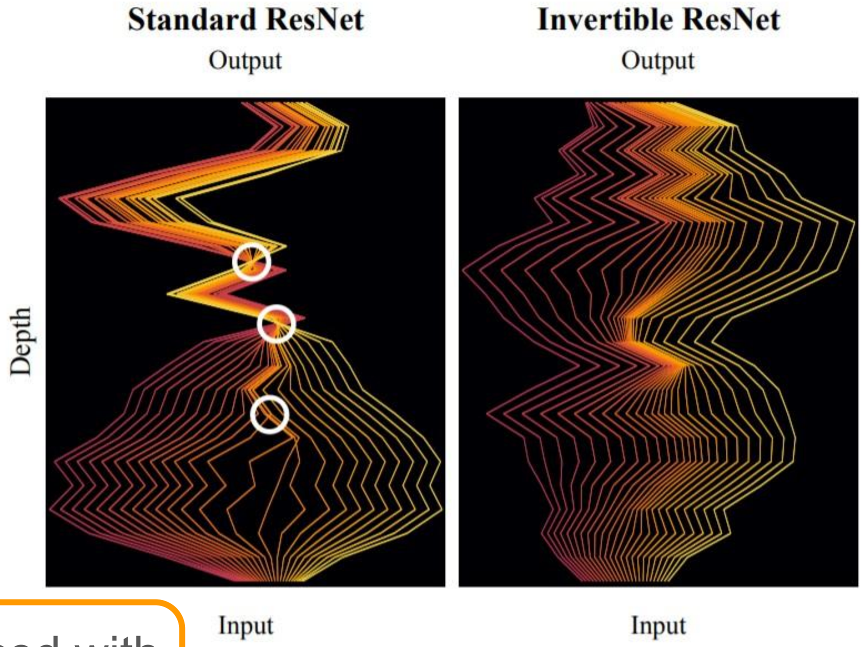
$$y = f(x) = x + g(x)$$

can be inverted by fixed-point iteration

$$x^{(i)} = y - g(x^{(i-1)})$$

and has a unique inverse (ie. invertible)

if $$|g(x) - g(y)| < |x - y|$$



i.e. Lipschitz. Enforced with **spectral normalization.**

(Behrmann et al. 2019)

Applying Change of Variables to i-ResNets

If

$$y = f(x) = x + g(x)$$

Then

$$\log p(x) = \log p(f(x)) + \log \left| \det \frac{df(x)}{dx} \right|$$

$$\log p(x) = \log p(f(x)) + \sum_{i=1}^{\infty} \frac{(-1)^{k+1}}{k} \operatorname{tr}([J_g(x)]^k)$$

(Behrmann et al. 2019)

Unbiased Estimation of Log Probability Density

Enter the “Russian roulette” estimator (Kahn, 1955). Suppose we want to estimate

$$\sum_{k=1}^{\infty} \Delta_k$$

(Require $\sum_{k=1}^{\infty} |\Delta_k| < \infty$)

Unbiased Estimation of Log Probability Density

Enter the “Russian roulette” estimator (Kahn, 1955). Suppose we want to estimate

$$\sum_{k=1}^{\infty} \Delta_k$$

(Require $\sum_{k=1}^{\infty} |\Delta_k| < \infty$)

Flip a coin b with probability ***q***.

$$\mathbb{E}\left[\Delta_1 + \qquad\qquad\qquad\qquad\qquad\qquad \right]$$

Unbiased Estimation of Log Probability Density

Enter the "Russian roulette" estimator (Kahn, 1955). Suppose we want to estimate

$$\sum_{k=1}^{\infty} \Delta_k$$

(Require $\sum_{k=1}^{\infty} |\Delta_k| < \infty$)

Flip a coin b with probability ***q***.

$$\mathbb{E}\left[\Delta_1 + \left[\qquad \right] \mathbb{1}_{b=0} + [\ ] \mathbb{1}_{b=1}\right]$$

Unbiased Estimation of Log Probability Density

Enter the “Russian roulette” estimator (Kahn, 1955). Suppose we want to estimate

$$\sum_{k=1}^{\infty} \Delta_k$$

(Require $\sum_{k=1}^{\infty} |\Delta_k| < \infty$)

Flip a coin b with probability ***q***.

$$\mathbb{E}\left[\Delta_1 + \left[\frac{1}{1-q}\sum_{k=2}^{\infty} \Delta_k\right] \mathbb{1}_{b=0} + [0]\mathbb{1}_{b=1}\right]$$

Unbiased Estimation of Log Probability Density

Enter the "Russian roulette" estimator (Kahn, 1955). Suppose we want to estimate

$$\sum_{k=1}^{\infty} \Delta_k$$

(Require $\sum_{k=1}^{\infty} |\Delta_k| < \infty$)

Flip a coin b with probability ***q***.

$$\mathbb{E}\left[\Delta_1 + \left[\frac{1}{1-q}\sum_{k=2}^{\infty} \Delta_k\right]\mathbb{1}_{b=0} + [0]\mathbb{1}_{b=1}\right]$$

$$= \Delta_1 + \left[\frac{1}{1-q}\sum_{k=2}^{\infty} \Delta_k\right](1-q)$$

$$= \sum_{k=1}^{\infty} \Delta_k$$

Has probability q of being evaluated in **finite** time.

Unbiased Estimation of Log Probability Density

If we repeatedly apply the same procedure *infinitely many times*, we obtain an unbiased estimator of the infinite series.

$$\sum_{k=1}^{\infty} \Delta_k = \mathbb{E}_{n \sim p(N)} \left[\sum_{k=1}^{n} \frac{\Delta_k}{\mathbb{P}(N \geq k)} \right]$$

Computed in **finite** time with **prob. 1**!!

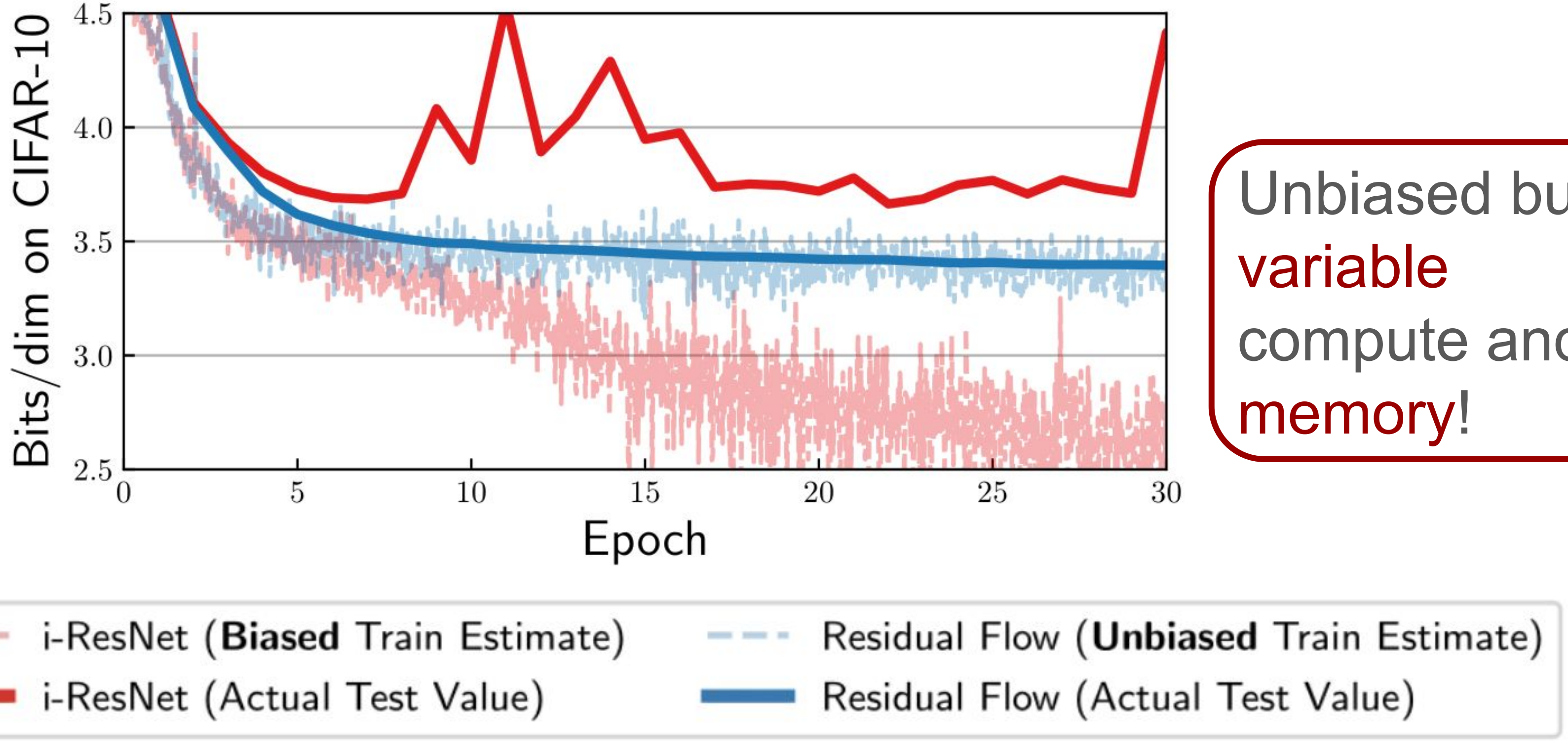
Directly sample the first successful coin toss.

k-th term is weighted by prob. of seeing >= k tosses.

Residual Flow:

$$\log p(x) = \log p(f(x)) + \mathbb{E}_{n,v} \left[\sum_{k=1}^{n} \frac{(-1)^{k+1}}{k} \frac{v^T [J_g(x)]^k v}{\mathbb{P}(N \geq k)} \right]$$

Decoupled Training Objective & Estimation Bias



Unbiased but... variable compute and memory!

Constant-Memory Backpropagation

Naive gradient computation:

$$\mathbb{E}_{n,v}\left[\sum_{k=1}^{n} \alpha_k \frac{\partial v^T [J_g(x)]^k v}{\partial \theta}\right]$$

Alternative (Neumann series) gradient formulation:

$$\mathbb{E}_{n,v}\left[\left(\sum_{k=1}^{n} \alpha_k v^T [J_g(x)]^k\right) \frac{\partial J_g(x) v}{\partial \theta}\right]$$

1. Estimate
2. Differentiate

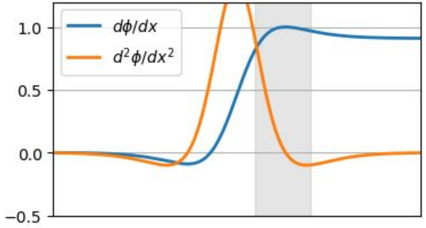
↓

1. Analytically Differentiate
2. Estimate

Don't need to store random number of terms in memory!!

Density Estimation Experiments

(LipSwish)



Contribution Summary:

- Unbiased estimator of log-likelihood.
- Memory-efficient computation of log-likelihood.
- LipSwish activation function [not discussed in talk].

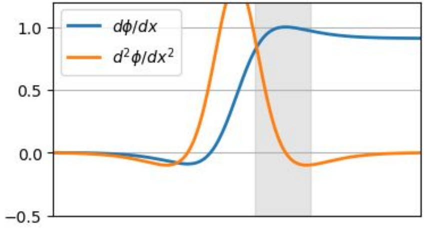
Model	MNIST	CIFAR-10	ImageNet 32	ImageNet 64	CelebA-HQ 256
Real NVP (Dinh et al., 2017)	1.06	3.49	4.28	3.98	—
Glow (Kingma and Dhariwal, 2018)	1.05	3.35	4.09	3.81	1.03
FFJORD (Grathwohl et al., 2019)	0.99	3.40	—	—	—
Flow++ (Ho et al., 2019)	—	3.29 (3.09)	— (3.86)	— (3.69)	—
i-ResNet (Behrmann et al., 2019)	1.05	3.45	—	—	—
Residual Flow (Ours)	**0.970**	**3.280**	**4.010**	**3.757**	**0.992**

Density Estimation Experiments

Contribution Summary:

- Unbiased estimator of log-likelihood.
- Memory-efficient computation of log-likelihood.
- LipSwish activation function [not discussed in talk].

(LipSwish)



Training Setting	MNIST	CIFAR-10	CIFAR-10†
i-ResNet + ELU	1.05	3.45	3.66∼4.78
Residual Flow + ELU	1.00	3.40	3.32
Residual Flow + LipSwish	**0.97**	**3.39**	**3.28**

Table: Ablation results. †Larger network.

Qualitative Samples

CelebA:

Data

Residual Flow

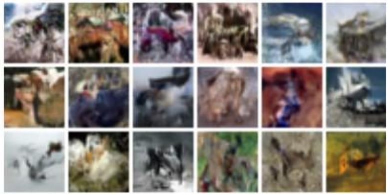
CIFAR10:

Data

Residual Flow

PixelCNN

Flow++



Model	CIFAR10 FID
PixelCNN*	65.93
PixelIQN*	49.46
i-ResNet	65.01
Residual Flow	**46.37**
DCGAN*	37.11
WGAN-GP*	36.40

Qualitative Samples

CelebA:

Data

Residual Flow

CelebA-HQ **256x256**:

Model	CIFAR10 FID
PixelCNN*	65.93
PixelIQN*	49.46
i-ResNet	65.01
Residual Flow	**46.37**
DCGAN*	37.11
WGAN-GP*	36.40

Thanks for Listening!

Code and pretrained models: https://github.com/rtqichen/residual-flows

Co-authors:

Jens Behrmann

David Duvenaud

Jörn-Henrik Jacobsen