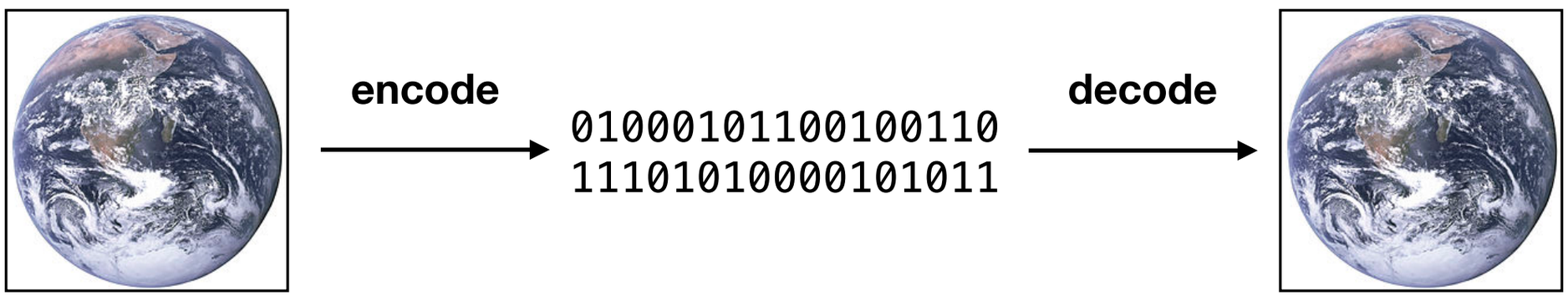
Compression with Flows via Local Bits-Back Coding

Jonathan Ho, Evan Lohn, Pieter Abbeel

Background

- Lossless compression with likelihood-based generative model p(**x**)



- Information theory: a uniquely decodable code **exists** with lengths

$$\approx -\log p(x)$$

  - Training (maximum likelihood) optimizes expected codelength
  - But what about **computational efficiency** of coding?

Existing compression algorithms

- Naive algorithm requires enumerating all data. Needs exponential resources in data dimension
- Must harness structure of p(**x**) to code efficiently
  - Autoregressive model: code one dimension at a time
  - **Latent variable models** trained with variational inference: **bits-back coding**

Flow models

- Flow model: smooth invertible map between noise and data

- They are likelihood-based, so coding algorithm must exist

- This work: computationally efficient coding for flows

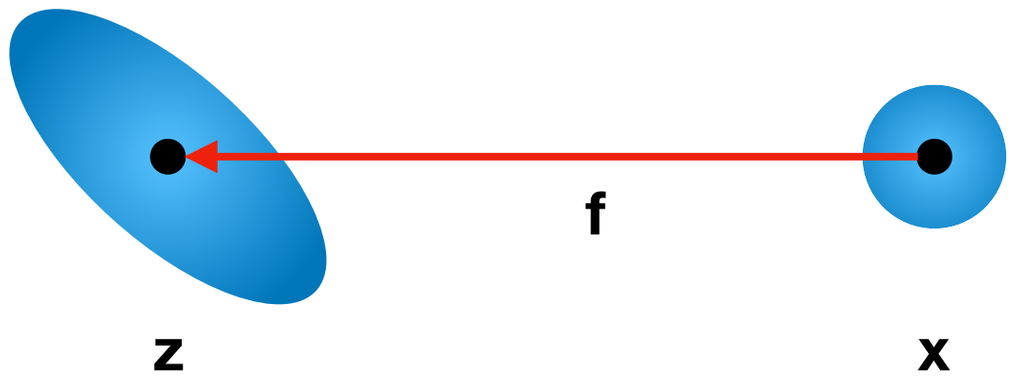
$$\mathbf{z} \sim \mathcal{N}(0, I)$$

Local approximations of flows

- Strategy for coding: **locally approximate the flow as a VAE**, then apply bits-back coding

- Flow model maps data to latent: **z** = f(**x**)

- Construct a VAE where f is q(**z**|**x**) and f^{-1} is p(**x**|**z**)



- The VAE bound will closely match the flow's log likelihood

Local bits-back coding

- Our algorithm is bits-back coding on this VAE approximation of the flow

- Straightforward implementation needs cubic time in data dimension. No assumptions on flow structure.

- Better than exponential, but not fast enough

Specializing local bits-back coding

- Making extra assumptions on the flow lets us speed up compression

- For RealNVP family: **linear time, fully parallelizable compression** by exploiting structure of coupling layers and composition

Results

- Implemented for Flow++, a RealNVP-type flow model

Compression algorithm	CIFAR10	ImageNet 32x32	ImageNet 64x64
Theoretical	3.116	3.871	3.701
Local bits-back (ours)	3.118	3.875	3.703

- State of the art fully parallelizable compression on these datasets
  - Requires “auxiliary bits” for bits-back coding
  - Codelength can degrade if auxiliary bits are unavailable

Results: speed

- Specializing local bits-back to the RealNVP structure speeds up compression by orders of magnitude

Algorithm	Batch size	CIFAR10	ImageNet 32x32	ImageNet 64x64
Black box (Algorithm 1)	1	64.37 ± 1.05	534.74 ± 5.91	1349.65 ± 2.30
Compositional (Section 3.4.3)	1	0.77 ± 0.01	0.93 ± 0.02	0.69 ± 0.02
	64	0.09 ± 0.00	0.17 ± 0.00	0.18 ± 0.00
Neural net only, without coding	1	0.50 ± 0.03	0.76 ± 0.00	0.44 ± 0.00
	64	0.04 ± 0.00	0.13 ± 0.00	0.05 ± 0.00

Conclusion

- Local bits-back coding: **compression with flow models**
  - Naive algorithm: exponential time in data dimension
  - Our algorithm for general flows: polynomial time
    - Our algorithm for RealNVP family: linear time and parallelizable
- For algorithm details and comparisons to other types of models, come to our poster!
- Open source: **github.com/hojonathanho/localbitsback**