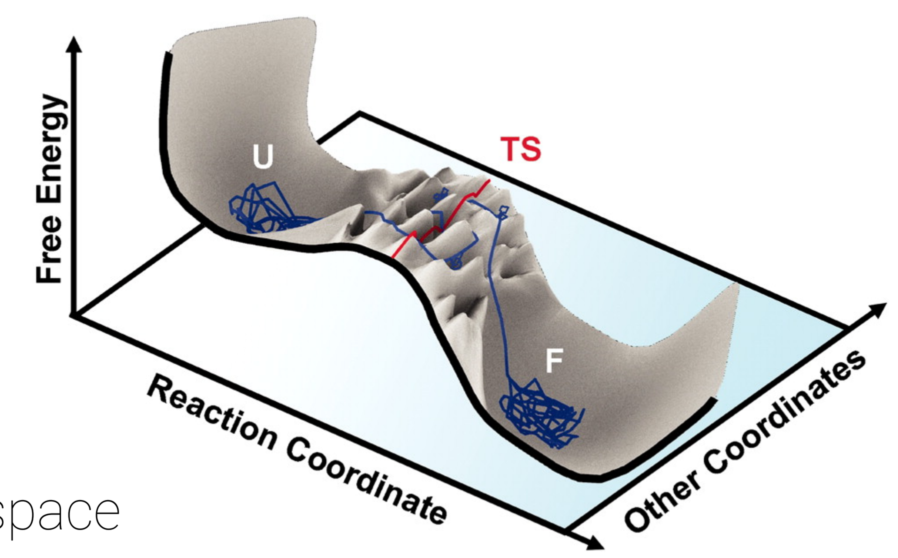
Efficient Sampling of Equilibrium States using Boltzmann Generators

Jeremy Binagia, Sean Friedowitz, Kevin Jia-Yu Hou
CS230 Final Project (Fall 2019)

Background and Motivation

Molecular Simulations are Useful

- Protein folding
- Drug discovery
- Materials design



Sampling is difficult

- Time consuming
- Huge state space
- Difficult to sample full state space
- Rare events often important
- Approximate sampling with MD or MCMC

Objective

- Apply Deep Learning to draw more representative samples, using approximate samplings + energy function as train data

Boltzmann Generators

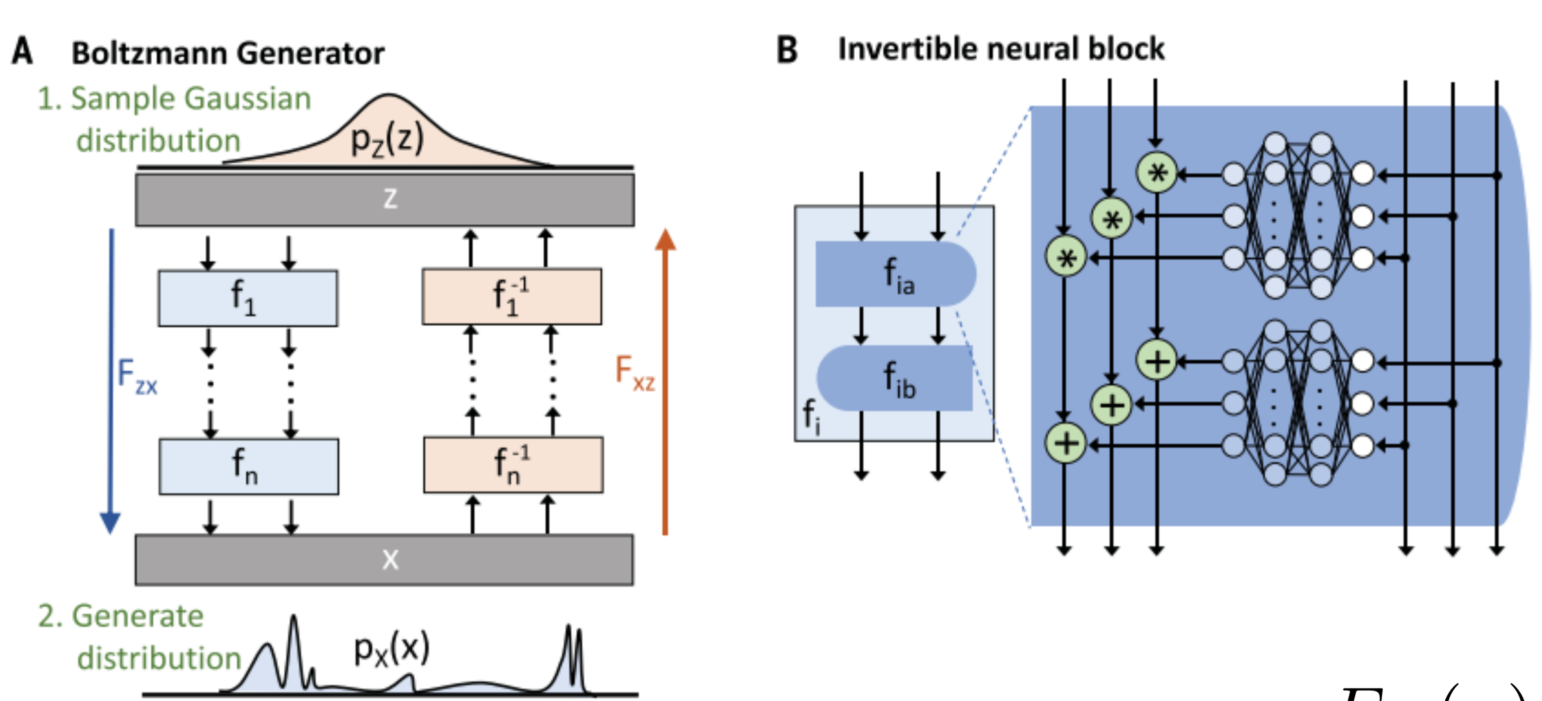
Boltzmann Distribution

- Energy function H(x) from physics
- Only know un-normalized probability

$$p(\mathbf{x}) = \frac{\exp(-H(\mathbf{x}))}{\mathcal{Z}}$$

$$p(\mathbf{z}) \sim \mathcal{N}(\mathbf{z})$$

Boltzmann Generators [2, 3]



$$\mathbf{z} = F_{xz}(\mathbf{x})$$

$$\mathbf{x} = F_{zx}(\mathbf{z})$$

$$J_{KL} = \mathbb{E}_{\mathbf{z}}\left[H(F_{zx}(\mathbf{z})) - \log R_{zx}(\mathbf{z})\right]$$

$$J_{ML} = \mathbb{E}_{\mathbf{x}}\left[\frac{1}{2}\|F_{xz}(\mathbf{x})\|^2 - \log R_{xz}(\mathbf{x})\right]$$

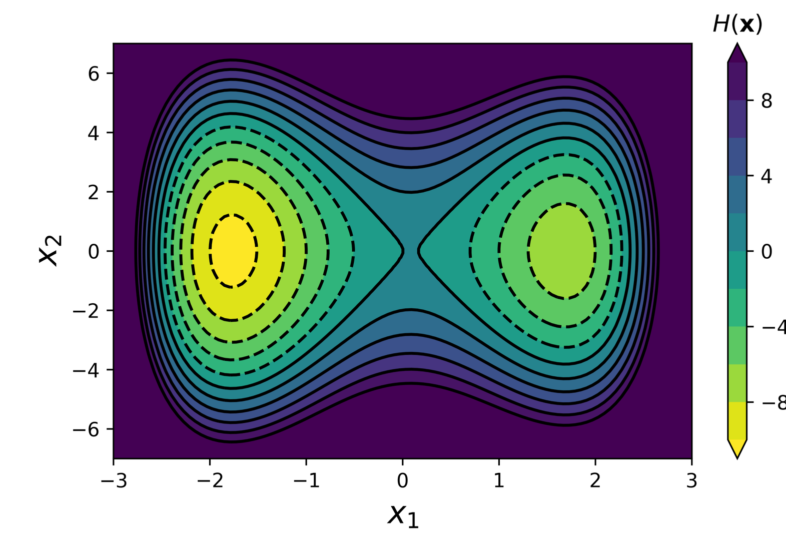
$$R_{zx} = \det\left(\frac{d\mathbf{z}}{d\mathbf{x}}\right)$$

Model 1: Double Well Potential

Model Description:

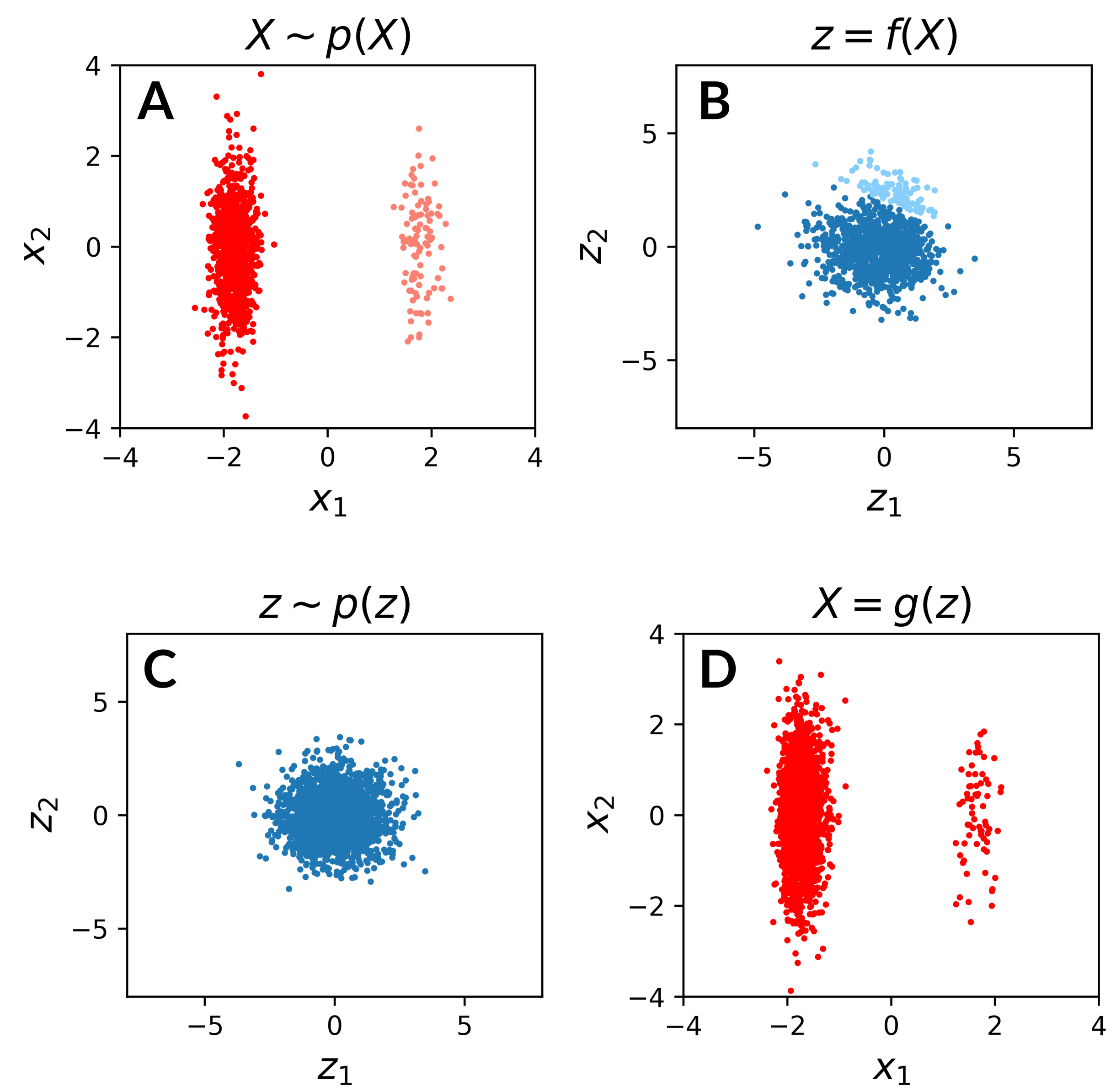
- Two stable states
- Reaction transition along x1

$$H(x_1, x_2) = x_1^4 - x_1^2 + x_1 + \frac{x_2^2}{2}$$



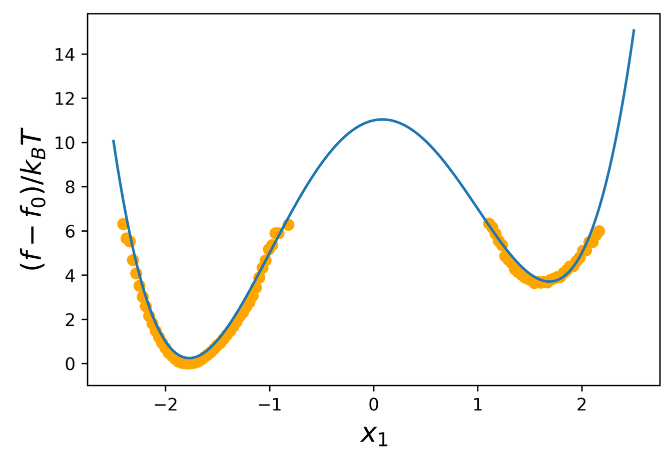
Training Results:

- After training by example (J_{ML}) and training by KL-loss (J_{KL}):



Commentary

- The network learns a transformation such that sampling in latent space (B) recovers the Boltzmann distribution in real space (D)
- Can now easily calculate equilibrium properties of interest (e.g. free energy)



IO/Parameters

- Adam optimizer
- Learning rate: 1e-4
- Batch size: 1000
- 4 real NVP blocks
- Translation (+) and scaling (*) networks have 3 hidden layers
- 256 units in each layer
- tanh and ReLU activations respectively in (*) and (+) nets

Model 2: Harmonic Oscillator

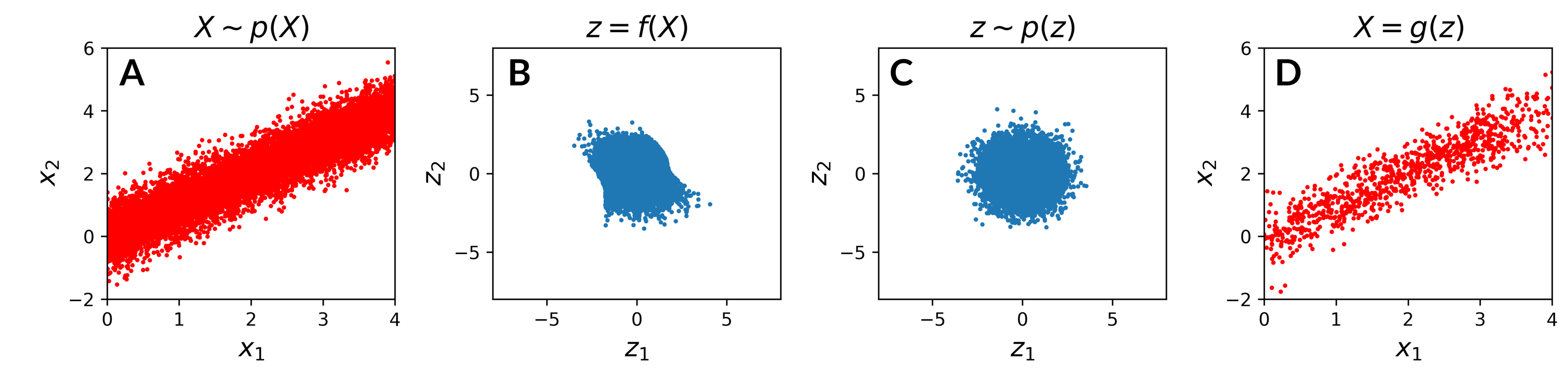
Model Description:

- Analytically tractable
- Illustrates how to handle simulation constraints

$$H(\vec{x}) = \begin{cases} k(x_2 - x_1)^2 & : \text{ if } 0 \le x_1 \le L \\ \infty & : \text{ otherwise} \end{cases}$$

$$f(x_1, x_2) = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} \Phi^{-1}\left(\frac{x_1}{L}\right) \\ \sqrt{2k}\,|x_2 - x_1| \end{bmatrix}$$

Training Results



Commentary

- Generated distribution (D) and transformed actual distribution (B) shows under sampling near constraints (x_1 = 0 and x_1 = L)

IO/Parameters

- Same as for double well except:
- Batch size: 128
- 100 units in each hidden layer
- Training by example only

Next Steps

Conclusions

- Successfully implemented the method of Boltzmann generators from scratch, validating on the double well potential
- Applied the method to a harmonic oscillator, where we derived the exact transformation the network Is learning

Next Steps

- Optimize the network architecture for exponential distributions

$$\det\left(\frac{df(\mathbf{x})}{d\mathbf{x}}\right) = \frac{\exp(-H(\mathbf{x}))}{\exp(-|\mathbf{z}|^2/2)} = \exp\left(\frac{|f(\mathbf{x})|^2}{2} - H(\mathbf{x})\right)$$

References

- [1] http://dlab.clemson.edu/?p=186 (intro figure)
- [2] Frank Noé, Simon Olsson, Jonas Köhler, and Hao Wu. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457), 2019.
- [3] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. *CoRR*, abs/1605.08803, 2016