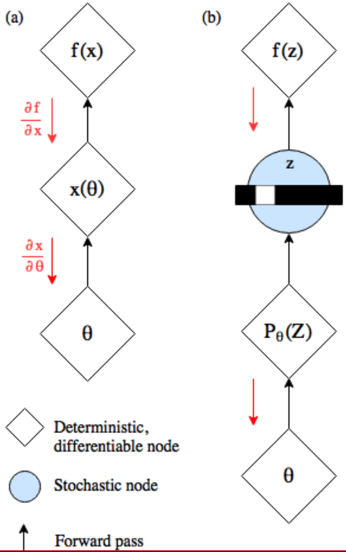
Gumbel Softmax

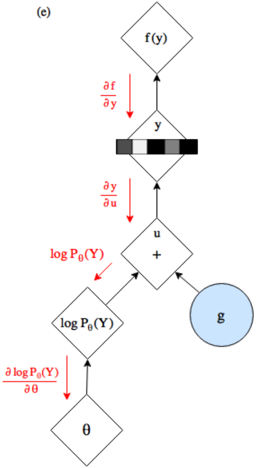
Machine Learning: Jordan Boyd-Graber
University of Maryland
SLIDES ADAPTED FROM ERIC JANG

Sampling screws up Backprop



- Problem for any single sample
- Can't backprop through sample

Sampling screws up Backprop



- Problem for any single sample
- Can't backprop through sample
- Express sample so gradient avoids randomness
- For example, $z \sim \mathcal{N}(\mu, \sigma)$ as $z = \mu + \sigma\epsilon$, $\epsilon \sim \mathcal{N}(0, 1)$

Gumbel

- Want to do the same thing for discrete distributions
- Instead of ϵ, we'll use Gumbel distribution
  - Sample $u \sim \text{Uniform}(0, 1)$
  - Compute $g = -\log(-\log(u))$
- We then could then draw samples from π_i with $\arg\max_i \left[g_i + \log \pi_i\right]$

Gumbel

- Want to do the same thing for discrete distributions
- Instead of ϵ, we'll use Gumbel distribution
  - Sample $u \sim \text{Uniform}(0,1)$
  - Compute $g = -\log(-\log(u))$
- We then could then draw samples from π_i with $\arg\max_i \left[g_i + \log \pi_i\right]$
- But $\arg\max$ isn't differentiable

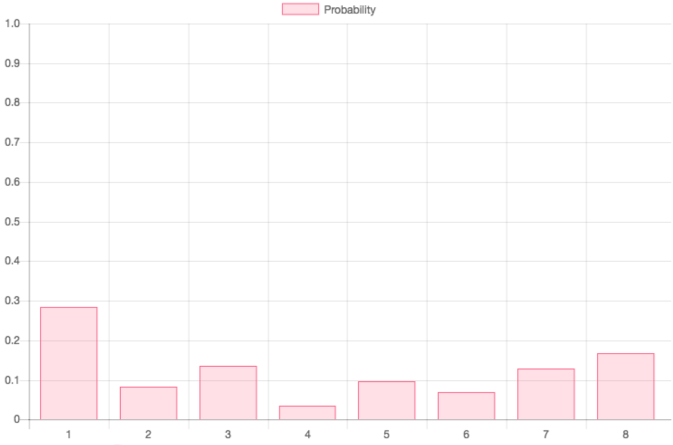
Backpropagate through Softmax

- "softmax" is a continuous approximation

$$y_i = \frac{\exp\left\{\frac{\log(\pi)+g_i}{\tau}\right\}}{\sum_j \exp\left\{\frac{\log(\pi_j)+g_j}{\tau}\right\}} \tag{1}$$

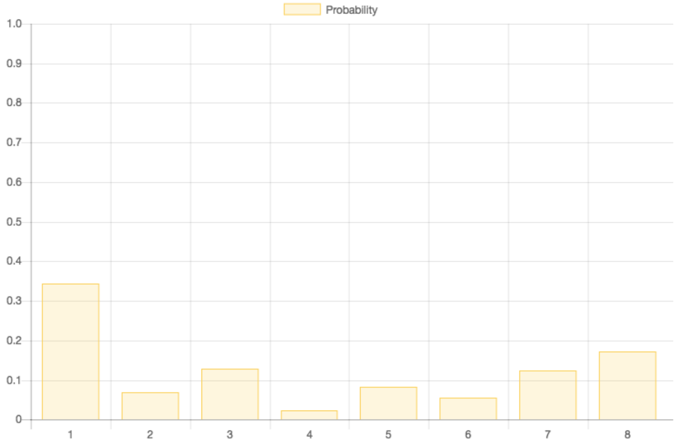
- τ is temperature that controls how close to max it is

Visualization



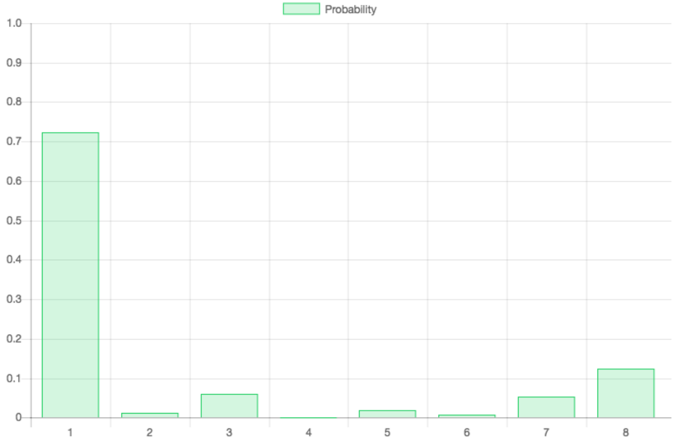
$\tau = 3$

Visualization



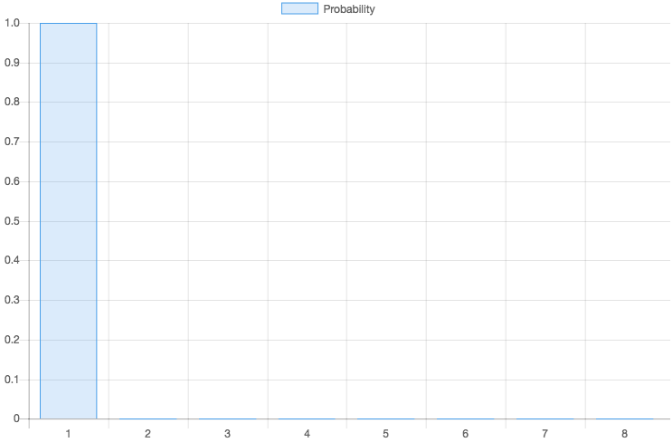
$\tau = 2$

Visualization



$\tau = 1$

Visualization



$\tau = 0.1$

Generative Modeling with Deep Networks

- Learning a distribution harder than learning a single prediction
- Very hard to evaluate too!
- Becomes even harder with discrete data