Sinkhorn Modeling

Instructor: John Thickstun

Discussion Board: Available on Ed

Zoom Link: Available on Canvas

Instructor Contact: thickstn@cs.washington.edu

Course Webpage: https://courses.cs.washington.edu/courses/cse599i/20au/

Entropy-Regularized OT

- Recall that our goal is to find $\arg\min_{\pi \in \Pi(p,q)} \langle c, \pi \rangle - \lambda H(\pi) = \arg\min_{\pi \in \Pi(p,q)} D(\pi \parallel p_k^\lambda)$.
- We decomposed this into an alternating minimization problem:

$$\pi_\lambda^{(\ell+1)} \equiv \begin{cases} \arg\min_{\pi \mathbf{1}_m} D(\pi \parallel \pi_\lambda^{(\ell)}) & \ell \text{ even}, \\ \arg\min_{\pi^\top \mathbf{1}_n} D(\pi \parallel \pi_\lambda^{(\ell)}) & \ell \text{ odd}. \end{cases}$$

- The sub-problems have closed form:

$$\pi_\lambda^{(2\ell)} = \operatorname{diag}\left(\frac{p}{\pi_\lambda^{(2\ell-1)} \mathbf{1}_m}\right) \pi_\lambda^{(2\ell-1)}, \text{ and } \pi_\lambda^{(2\ell+1)} = \operatorname{diag}\left(\frac{q}{\mathbf{1}_n^\top \pi_\lambda^{(2\ell)}}\right) \pi_\lambda^{(2\ell)}.$$

Entropy-Regularized OT

- Alternating minimization updates:

$$\pi_\lambda^{(\ell+1)} \equiv \begin{cases} \underset{\pi \mathbf{1}_m}{\arg\min}\, D(\pi \parallel \pi_\lambda^{(\ell)}) & \ell \text{ even}, \\ \underset{\pi^\top \mathbf{1}_n}{\arg\min}\, D(\pi \parallel \pi_\lambda^{(\ell)}) & \ell \text{ odd}. \end{cases}$$

- Each iterate $\pi_\lambda^{(\ell)} \in \mathbb{R}^{n \times m}$.

- But there are only $n + m$ constraints.

- We can optimize more efficiently in the dual space.

The Structure of the Minimizer

- Our goal is to find $\pi_\lambda = \underset{\pi \in \Pi(p,q)}{\arg\min} \langle c, \pi \rangle - \lambda H(\pi) = \underset{\pi \in \Pi(p,q)}{\arg\min} D(\pi \parallel p_k^\lambda)$.
- Equivalence is satisfied by

$$p_k^\lambda(x, y) = \frac{e^{-c(x,y)/\lambda}}{\sum_{x',y'} e^{-c(x',y')/\lambda}}.$$

- The minimizer π_λ has a special structure. For some $u \in \mathbb{R}^n, v \in \mathbb{R}^n$,

$$\pi_\lambda = \mathrm{diag}(u) K \, \mathrm{diag}(v).$$

- Where $K \in \mathbb{R}^{n \times m}$ and in particular, $K(x, y) \equiv e^{-c(x,y)/\lambda}$.

The Dual Perspective

- Our goal is to find $\pi_\lambda = \underset{\pi \in \Pi(p,q)}{\arg\min} \langle c, \pi \rangle - \lambda H(\pi) = \underset{\pi \in \Pi(p,q)}{\arg\min} D(\pi \parallel p_k^\lambda)$.
- Claim. For some $u \in \mathbb{R}^n, v \in \mathbb{R}^n$,

$$\pi_\lambda = \operatorname{diag}(u) K \operatorname{diag}(v).$$

- Proof. Consider the Lagrangian:

$$\mathcal{L}(\pi, f, g) = \langle c, \pi \rangle - \lambda H(\pi) - \langle f, \pi \mathbf{1}_m - p \rangle - \langle g, \pi^\top \mathbf{1}_n - q \rangle.$$

- First order optimality: $c(x,y) + \lambda \log \pi_\lambda(x,y) - f_x - g_y = 0$.
- Solution: $\pi_\lambda(x,y) = e^{f_x/\lambda - 1/2} e^{-c(x,y)/\lambda} e^{g_y/\lambda - 1/2}$.

Matrix Scaling

- Our goal is to find $\pi_\lambda = \underset{\pi \in \Pi(p,q)}{\arg\min} \langle c, \pi \rangle - \lambda H(\pi) = \underset{\pi \in \Pi(p,q)}{\arg\min} D(\pi \parallel p_k^\lambda)$.
- Claim. For some $u \in \mathbb{R}^n, v \in \mathbb{R}^n$, $\pi_\lambda = \operatorname{diag}(u) K \operatorname{diag}(v)$.
- What are the values u, v? Must satisfy the constraints:

$$p = \pi_\lambda \mathbf{1}_m = \operatorname{diag}(u)(Kv) \text{ and } q = \pi_\lambda^\top \mathbf{1}_n = \operatorname{diag}(v)(K^\top u).$$

- This is an instance of the general “matrix-scaling problem.”
- Find a scaling of matrix K so that its columns sum to p and rows sum to q.

Sinkhorn-Knopp Algorithm

- Matrix scaling problem: find u, v such that

$$p = \pi_\lambda \mathbf{1}_m = \mathrm{diag}(u)(Kv) \text{ and } q = \pi_\lambda^\top \mathbf{1}_n = \mathrm{diag}(v)(K^\top u).$$

- The Sinkhorn-Knopp algorithm:

  1. Initialize $u^{(1)} = \mathbf{1}_n, \text{ and } v^{(1)} = \mathbf{1}_m.$
  2. Alternating updates: $u^{(\ell+1)} \equiv \frac{p}{Kv^{(\ell)}}, \text{ and } v^{(\ell+1)} \equiv \frac{q}{K^\top u^{(\ell+1)}}.$

- Division is interpreted entry-wise.

- Why is this the right thing to do???

From Dual to Primal Iterates

- The Sinkhorn-Knopp: alternating updates

$$u^{(\ell+1)} \equiv \frac{p}{Kv^{(\ell)}}, \text{ and } v^{(\ell+1)} \equiv \frac{q}{K^\top u^{(\ell+1)}}.$$

- Convert the dual iterates back to primal iterates:

$$\tilde{\pi}_\lambda^{(2\ell)} \equiv \mathrm{diag}(u^{(\ell+1)}) K \, \mathrm{diag}(v^{(\ell)}),$$
$$\tilde{\pi}_\lambda^{(2\ell+1)} \equiv \mathrm{diag}(u^{(\ell+1)}) K \, \mathrm{diag}(v^{(\ell+1)}).$$

- Compare to iterates of the primal algorithm:

$$\pi_\lambda^{(2\ell)} = \mathrm{diag}\left(\frac{p}{\pi_\lambda^{(2\ell-1)}\mathbf{1}_m}\right)\pi_\lambda^{(2\ell-1)}, \text{ and } \pi_\lambda^{(2\ell+1)} = \mathrm{diag}\left(\frac{q}{\mathbf{1}_n^\top \pi_\lambda^{(2\ell)}}\right)\pi_\lambda^{(2\ell)}.$$

Primal/Dual Equivalence

- Focus on the even case. The primal update is

$$\pi_\lambda^{(2\ell)} = \operatorname{diag}\left(\frac{p}{\pi_\lambda^{(2\ell-1)}\mathbf{1}_m}\right)\pi_\lambda^{(2\ell-1)}.$$

- And the corresponding dual update is:

$$\tilde{\pi}_\lambda^{(2\ell)} = \operatorname{diag}(u^{(\ell+1)})K\operatorname{diag}(v^{(\ell)}) = \operatorname{diag}\left(\frac{p}{Kv^{(\ell)}}\right)\frac{\tilde{\pi}_\lambda^{(2\ell-1)}}{\operatorname{diag}(u^{(\ell)})}$$

$$= \operatorname{diag}\left(\frac{p}{\operatorname{diag}(u^{(\ell)})Kv^{(\ell)}}\right)\tilde{\pi}_\lambda^{(2\ell-1)} = \operatorname{diag}\left(\frac{p}{\tilde{\pi}_\lambda^{(2\ell-1)}\mathbf{1}_m}\right)\tilde{\pi}_\lambda^{(2\ell-1)}.$$

5-Minute Break

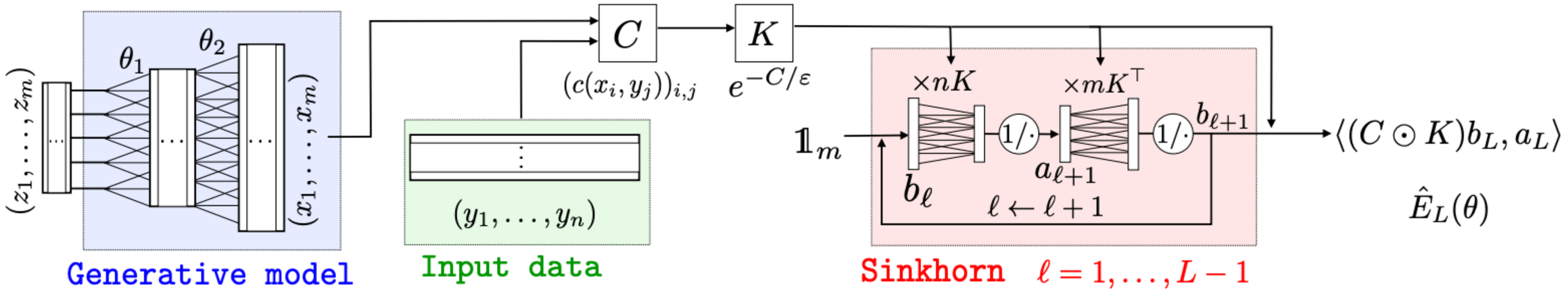
Generative Sinkhorn Modeling

- Draw mini batches $x_1, \ldots, x_B \sim p$ and $y_1, \ldots, y_B \sim p_\theta$.
- Approximate $W_1(p, p_\theta) \approx W_1(\hat{p}, \hat{p}_\theta)$.
- Where (empirical distribution)

$$\hat{p}(x) = \frac{1}{B} \sum_{i=1}^{B} \mathbf{1}_{x_i = x}, \text{ and } \hat{p}_\theta(y) = \frac{1}{B} \sum_{i=1}^{B} \mathbf{1}_{y_i = y}.$$

- Estimate $W_1(\hat{p}, \hat{p}_\theta)$ using Sinkhorn's algorithm.
- Compute gradient updates $\theta' = \theta - \eta \nabla_\theta W_1(\hat{p}, \hat{p}_\theta)$.

Sinkhorn Modeling Illustrated



Genevay, Peyré, and Cuturi, 2018

- Looks like a GAN architecture. But there are no parameters in the red block!
- Notational translation: a, b, ε in the Figure are u, v, λ in our discussion.

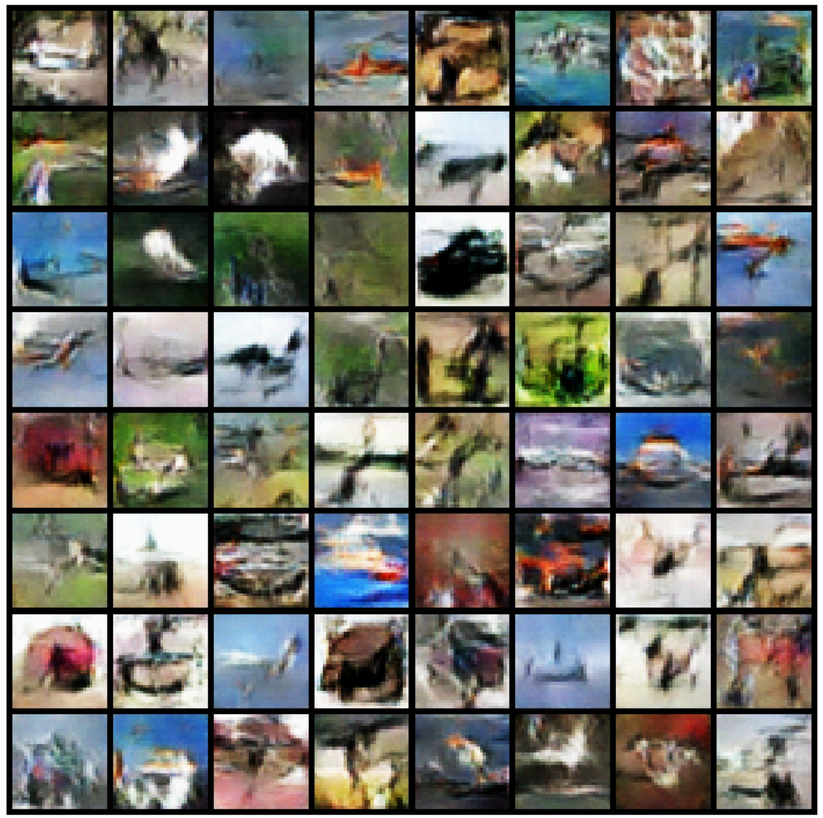
Comparing to Wasserstein GAN

- Wasserstein GAN: solve a saddle point using dual approximation

$$W(p, q) = \sup_{\|h\|_L \leq 1} \left[\mathop{\mathbb{E}}_{x \sim p} h(x) - \mathop{\mathbb{E}}_{x \sim q} h(x) \right].$$

- Use samples to approximate the expectation (monte carlo).
- Sinkhorn modeling: estimate $W(p, q) \approx W(\hat{p}, \hat{q})$ with empirical distributions.
- Solve discrete problem $W(\hat{p}, \hat{q})$ directly (Sinkhorn-Knopp algorithm).
- Very different statistical estimators!

Why Doesn't this Work Better?



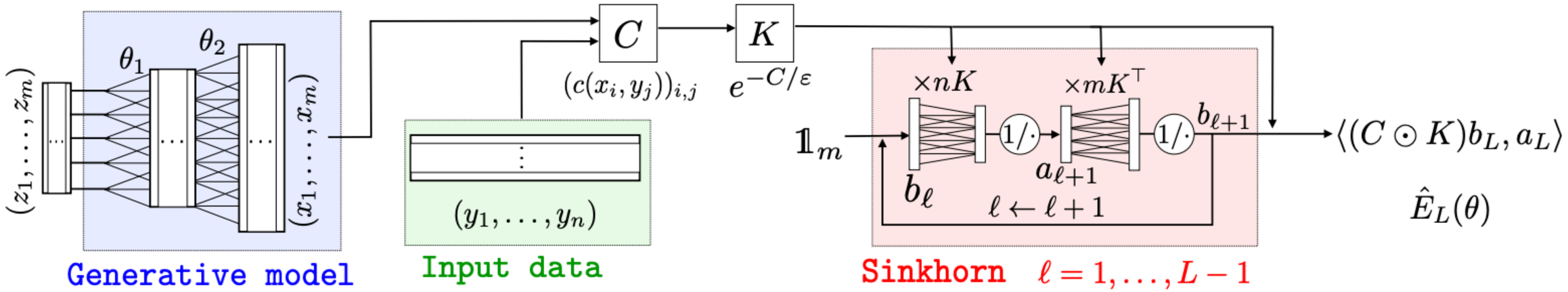
Sinkhorn Modeling (IS = 4.8)

Genevay, Peyré, and Cuturi, 2018

Wasserstein GAN (IS = 8.2)

Results of Homework 3

Sinkhorn Modeling Illustrated



Genevay, Peyré, and Cuturi, 2018

- How many steps do we need to take for Sinkhorn (what is the rate?)
- Is back-prop through the Sinkhorn solver a good gradient estimator?

Learning the Cost Function?

- Wasserstein distance lifts a metric on $\mathcal{X}$ to metric on probability distributions:

$$W(p, q) = \inf_{\pi \in \Pi(p,q)} \mathop{\mathbb{E}}_{(x,y)\sim\pi} \left[c(x, y)\right].$$

- Wasserstein GAN just uses $c(x, y) = \|x - y\|_2$.

- Genevay et. al. learn a cost function $c_\phi(x, y) = \|f_\phi(x) - f_\phi(y)\|_2$.

- Could we learn a cost function for the Wasserstein GAN?

- Do we really need a learned cost for Sinkhorn modeling?