ECE/CS 559 Lecture 19/20 Th 10/31; Th 11/7

Last time: Hebbian updates

① Generative Models

- All unsupervised learning is about understanding $P(x)$
- Generative models attempt to create more x:
  - They can do this by modeling $P(x)$ & sampling from it.
  - Or, they can directly model the sampling mechanism
    e.g., sample z (some standard distribution), then let $x = f(z) + \text{noise}$
- Conditional generative models extend to generation via prompts: $P(x|y)$ ↑

Examples: y = caption, x = image (diffusion models)
y = question, x = answer (transformers)

② Density Estimation

Given Data x, assume i.i.d. from P.

Parametric methods: Let $P(x) = f(x; w)$

Maximize likelihood: $\arg\max_w P(\text{Data}) = \prod_{x \in \text{Data}} P(x)$

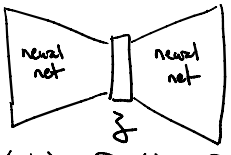
$\Leftrightarrow \arg\min_w \sum_{x \in \text{Data}} -\log P(x)$ ← log loss

Non-parametric methods: Allow # of parameters (size of w) to grow with data size. Example, kernel density estimation

Works well when x discrete or small dimensional.
Otherwise, requires regularization, e.g. smooth densities.

③ Autoencoders (Variational; VAEs)

- They try to capture $P(x)$ by reducing the "essence" of x to a simple representation z (encoding)
  then generating x from a sample of z (decoding)



$f(z)$, approximates x (+ noise)

Generally: $P_{enc}(z|x)$ $P_{prior}(z)$ $P_{dec}(x|z)$

- Why sampling $z \sim P_{prior}(z)$ then letting $x \sim f(z)$ work? How do we choose P_{enc}, P_{dec}?

④ Derived distributions:

- Say $x \in \mathbb{R}^1$. Let $z \sim P_z(z)$ and let $x = f(z)$.
- What is $P_x(x)$? Let's assume P's are densities, f monotonic)
- $F_x(x) = P(X \le x) = P(f(z) \le x) = P(z \le f^{-1}(x))$ decreasing

thus $P_x(x) = \frac{dF}{dx} = \frac{d}{dx} F_z(f^{-1}(x)) = \frac{d}{dz} F_z \cdot \frac{dz}{dx} = P_z(f^{-1}(x)) \cdot \left|\frac{df^{-1}}{dx}\right|$

Example: $P_z \sim \text{Uniform}[0,1]$ $x = -\log z$ $f^{-1}(x) = e^{-x}$

$\Rightarrow P_x(x) = \begin{cases} -1 \cdot (-e^{-x}) & e^{-x} \in (0,1) \\ 0 & \text{otherwise} \end{cases} = \begin{cases} e^{-x} & x \ge 0 \\ 0 & x < 0 \end{cases}$

- Can create any new random variable from standard random variables (e.g. uniform, Gaussian, etc.)

⑤ From maximum likelihood to ELBO

How do we train an autoencoder?

Idea 1: Fix $P_{prior}(z)$ and only train the decoder $P_{dec}(x|z)$

by minimizing $\sum_{x \in \text{Data}, z \sim P_{prior}} -\log P_{dec}(x|z)$

Issue: Creates disconnect between x & z (close x's $\not\Rightarrow$ close z's)

Idea 2: Fix $P_{enc}(z|x)$ and train $P_{prior}(z)$ & $P_{dec}(x|z)$

by minimizing $\sum_{x \in \text{Data}, z \sim P_{enc}(\cdot|x)} -\log P_{prior}(z) \cdot P_{dec}(x|z)$

approximates $\mathbb{E}_x \mathbb{E}_{z \sim P_{enc}|x} -\log P_{prior}(z) P_{dec}(x|z)$ $\Leftrightarrow$

implicit $P_{gen}(x) \quad P_{gen}(z|x)$

$\mathbb{E}_x\left[\log \frac{1}{P_{gen}(x)}\right] + \mathbb{E}_x\left[\mathbb{E}_{z \sim P_{enc}|x}\left[\log \frac{1}{P_{gen}(z|x)}\right]\right] - \mathbb{E}_x\left[\mathbb{E}_{z \sim P_{enc}|x}\left[\log \frac{1}{P_{enc}(z|x)}\right]\right]$

min at $P_{gen}(x) = P_x(x)$ min at $P_{gen}(z|x) = P_{enc}(z|x)$ $-$ Entropy (Penc)

Issue: If the neural net of the decoder cannot easily "invert" the choice of P_{enc}, it won't work well.

Idea 3 Train P_{enc} and P_{dec} jointly.

Issue: P_{enc} in the loss is just a sampling distribution.

Hard to optimize: bad local minima / P_{enc} can degenerate by colluding w/ P_{dec}

Idea 4: Include P_{enc} in the loss: $\mathbb{E}\left[-\log \frac{P_{prior}(z) P_{dec}(x|z)}{P_{enc}(z|x)}\right]$

"max-entropy" effect

Evidence Lower Bound (ELBO) Loss $< \log P(x)$

③ Example: Gaussian Variational Auto encoders

$p_{prior}(z) \propto \exp\left(-\frac{\|z\|^2}{2}\right)$ $\quad \mathcal{N}(0, I)$

$p_{encoder}(z|x) \propto \exp\left(-\frac{\|z - g(x;w)\|^2}{2}\right)$ $\quad \mathcal{N}(g(x;w), I)$

$p_{decoder}(x|z) \propto \exp\left(-\frac{\|x - f(z;w)\|^2}{2}\right)$ $\quad \mathcal{N}(f(z;w), I)$

ELBO has explicit form: $\mathbb{E}\left[\frac{\|z\|^2}{2} - \frac{\|z - g(x;w)\|^2}{2}\right] = \frac{1}{2}\mathbb{E}\left[\|g(x;w)\|^2\right]$

(minimize representation norm)

$+ \frac{1}{2}\mathbb{E}\left[\|x - f(\underbrace{g(x;w) + N}_{\text{encode}}; w)\|^2\right]$ $\quad N \sim \mathcal{N}(0, I)$

(minimize reconstruction error) $\uparrow$ $\quad$ decode

This looks almost like kmeans and it's not a coincidence! The similarity is closer if we quantize z.

④ Hierchical VAEs

- Instead of one representation stage, have multiple:

$$\underbrace{x \to z_1 \to z_2 \to z_3}_{\text{encode}} \to \underbrace{z_2 \to z_1 \to x}_{\text{decode}}$$

- Why? Allows for gradual tweaks, making it easier to train.

Example: Diffusion models use encoders that just add noise.

ELBO: $\log \frac{p_{prior}(z_3)\, p_{dec}(z_2|z_3)\, p_{dec}(z_1|z_2)\, p_{dec}(x|z_1)}{p_{enc}(z_3|z_2)\, p_{enc}(z_2|z_1)\, p_{enc}(z_1|x)}$