Flow contrastive estimation of energy based models

a tea talk, Jan 27

Gao *et al.*, NuerIPS 2019

Background - EBM

- focus on estimating **energy based models** (EBMs):
  - express as density $p(\mathbf{x})$ for $\mathbf{x} \in \mathbb{R}^d$ as:

$$p(\mathbf{x}) = \frac{\exp\left(-E_\theta(\mathbf{x})\right)}{Z(\theta)}$$

    where $E_\theta : \mathbb{R}^d \to \mathbb{R}$ is the *energy function*.
  - so we can parameterize an energy based model with any function that maps $\mathbb{R}^d$ to a scalar.
  - but, computing $Z(\theta) = \int \exp\left(-E_\theta(\mathbf{x})\right) d\mathbf{x}$ is difficult
  - $\Rightarrow$ several approaches: contrastive divergence, score matching, **noise contrastive estimation**

Background - NCE

- setup:
  - observe $\mathbf{x}_1, \ldots, \mathbf{x}_n \sim p_d(\cdot)$.
  - wish to approximation $p_d(\cdot)$ with $p_\theta(\cdot)$ which is an *unnormalized* EBM (i.e., $Z(\theta)$ difficult to compute).

Background - NCE

- setup:
  - observe $\mathbf{x}_1, \ldots, \mathbf{x}_n \sim p_d(\cdot)$.
  - wish to approximation $p_d(\cdot)$ with $p_\theta(\cdot)$ which is an *unnormalized* EBM (i.e., $Z(\theta)$ difficult to compute).
- **noise contrastive estimation** (NCE; Gutmann & Hyvärinen, 2012):
  - propose a *noise* distribution $p_n(\cdot)$ and sample $\mathbf{y}_1, \ldots, \mathbf{y}_n \sim p_n(\cdot)$
  - learn to classify the mixture $U \sim \frac{1}{2} p_d(\cdot) + \frac{1}{2} p_n(\cdot)$ based on the log-odds ratio:

$$r(\cdot) = \texttt{sigmoid}(\ \log p_\theta(\cdot) + c - \log p_n(\cdot)\ )$$

Background - NCE

- setup:
  - observe $\mathbf{x}_1, \ldots, \mathbf{x}_n \sim p_d(\cdot)$.
  - wish to approximation $p_d(\cdot)$ with $p_\theta(\cdot)$ which is an *unnormalized* EBM (i.e., $Z(\theta)$ difficult to compute).
- **noise contrastive estimation** (NCE; Gutmann & Hyvärinen, 2012):
  - propose a *noise* distribution $p_n(\cdot)$ and sample $\mathbf{y}_1, \ldots, \mathbf{y}_n \sim p_n(\cdot)$
  - learn to classify the mixture $U \sim \frac{1}{2} p_d(\cdot) + \frac{1}{2} p_n(\cdot)$ based on the log-odds ratio:

$$r(\cdot) = \texttt{sigmoid}(\ \log p_\theta(\cdot) + c - \log p_n(\cdot)\ )$$

  - noise distribution must satisfy:
    1. easy to sample from (in order to get $\mathbf{y}_i$)
    2. easy to evaluate (log) density
    3. (somewhat) similar to data distribution, $p_d(\cdot)$

Flow contrastive estimation - FCE

- **idea**: use a deep net to parameterize noise, $p_n(\cdot)$
  - use a **flow** model as they satisfy all requirements (can evaluate normalized density and easy to sample from)
  - flow models are parameterized by a series of *invertible* transformations, designed to ensure Jacobian is tractable

$$\mathbf{y} = g_\alpha(\mathbf{z}); \quad z \sim q_0(\cdot)$$

$$\log p_\alpha(\mathbf{y}) = \log q_0(g_\alpha^{-1}(\mathbf{y})) + \log \det \mathbf{J} g_\alpha^{-1}$$

Flow contrastive estimation - FCE

- **idea**: use a deep net to parameterize noise, $p_n(\cdot)$
  - use a **flow** model as they satisfy all requirements (can evaluate normalized density and easy to sample from)
  - flow models are parameterized by a series of *invertible* transformations, designed to ensure Jacobian is tractable

$$\mathbf{y} = g_\alpha(\mathbf{z}); \quad z \sim q_0(\cdot)$$

$$\log p_\alpha(\mathbf{y}) = \log q_0(g_\alpha^{-1}(\mathbf{y})) + \log \det \mathbf{J} g_\alpha^{-1}$$

- **flow contrastive estimation**:
  - sample $\mathbf{y}_1, \ldots, \mathbf{y}_n \sim p_\alpha(\cdot)$
  - for θ, learn to classify the mixture $U \sim \frac{1}{2} p_d(\cdot) + \frac{1}{2} p_\alpha(\cdot)$ based on the log-odds ratio:

$$r(\cdot) = \texttt{sigmoid}(\ \log p_\theta(\cdot) + c - \log p_\alpha(\cdot)\ )$$

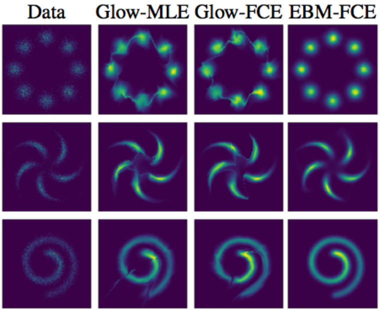
  - for α, learn to fool the EBM. Corresponds to learning a flow model via minimizing JSD instead of MLE.

Why is this a reasonable idea?

- Flow models:
  - popular because they allow for efficient evaluation of density and sampling $\Rightarrow$ can train via MLE
  - but must assume true density can be approximated via a series of invertible transformations

- energy based models:
  - parameterize the data density using only the energy (no assumptions implicit in the flow model) $\Rightarrow$ more flexible
  - also easy to compute log-density (up to norm. constant)
  - but sampling from EBMs is very expensive

Experimental results



Experimental results