Latent Graph Models

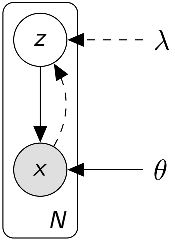
Wilker Aziz
University of Amsterdam

May 14, 2018

Outline

Variational auto-encoder

Generative model with NN likelihood



- complex (non-linear) observation model $p_\theta(x|z)$
- complex (non-linear) mapping from data to latent variables $q_\lambda(z|x)$

Jointly optimise generative model $p_\theta(x|z)$ and inference model $q_\lambda(z|x)$ under the same objective (ELBO)

Kingma and Welling (2013)

Objective

$$\log p_\theta(x) \geq \overbrace{\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|Z)\right] - \mathrm{KL}\left(q_\lambda(z|x) \,||\, p(z)\right)}^{\text{ELBO}}$$

Objective

$$\log p_\theta(x) \geq \overbrace{\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|Z)\right] - \mathrm{KL}\left(q_\lambda(z|x) \;||\; p(z)\right)}^{\text{ELBO}}$$

Parameter estimation

$$\underset{\theta,\lambda}{\arg\max}\ \mathbb{E}_{q(\epsilon)}\left[\log p_\theta(x|\underbrace{h^{-1}(\epsilon,\lambda)}_{=z})\right] - \mathrm{KL}\left(q_\lambda(z|x) \;||\; p(z)\right)$$

Objective

$$\log p_\theta(x) \geq \overbrace{\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|Z)\right] - \mathrm{KL}\left(q_\lambda(z|x) \mid\mid p(z)\right)}^{\text{ELBO}}$$

Parameter estimation

$$\underset{\theta,\lambda}{\arg\max}\ \mathbb{E}_{q(\epsilon)}\left[\log p_\theta(x|\underbrace{h^{-1}(\epsilon,\lambda)}_{=z})\right] - \mathrm{KL}\left(q_\lambda(z|x) \mid\mid p(z)\right)$$

- assume $\mathrm{KL}\left(q_\lambda(z|x) \mid\mid p(z)\right)$ analytical
  true for exponential families

Objective

$$\log p_\theta(x) \geq \overbrace{\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|Z)\right] - \mathrm{KL}\left(q_\lambda(z|x) \mid\mid p(z)\right)}^{\text{ELBO}}$$

Parameter estimation

$$\arg\max_{\theta,\lambda}\ \mathbb{E}_{q(\epsilon)}\left[\log p_\theta(x|\underbrace{h^{-1}(\epsilon,\lambda)}_{=z})\right] - \mathrm{KL}\left(q_\lambda(z|x) \mid\mid p(z)\right)$$

- assume $\mathrm{KL}\left(q_\lambda(z|x) \mid\mid p(z)\right)$ analytical
  true for exponential families
- approximate $\mathbb{E}_{q(\epsilon)}\left[\log p_\theta(x|h^{-1}(\epsilon,\lambda))\right]$ by sampling
  requires a reparameterisation
  $h^{-1}(\epsilon,\lambda) \sim q_\lambda(z|x) \Leftrightarrow h(z,\lambda) \sim q(\epsilon)$

Outline

Discrete variables

Suppose z is a d-dimensional binary vector
i.e. $z_i \in \{0, 1\}$

Discrete variables

Suppose z is a d-dimensional binary vector
i.e. $z_i \in \{0, 1\}$

Then let's define an inference model

$$q_\lambda(z|x) = \underbrace{\prod_{i=1}^{d} q_\lambda(z_i|x)}_{\text{mean field}}$$

Discrete variables

Suppose z is a d-dimensional binary vector
i.e. $z_i \in \{0, 1\}$

Then let's define an inference model

$$q_\lambda(z|x) = \underbrace{\prod_{i=1}^{d} q_\lambda(z_i|x)}_{\text{mean field}} = \prod_{i=1}^{d} \text{Bern}(z_i|\underbrace{\text{sigmoid}(f_\lambda(x))}_{\text{NN}}) \tag{1}$$

Discrete variables

Suppose z is a d-dimensional binary vector
i.e. $z_i \in \{0, 1\}$

Then let's define an inference model

$$q_\lambda(z|x) = \underbrace{\prod_{i=1}^{d} q_\lambda(z_i|x)}_{\text{mean field}} = \prod_{i=1}^{d} \text{Bern}(z_i|\underbrace{\text{sigmoid}(f_\lambda(x))}_{\text{NN}}) \quad (1)$$

Can we reparameterise $q_\lambda(z|x)$?

Bernoulli pmf

$$\text{Bern}(z_i|b_i) = b_i^{z_i}(1 - b_i)^{1-z_i}$$

Bernoulli pmf

$$
\begin{aligned}
\mathrm{Bern}(z_i|b_i) &= b_i^{z_i}(1-b_i)^{1-z_i} \\
&= \begin{cases} b_i & \text{if } z_i = 1 \\ 1-b_i & \text{if } z_i = 0 \end{cases}
\end{aligned} \tag{2}
$$

Bernoulli pmf

$$\begin{aligned} \mathrm{Bern}(z_i|b_i) &= b_i^{z_i}(1-b_i)^{1-z_i} \\ &= \begin{cases} b_i & \text{if } z_i = 1 \\ 1-b_i & \text{if } z_i = 0 \end{cases} \end{aligned} \tag{2}$$

Can we reparameterise a Bernoulli variable?

Reparameterisation requires a Jacobian matrix

Not really :(

$$q(z;\lambda) = \underbrace{\phi(\epsilon = h(z,\lambda))\left|\det J_{h(z,\lambda)}\right|}_{\text{change of density}} \tag{3}$$

Elements in the Jacobian matrix

$$J_{h(z,\lambda)}[i,j] = \frac{\partial h_i(z,\lambda)}{\partial z_j}$$

are not defined for non-differentiable functions

Relaxation

Let's redefine z_i to live in the interval $(0, 1)$

- and find an alternative reparameterisable **density**

Examples

- $Z \sim \mathcal{LN}(u, s^2)$
  $z = \text{sigmoid}(u + s\epsilon)$ with $\epsilon \sim \mathcal{N}(0, 1)$
- $Z \sim \text{Concrete}(u, \tau)$
  $z = \text{sigmoid}(\frac{u+\epsilon}{\tau})$ with $\epsilon \sim \text{Gumbel}(0, 1)$

Relaxation

Let's redefine z_i to live in the interval $(0, 1)$

- and find an alternative reparameterisable **density**

Examples

- $Z \sim \mathcal{LN}(u, s^2)$
  $z = \text{sigmoid}(u + s\epsilon)$ with $\epsilon \sim \mathcal{N}(0, 1)$
- $Z \sim \text{Concrete}(u, \tau)$
  $z = \text{sigmoid}(\frac{u+\epsilon}{\tau})$ with $\epsilon \sim \text{Gumbel}(0, 1)$

But note that we **no longer** have a discrete variable

Concrete or Gumbel-Softmax

An alternative parameterisation of a Categorical variable

$$
\begin{aligned}
A &\sim \mathrm{Cat}(\mathrm{softmax}(\phi)) \\
A &= \arg\max_i \ [\phi_i + \epsilon_i]_{i=1}^K \quad \text{where } \epsilon \sim \mathrm{Gumbel}(0, I)
\end{aligned} \tag{4}
$$

Concrete or Gumbel-Softmax

An alternative parameterisation of a Categorical variable

$$\begin{aligned} A &\sim \mathrm{Cat}(\mathrm{softmax}(\phi)) \\ A &= \arg\max_i \; [\phi_i + \epsilon_i]_{i=1}^K \quad \text{where } \epsilon \sim \mathrm{Gumbel}(0, I) \end{aligned} \tag{4}$$

We can sample a one-hot encoding of the categorical variable

$$B = \mathrm{onehot}\left(\arg\max_i \; [\phi_i + \epsilon_i]_{i=1}^K\right) \tag{5}$$

Concrete or Gumbel-Softmax

An alternative parameterisation of a Categorical variable

$$A \sim \text{Cat}(\text{softmax}(\phi))$$
$$A = \arg\max_i \; [\phi_i + \epsilon_i]_{i=1}^K \quad \text{where } \epsilon \sim \text{Gumbel}(0, I) \tag{4}$$

We can sample a one-hot encoding of the categorical variable

$$B = \text{onehot}\left(\arg\max_i \; [\phi_i + \epsilon_i]_{i=1}^K\right) \tag{5}$$

And we get a continuous relaxation with softmax

$$B = \text{softmax}(\phi + \epsilon) \tag{6}$$

Concrete or Gumbel-Softmax

An alternative parameterisation of a Categorical variable

$$
\begin{aligned}
A &\sim \mathrm{Cat}(\mathrm{softmax}(\phi)) \\
A &= \arg\max_i \; [\phi_i + \epsilon_i]_{i=1}^K \quad \text{where } \epsilon \sim \mathrm{Gumbel}(0, I)
\end{aligned} \tag{4}
$$

We can sample a one-hot encoding of the categorical variable

$$
B = \mathrm{onehot}\left(\arg\max_i \; [\phi_i + \epsilon_i]_{i=1}^K\right) \tag{5}
$$

And we get a continuous relaxation with softmax

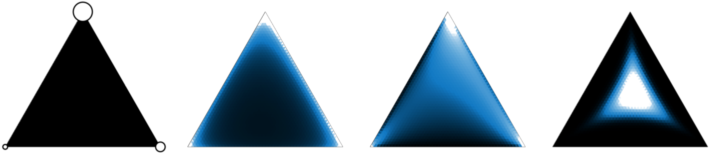
$$
B = \mathrm{softmax}(\phi + \epsilon) \tag{6}
$$

Finally, with a temperature τ we can approach a one-hot encoding of the most likely category as $\tau \to 0$

$$
B = \mathrm{softmax}\left(\frac{\phi + \epsilon}{\tau}\right) \tag{7}
$$

Simplex

The tips of the simplex represent a one-hot encoding of a 3-way Categorical variable

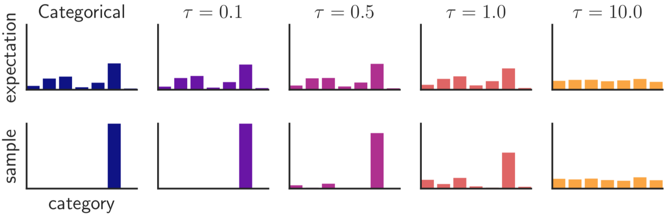


(a) $\lambda = 0$ (b) $\lambda = 1/2$ (c) $\lambda = 1$ (d) $\lambda = 2$

- the softmax relaxes the variable to take on values in the interior of the simplex
- as we cool down the system we push most of the mass towards the tips

Illustrations from (Maddison et al., 2016).

Concrete samples



Illustrations from (Jang et al., 2016).

Summary

Deep learning in NLP

- task-driven feature extraction
- models with more realistic assumptions

Probabilistic modelling

- better (or at least more explicit) statistical assumptions
- compact models
- semi-supervised learning

Literature I

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. 2013. URL `http://arxiv.org/abs/1312.6114`.

Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.