Variational Auto-Encoders

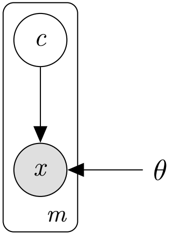
Wilker Aziz

Universiteit van Amsterdam
w.aziz@uva.nl

May 18, 2017

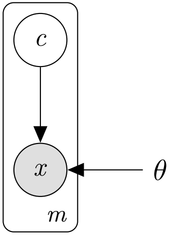
Generative models with neural networks

Mixture model



Generative models with neural networks

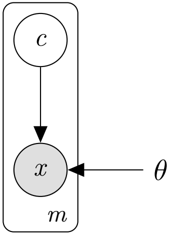
Mixture model



- sample a latent class $c \in \{1, \ldots, K\}$
  $c \sim U(\frac{1}{K})$

Generative models with neural networks

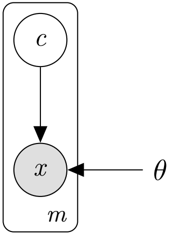
Mixture model



- sample a latent class $c \in \{1, \ldots, K\}$
  $c \sim U(\frac{1}{K})$
- generate categorical observation x from c
  $x \sim P(X|C = c)$

Generative models with neural networks

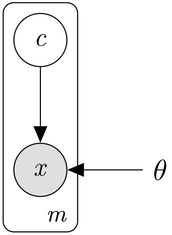
Mixture model



- sample a latent class $c \in \{1, \dots, K\}$
  $c \sim U(\frac{1}{K})$
- generate categorical observation x from c
  $x \sim P(X|C = c)$
- where $P(X|C = c) = \mathrm{Cat}(f_\theta(c))$

Generative models with neural networks

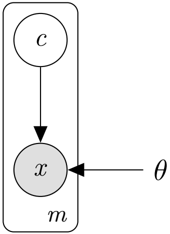
Mixture model



- sample a latent class $c \in \{1, \dots, K\}$
  $c \sim U(\frac{1}{K})$
- generate categorical observation x from c
  $x \sim P(X|C = c)$
- where $P(X|C = c) = \mathrm{Cat}(f_\theta(c))$
  - e.g. $f_\theta(c) = \mathrm{softmax}(g_\theta(c))$

Generative models with neural networks

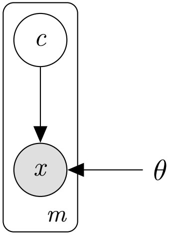
Mixture model



- sample a latent class $c \in \{1, \dots, K\}$
  $c \sim U(\frac{1}{K})$
- generate categorical observation x from c
  $x \sim P(X|C = c)$
- where $P(X|C = c) = \mathrm{Cat}(f_\theta(c))$
  - e.g. $f_\theta(c) = \mathrm{softmax}(g_\theta(c))$
    and $g_\theta(c) = \tanh(Wr(c) + b)$

Generative models with neural networks

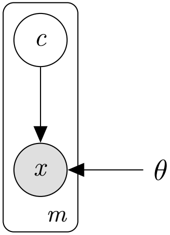
Mixture model



- sample a latent class $c \in \{1, \ldots, K\}$
  $c \sim U(\frac{1}{K})$
- generate categorical observation x from c
  $x \sim P(X|C = c)$
- where $P(X|C = c) = \mathrm{Cat}(f_\theta(c))$
  - e.g. $f_\theta(c) = \mathrm{softmax}(g_\theta(c))$
    and $g_\theta(c) = \tanh(Wr(c) + b)$
    and $r(c) = Ec$

Generative models with neural networks

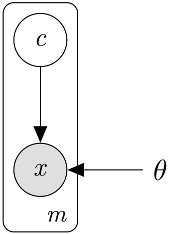
Mixture model



- sample a latent class $c \in \{1, \dots, K\}$
  $c \sim U(\frac{1}{K})$
- generate categorical observation x from c
  $x \sim P(X|C = c)$
- where $P(X|C = c) = \mathrm{Cat}(f_\theta(c))$
  - e.g. $f_\theta(c) = \mathrm{softmax}(g_\theta(c))$
    and $g_\theta(c) = \tanh(Wr(c) + b)$
    and $r(c) = Ec$
- with $\theta = (E, W, b)$

Generative models with neural networks

Mixture model

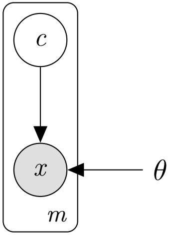


- sample a latent class $c \in \{1, \ldots, K\}$
  $c \sim U(\frac{1}{K})$
- generate categorical observation x from c
  $x \sim P(X|C = c)$
- where $P(X|C = c) = \mathrm{Cat}(f_\theta(c))$
  - e.g. $f_\theta(c) = \mathrm{softmax}(g_\theta(c))$
    and $g_\theta(c) = \tanh(Wr(c) + b)$
    and $r(c) = Ec$
- with $\theta = (E, W, b)$

$$P(x) = \underbrace{\sum_{c=1}^{K} \underbrace{P(c)P(x|c)}_{\text{differentiable function of } \theta}}_{\text{tractable for small } K}$$

Generative models with neural networks

Mixture model



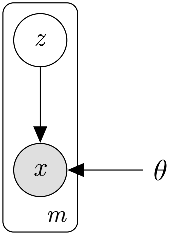
- sample a latent class $c \in \{1, \ldots, K\}$
  $c \sim U(\frac{1}{K})$
- generate categorical observation x from c
  $x \sim P(X|C = c)$
- where $P(X|C = c) = \mathrm{Cat}(f_\theta(c))$
  - e.g. $f_\theta(c) = \mathrm{softmax}(g_\theta(c))$
    and $g_\theta(c) = \tanh(Wr(c) + b)$
    and $r(c) = Ec$
- with $\theta = (E, W, b)$

$$P(x) = \underbrace{\sum_{c=1}^{K} \underbrace{P(c)P(x|c)}_{\text{differentiable function of } \theta}}_{\text{tractable for small } K}$$

Gradient-based optimisation! $\nabla_\theta \log P_\theta(x)$

Generative models with neural networks

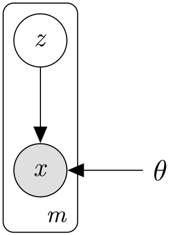
Continuous mixture model



Generative models with neural networks

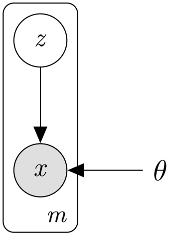
Continuous mixture model

- sample a latent embedding $z \in \mathbb{R}^d$
  $z \sim \mathcal{N}(0, 1)$



Generative models with neural networks

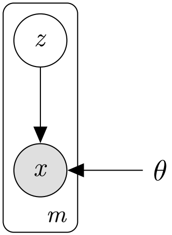
Continuous mixture model



- sample a latent embedding $z \in \mathbb{R}^d$
  $z \sim \mathcal{N}(0, 1)$
- generate categorical observation x from z
  $x \sim P(X|Z = z)$

Generative models with neural networks

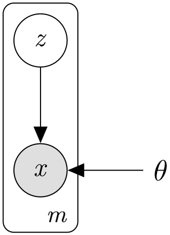
Continuous mixture model



- sample a latent embedding $z \in \mathbb{R}^d$
  $z \sim \mathcal{N}(0, 1)$
- generate categorical observation x from z
  $x \sim P(X|Z = z)$
- where $P(X|Z = z) = \mathrm{Cat}(f_\theta(z))$

Generative models with neural networks

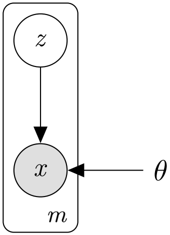
Continuous mixture model



- sample a latent embedding $z \in \mathbb{R}^d$
  $z \sim \mathcal{N}(0, 1)$
- generate categorical observation x from z
  $x \sim P(X|Z = z)$
- where $P(X|Z = z) = \mathrm{Cat}(f_\theta(z))$
  - e.g. $f_\theta(z) = \mathrm{softmax}(g_\theta(z))$

Generative models with neural networks

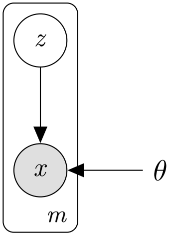
Continuous mixture model



- sample a latent embedding $z \in \mathbb{R}^d$
  $z \sim \mathcal{N}(0, 1)$
- generate categorical observation x from z
  $x \sim P(X|Z = z)$
- where $P(X|Z = z) = \mathrm{Cat}(f_\theta(z))$
  - e.g. $f_\theta(z) = \mathrm{softmax}(g_\theta(z))$
    and $g_\theta(z) = \tanh(Wz + b)$

Generative models with neural networks

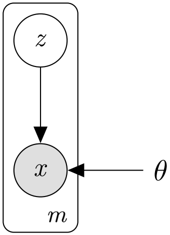
Continuous mixture model



- sample a latent embedding $z \in \mathbb{R}^d$
  $z \sim \mathcal{N}(0, 1)$
- generate categorical observation x from z
  $x \sim P(X|Z = z)$
- where $P(X|Z = z) = \mathrm{Cat}(f_\theta(z))$
  - e.g. $f_\theta(z) = \mathrm{softmax}(g_\theta(z))$
    and $g_\theta(z) = \tanh(Wz + b)$
- with $\theta = (E, W, b)$

Generative models with neural networks

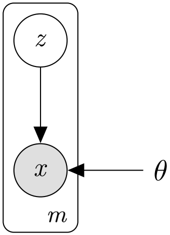
Continuous mixture model



- sample a latent embedding $z \in \mathbb{R}^d$
  $z \sim \mathcal{N}(0, 1)$
- generate categorical observation x from z
  $x \sim P(X|Z = z)$
- where $P(X|Z = z) = \mathrm{Cat}(f_\theta(z))$
  - e.g. $f_\theta(z) = \mathrm{softmax}(g_\theta(z))$
    and $g_\theta(z) = \tanh(Wz + b)$
- with $\theta = (E, W, b)$
- Intractability

Generative models with neural networks

Continuous mixture model



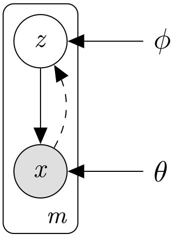
- sample a latent embedding $z \in \mathbb{R}^d$
  $z \sim \mathcal{N}(0, 1)$
- generate categorical observation x from z
  $x \sim P(X|Z = z)$
- where $P(X|Z = z) = \mathrm{Cat}(f_\theta(z))$
  - e.g. $f_\theta(z) = \mathrm{softmax}(g_\theta(z))$
    and $g_\theta(z) = \tanh(Wz + b)$
- with $\theta = (E, W, b)$
- Intractability
  - $P(x) = \int p(z)P(x|z)\mathrm{d}z$
  - $P(z|x) = \frac{p(z)P(x|z)}{\int p(z')P(x|z')\mathrm{d}z'}$

Variational inference

but we know VI :D

Variational inference

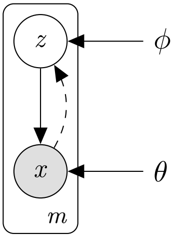
but we know VI :D



- approximate the posterior with $q_\phi(Z|x) = \mathcal{N}(\mu_\phi(x), I\sigma^2_\phi(x))$

Variational inference

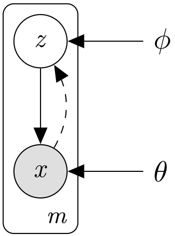
but we know VI :D



- approximate the posterior with $q_\phi(Z|x) = \mathcal{N}(\mu_\phi(x), I\sigma^2_\phi(x))$
- where
  - $\mu_\phi(x) = u_\phi(r_\phi(x))$
    e.g. $r_\phi(x) = E_\mathsf{u} x$
    and $u_\phi(r) = \tanh(W_\mathsf{u} r + b_\mathsf{u})$

Variational inference

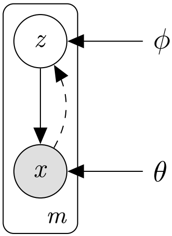
but we know VI :D



- approximate the posterior with $q_\phi(Z|x) = \mathcal{N}(\mu_\phi(x), I\sigma^2_\phi(x))$
- where
  - $\mu_\phi(x) = u_\phi(r_\phi(x))$
    e.g. $r_\phi(x) = E_{\mathsf{u}} x$
    and $u_\phi(r) = \tanh(W_{\mathsf{u}} r + b_{\mathsf{u}})$
  - $\sigma^2_\phi(x) = \exp(v_\phi(r'_\phi(x)))$
    e.g. $r'_\phi(x) = E_{\mathsf{v}} x$
    and $v_\phi(r) = \tanh(W_{\mathsf{v}} r + b_{\mathsf{v}})$

Variational inference

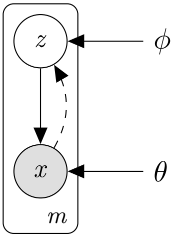
but we know VI :D



- approximate the posterior with $q_\phi(Z|x) = \mathcal{N}(\mu_\phi(x), I\sigma^2_\phi(x))$
- where
  - $\mu_\phi(x) = u_\phi(r_\phi(x))$
    e.g. $r_\phi(x) = E_{\mathsf{u}}x$
    and $u_\phi(r) = \tanh(W_{\mathsf{u}}r + b_{\mathsf{u}})$
  - $\sigma^2_\phi(x) = \exp(v_\phi(r'_\phi(x)))$
    e.g. $r'_\phi(x) = E_{\mathsf{v}}x$
    and $v_\phi(r) = \tanh(W_{\mathsf{v}}r + b_{\mathsf{v}})$
- with $\phi = (E_{\mathsf{u}}, E_{\mathsf{v}}, W_{\mathsf{u}}, W_{\mathsf{v}}, b_{\mathsf{u}}, b_{\mathsf{v}})$

Variational inference

but we know VI :D



- approximate the posterior with $q_\phi(Z|x) = \mathcal{N}(\mu_\phi(x), I\sigma^2_\phi(x))$
- where
  - $\mu_\phi(x) = u_\phi(r_\phi(x))$
    e.g. $r_\phi(x) = E_{\mathsf{u}}x$
    and $u_\phi(r) = \tanh(W_{\mathsf{u}}r + b_{\mathsf{u}})$
  - $\sigma^2_\phi(x) = \exp(v_\phi(r'_\phi(x)))$
    e.g. $r'_\phi(x) = E_{\mathsf{v}}x$
    and $v_\phi(r) = \tanh(W_{\mathsf{v}}r + b_{\mathsf{v}})$
- with $\phi = (E_{\mathsf{u}}, E_{\mathsf{v}}, W_{\mathsf{u}}, W_{\mathsf{v}}, b_{\mathsf{u}}, b_{\mathsf{v}})$

Mean field assumption

- $q_{\phi_i}(Z|x_i)$ is specified for each observation x_i by locally predicting its mean and variance

Approximate inference by optimisation

Maximise ELBO

$$\log P_\theta(x) \geq \underbrace{\mathbb{E}_{q_\phi(Z|x)}\left[\log \frac{q_\phi(Z|x)}{p_\theta(Z)}\right]}_{-\mathrm{KL}(q_\theta(Z|x)||p_\theta(Z))} + \underbrace{\mathbb{E}_{q_\phi(Z|x)}\left[\log P_\theta(X = x|Z)\right]}_{\text{intractable!}}$$

Approximate inference by optimisation

Maximise ELBO

$$\log P_\theta(x) \geq \underbrace{\mathbb{E}_{q_\phi(Z|x)}\left[\log \frac{q_\phi(Z|x)}{p_\theta(Z)}\right]}_{-\mathrm{KL}(q_\theta(Z|x)||p_\theta(Z))} + \underbrace{\mathbb{E}_{q_\phi(Z|x)}\left[\log P_\theta(X=x|Z)\right]}_{\text{intractable!}}$$

Prior term

$$\mathrm{KL}(q_\phi(Z|x)||p_\theta(Z)) = -\frac{1}{2}\sum_{j=1}^{d}(1+\log\sigma_\phi^2(x)_j - \mu_\phi^2(x)_j - \sigma_\phi^2(x)_j)$$

Approximate inference by optimisation

Maximise ELBO

$$\log P_\theta(x) \geq \underbrace{\mathbb{E}_{q_\phi(Z|x)}\left[\log \frac{q_\phi(Z|x)}{p_\theta(Z)}\right]}_{-\mathrm{KL}(q_\theta(Z|x)||p_\theta(Z))} + \underbrace{\mathbb{E}_{q_\phi(Z|x)}\left[\log P_\theta(X = x|Z)\right]}_{\text{intractable!}}$$

Prior term

$$\mathrm{KL}(q_\phi(Z|x)||p_\theta(Z)) = -\frac{1}{2}\sum_{j=1}^{d}(1 + \log \sigma^2_\phi(x)_j - \mu^2_\phi(x)_j - \sigma^2_\phi(x)_j)$$

Likelihood term is intractable

- the Categorical likelihood is not conjugate with the Normal approximate posterior

Change of variable for location-scale distributions

For $Z \sim \mathcal{N}(\mu, \sigma^2)$ we can re-express Z in terms of $E \sim \mathcal{N}(0, I)$

- $Z = \mu + \sigma E$

Change of variable for location-scale distributions

For $Z \sim \mathcal{N}(\mu, \sigma^2)$ we can re-express Z in terms of $E \sim \mathcal{N}(0, I)$

- $Z = \mu + \sigma E$

then we can re-express expectations

$$\mathbb{E}_{\mathcal{N}(\mu,\sigma^2)}[f(Z)] = \mathbb{E}_{\mathcal{N}(0,I)}[f(\mu + \sigma E)]$$

Change of variable for location-scale distributions

For $Z \sim \mathcal{N}(\mu, \sigma^2)$ we can re-express Z in terms of $E \sim \mathcal{N}(0, I)$

- $Z = \mu + \sigma E$

then we can re-express expectations

$$\mathbb{E}_{\mathcal{N}(\mu,\sigma^2)}[f(Z)] = \mathbb{E}_{\mathcal{N}(0,I)}[f(\mu + \sigma E)]$$

back to the ELBO

$$\mathbb{E}_{q_\phi(Z|x)}\left[\log P(x|Z)\right] = \mathbb{E}_{\epsilon \sim N(0,I)}\left[\log P(x|Z = \mu_\phi(x) + \sigma_\phi(x)\epsilon))\right]$$

Monte Carlo estimate

$$
\begin{aligned}
\mathbb{E}_{q_\phi(Z|x)}\left[\log P(x|Z)\right] &= \mathbb{E}_{\epsilon\sim N(0,I)}\left[\log P(x|Z=\mu_\phi(x)+\sigma_\phi(x)\epsilon))\right] \\
&\approx \frac{1}{N}\sum_{n=1}^{N}\log P\left(x|\mu_\phi(x)+\sigma_\phi(x)\epsilon^{(n)}\right)
\end{aligned}
$$

MC estimate of the ELBO

$$\log P_\theta(x) \geq \underbrace{\mathbb{E}_{q_\phi(Z|x)}\left[\log \frac{q_\phi(Z|x)}{p_\theta(Z)}\right]}_{-\,\mathrm{KL}(q_\theta(Z|x)||p_\theta(Z))} + \underbrace{\mathbb{E}_{q_\phi(Z|x)}\left[\log P_\theta(X=x|Z)\right]}_{\text{intractable!}}$$

$$\approx \underbrace{\frac{1}{2}\sum_{j=1}^{d}\Big(1+\log\sigma_\phi^2(x)_j - \mu_\phi^2(x)_j - \sigma_\phi^2(x)_j\Big)}_{-\,\mathrm{KL}(q_\theta(Z|x)||p_\theta(Z))}$$

$$+ \underbrace{\log P_\theta\left(x|\mu_\phi(x)+\sigma_\phi(x)\epsilon\right)}_{\text{single-sample estimate}}$$

Gradient-based optimisation

Let $\mathcal{L}(\theta, \phi|x)$ be our objective function

$$\mathcal{L}(\theta, \phi|x) = \underbrace{\frac{1}{2}\sum_{j=1}^{d}\Big(1 + \log\sigma_\phi^2(x)_j - \mu_\phi^2(x)_j - \sigma_\phi^2(x)_j\Big)}_{\text{differentiable function of } \phi}$$

$$+ \underbrace{\log P_\theta\left(x|\mu_\phi(x) + \sigma_\phi(x)\epsilon\right)}_{\text{differentiable function of } \theta \text{ and } \phi}$$

Gradient-based optimisation

Let $\mathcal{L}(\theta, \phi|x)$ be our objective function

$$\mathcal{L}(\theta, \phi|x) = \underbrace{\frac{1}{2}\sum_{j=1}^{d}\left(1 + \log\sigma_\phi^2(x)_j - \mu_\phi^2(x)_j - \sigma_\phi^2(x)_j\right)}_{\text{differentiable function of } \phi}$$
$$+ \underbrace{\log P_\theta\left(x|\mu_\phi(x) + \sigma_\phi(x)\epsilon\right)}_{\text{differentiable function of } \theta \text{ and } \phi}$$

We can update θ and ϕ using stochastic gradient steps

- we know chain rule (thus we can get a gradient)
- we have a noisy though unbiased estimate
- guaranteed convergence to a local optimum of $\mathcal{L}$ (with appropriate learning rate schedule)

Further reading

- Auto-Encoding variational Bayes [Kingma and Welling, 2014]

References I

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.