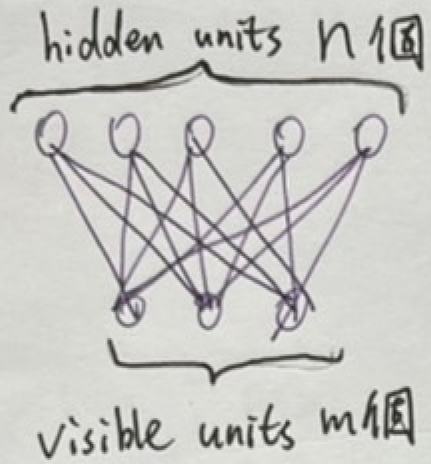
DLP Final Exam
2022/6/9
0716206
陳昱丞

1. RBM



$$E(v,h) = -b^T v - c^T h - v^T W h$$

$$P(v,h) = \frac{1}{Z}\exp(-E(v,h))$$

$$= \frac{1}{Z}\exp(b^T v + c^T h + v^T W h)$$

(a) 計算 $P(h|v)$：h層內沒有連接，h是否激活在給定 visible 的情況下條件獨立。

$$P(h|v) = \frac{P(v,h)}{P(v)} = \frac{1}{P(v)}\frac{1}{Z}\exp(b^T v + c^T h + v^T W h)$$

$$= \frac{1}{P(v)}\frac{1}{Z}\exp(b^T v)\cdot\exp(c^T h + v^T W h)$$

$$= \frac{1}{Z'}\exp(c^T h + v^T W h)$$

$$= \frac{1}{Z'}\exp\Big(\sum_{j=1}^{n}(c_j^T h_j + v^T W_{:,j} h_j)\Big)$$

$\Rightarrow$ 指數相加 = 分開相乘 $\Rightarrow P(h|v)$ is factorial.

$$P(h_j=1|v) = \frac{P(h_j=1|v)}{P(h_j=0|v)+P(h_j=1|v)} = \frac{\exp(c_j + v^T W_{:,j})}{\exp(0) + \exp(c_j + v^T W_{:,j})} = \frac{1}{1+\exp(-(c_j + v^T W_{:,j}))} = \sigma(c_j + v^T W_{:,j})$$

(b) 計算 $P(v|h)$： V層內沒有連接，V是否激活在給定 h 的情況下條件獨立。

$$P(V|h) = \frac{P(h,V)}{P(h)} = \frac{1}{P(h)} \frac{1}{Z} \exp(b^T V + C^T h + V^T W h)$$

$$= \frac{1}{P(h)} \frac{1}{Z} \exp(C^T h) \cdot \exp(b^T V + V^T W h)$$

$$= \frac{1}{Z'} \exp(b^T V + V^T W h)$$

$$= \frac{1}{Z'} \underbrace{\exp\left(\sum_{i=1}^{m} (b_i^T V_i + V_i^T W_{i,:} h)\right)}$$

$\Rightarrow$ 指數相加 = 分開相乘 $\Rightarrow P(V|h)$ is factorial.

$$P(V_i=1|h) = \frac{P(V_i=1|h)}{P(V_i=0|h) + P(V_i=1|h)} = \frac{\exp(b_i + W_{i,:}h)}{\exp(0) + \exp(b_i + W_{i,:}h)} = \frac{1}{1+\exp(-(b_i + W_{i,:}h))} = \sigma(b_i + W_{i,:}h)$$

(c) ① $P(h|v)$ 和 $P(V|h)$ 可以被用來計算所有樣本的梯度和。

② 但機率分佈有 2^{m+n} 種情況，計算量太大。因此要用 block Gibbs sampling 來 sample independent V 再求梯度。

2. (a) $P(X_1, X_2, X_3, X_4 \mid Z_1, Z_2, Z_3, Z_4) = P(X_1 \mid Z_1)\,P(X_2 \mid X_1, Z_2)\,P(X_3 \mid X_2, Z_3)\,P(X_4 \mid X_3, Z_4)$

$$\underline{P(X_{1-T} \mid Z_{1-T}) = \prod_{t=1}^{T} P(X_t \mid X_{t-1}, Z_t)} \ \#$$

(b) $P(Z_1, Z_2, Z_3, Z_4 \mid X_1, X_2, X_3, X_4) = P(Z_1)\,P(Z_2 \mid Z_1)\,P(Z_3 \mid Z_1, Z_2)\,P(Z_4 \mid Z_1, Z_2, Z_3)$

$$\underline{P(Z_{1-T} \mid X_{1-T})} \ /\!/ \ \underline{\prod_{t=1}^{T} P(Z_t \mid Z_{<t})} \ \#$$

(c) $P(Z_1, Z_2, Z_3, Z_4) = P(Z_1)\,P(Z_2 \mid Z_1)\,P(Z_3 \mid Z_1, Z_2)\,P(Z_4 \mid Z_1, Z_2, Z_3)$

$P(Z_2 \mid Z_1) = f_2(Z_2, Z_1) = P(Z_2)$

$P(Z_3 \mid Z_1, Z_2) = f_3(Z_3, Z_1, Z_2) = P(Z_3)$

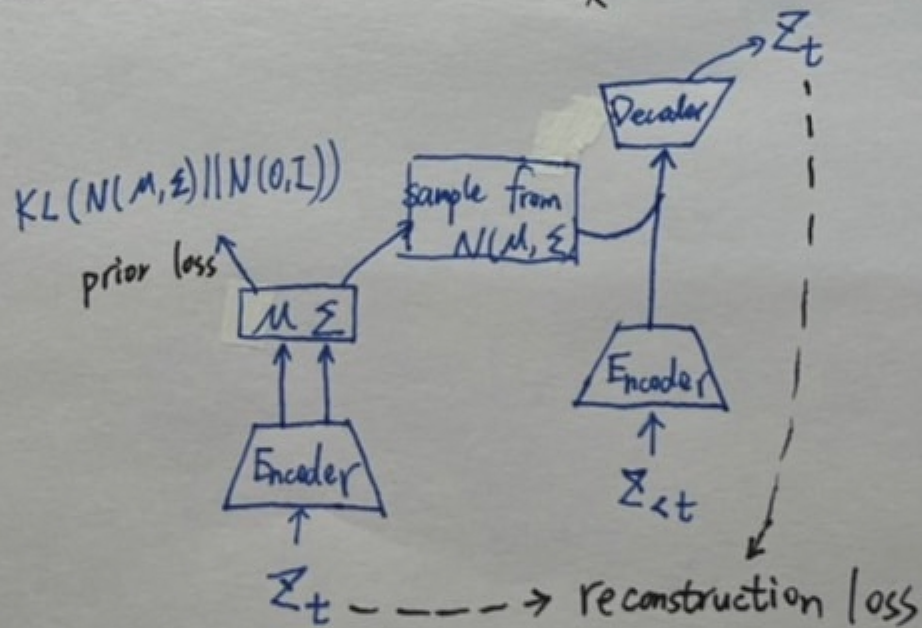
$P(Z_4 \mid Z_1, Z_2, Z_3) = f_4(Z_4, Z_1, Z_2, Z_3) = P(Z_4)$

$\Rightarrow P(Z_1, Z_2, Z_3, Z_4) = P(Z_1)\,P(Z_2)\,P(Z_3)\,P(Z_4)$

$$\underline{P(Z_{1-T}) = \prod_{t=1}^{T} P(Z_t \mid Z_{<t})} \ \#$$

(d) $$\underline{q(Z_{1-T} \mid X_{1-T}) = \prod_{t=1}^{T} q(Z_t \mid Z_{<t})} \ \#$$

(e) train VAE:



prior distribution:

$\underline{N(0, I)}$ #

Encoder、Decoder 為 RNN，因為 Z 有連續相關性。

objective function:

$$\underline{E_{y \sim q(y \mid Z_t; \theta')} \log P(Z_t \mid y; \theta) - KL(q(y \mid Z_t; \theta') \,\|\, N(0, I))} \ \#$$

(f) train flow model:

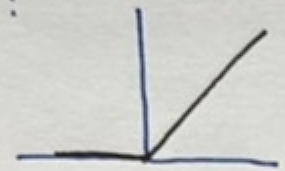
$T=3$

$$P(X_1) \xrightarrow{\boxed{G^{-1}}} P(Z_1)$$
$$P(X_2) \longrightarrow P(X_1, Z_2)$$
$$P(X_3) \longrightarrow P(X_2, Z_3)$$

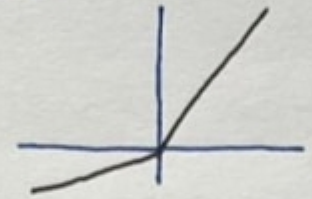
We trained G^{-1}, then we got flow model:

$$P(Z_1) \longrightarrow \boxed{G} \longrightarrow P(X_1),\ P(X_1, Z_2) \longrightarrow \boxed{G} \longrightarrow P(X_2),\ P(X_2, Z_3) \longrightarrow \boxed{G} \longrightarrow P(X_3)$$

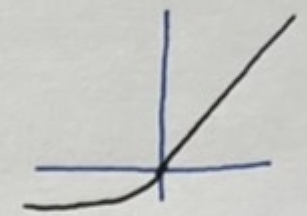
3. (a) ReLU:



Leaky ReLU:



ELU:



這三個 activation function 差異在當 $x<0$ 時，

- ReLU：直接輸出 0，沒有梯度。
- LeakyReLU：一條斜率很小的直線。
- ELU：幅度很小的曲線。

(b) dropout：隨機棄用一些 hidden unit 不參與計算。是一種 regularization 手段。

(c) 在 training 時使用 dropout，代表計算整個 network 的 subnetworks；
而在 testing 時不使用 dropout，就代表在計算 training 時那些 multiple subnetworks 的期望值。

(d) ReLU 因為在 $x<0$ 時會沒有梯度存在，因此可能會有 gradient vanish 的問題。
解決的方式是改用 (a) 所提到的 LeakyReLU 或 ELU。

(e)

$$\text{floor}\left(\frac{W+2\times pad-ks}{S}\right)+1$$

$$\Rightarrow \text{floor}\left(\frac{256+2\times2-3}{2}\right)+1 = \text{floor}\left(\frac{257}{2}\right)+1 = 128+1=129$$

feature map = (129, 129)

4. VAE

(a) main idea :

Reconstruction term :

latent code z generated by encoder $q(z|x;\theta')$ for input x should maximize the log-likelihood $\log P(x|z;\theta)$ for x

$$\Rightarrow E_{z\sim q(z|x;\theta')} \log P(x|z;\theta)$$

Regularization term :

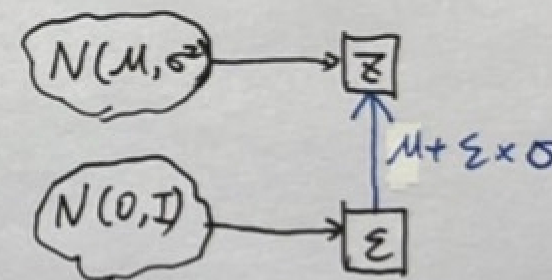
conditional distribution $q(z|x;\theta')$ of the latent code z given x should be compatible with the prior $p(z)$

$$\Rightarrow KL(q(z|x;\theta') \| P(z))$$

objective function : $E_{z\sim q(z|x;\theta')} \log P(x|z;\theta) - \underbrace{KL(q(z|x;\theta') \| P(z))}_{\text{KL divergence 恆} \geq 0} \leq \log P(x)$

(b) $q(z|x;\theta')$ and $P(z)$ are assumed to be Gaussians $\Rightarrow$ KL divergence tractable

(c) sample z from $N(\mu, \sigma^2)$ = sample ε from $N(0, I)$ and let $z = \mu + \varepsilon \times \sigma$



(d) True.

$\because KL(q(z|x) \| P(z)) = 0$ when $q(z|x) = P(z)$

$\therefore$ objective function $E_{z\sim q(z|x;\theta')} \log P(x|z;\theta) - \underbrace{KL(q(z|x;\theta') \| P(z))}_{=0} \leq \log P(x)$

$\Rightarrow E_{z\sim q(z|x;\theta')} \log P(x|z;\theta) \leq \log P(x)$ $\Rightarrow$ 左式越大，越接近右式

(e) GMVAE's $L_{ELBO} = \underbrace{E_{q(x|y)}[\log P_\theta(y|x)]}_{\text{reconstruction term}} - \underbrace{E_{q(w|y)p(z|x,w)}[KL(q_{\phi x}(x|y) \| P_\beta(x|w,z)]}_{\text{conditional prior term}} - \underbrace{KL(q_{\phi w}(w|y) \| P(w))}_{\text{w-prior term}}$

$- \underbrace{E_{q(x|y)q(w|y)}[KL(P_\beta(z|x,w) \| P(z)]}_{\text{z-prior term}}$

KL divergence in VAE $(KL(q(z|x) \| P(z)))$ becomes conditional prior term + w-prior term + z-prior term

5. grand truth learned

$P(z)$ $q(z;\theta)$

↑ ↑

two peak unimodal distribution : one peak

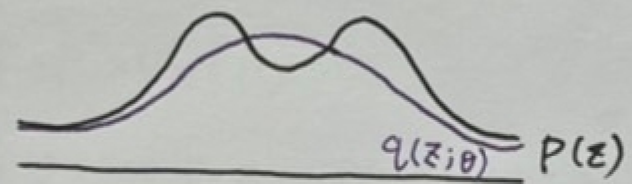
$$KL(p\|q) = -\int p(x) \log\left[\frac{q(x)}{p(x)}\right] dx$$

$$\cong \mathop{E}_{x\sim p}[\log P(x) - \log q(x)]$$

① $\min(KL(P(z)\|q(z;\theta))) = \min\left(\mathop{E}_{z\sim p}[\log P(z) - \log q(z;\theta)]\right)$

$= \min\left(\mathop{E}_{z\sim p}\left[\log\left(\frac{P(z)}{q(z;\theta)}\right)\right]\right)$ 已知 KL divergence 必為 ≥ 0，

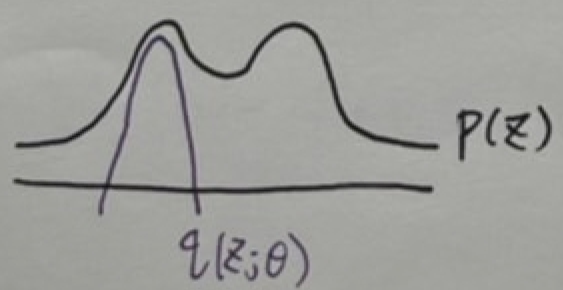
$\frac{P(z)}{q(z;\theta)}$ 越接近 1，取 log 後越接近 0 ⇒ 越小



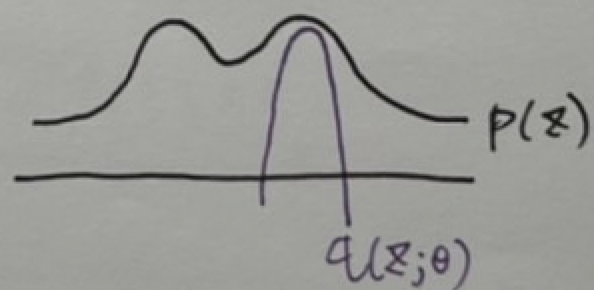
② $\min(KL(q(z;\theta)\|P(z))) = \min\left(\mathop{E}_{z\sim q}[\log q(z;\theta) - \log P(z)]\right)$

$= \min\left(\mathop{E}_{z\sim q}\left[\log\left(\frac{q(z;\theta)}{P(z)}\right)\right]\right)$

對 q 做 sample 只能局部接近 P，沒辦法知道 P 的整体

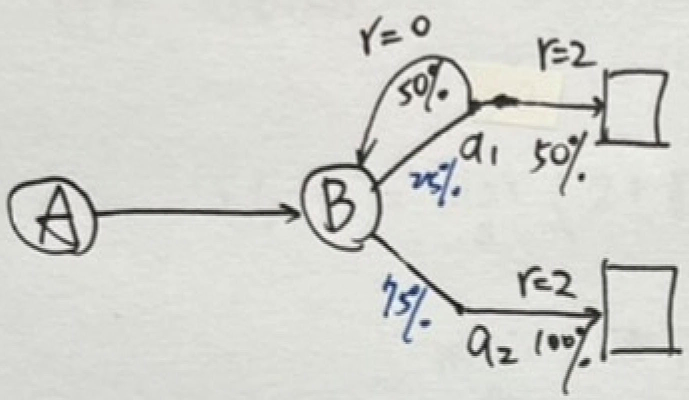


or



RL

1.



(a) $\pi(B, a_1) = 25\%$, $\pi(B, a_2) = 75\%$

(i) $\gamma = 1$ $\quad V_\pi(A) = 4 + \gamma \cdot V_\pi(B)$

$V_\pi(B) = R^\pi + \gamma \times P^\pi \times V_\pi(B)$

$= 0.75 \times 2 + 0.25 \times 0.5 \times 2 + 0.25 \times 0.5 \times V_\pi(B)$

$= 1.75 + 0.25 V_\pi(B)$

$\Rightarrow V_\pi(B) = \frac{1.75}{1-0.25} = 2$ (by 無窮等比)

$V_\pi(A) = 4 + 2 = \underline{6}$ ※

(ii) $\gamma = 0.5$ $\quad V_\pi(B) = 1.75 + 0.0625 V_\pi(B)$

$\Rightarrow V_\pi(B) = \frac{1.75}{1-0.0625} = 1.86667$ (by 無窮等比)

$V_\pi(A) = 4 + 0.5 \times 1.86667 = \underline{4.933335}$ ※

(b) $\gamma = 0.5$

(i) $V_*(A) = 4 + \gamma V_*(B)$

$= 4 + 0.5 V_*(B)$

$$V_*(B) = \max_a \Big(\underbrace{2 + \gamma \times 0}_{\text{choose } a_2},\ \underbrace{0.5 \times 2 + \gamma \times 0 + 0.5 \times 0 + 0.5 \times \gamma \times V_*(B)}_{\text{choose } a_1} \Big)$$

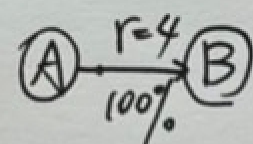
$$= \max_a \big(2,\ 1 + 0.25 V_*(B)\big) \Rightarrow \max_a \Big(2,\ \underbrace{\frac{1}{1-0.25}}_{\text{by 無窮等比}}\Big) = \max_a (2,\ 1.333) = 2$$

$V_*(A) = 4 + 0.5 \times 2 = \underline{5}$ #

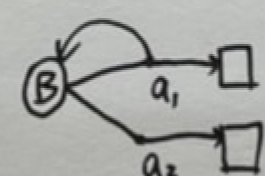
(ii) Optimal policy π' ⇒ choose the best action (greedy)

original policy π
improved policy π' } $\pi'(s) = \arg\max_{a \in A} q_\pi(s, a)$

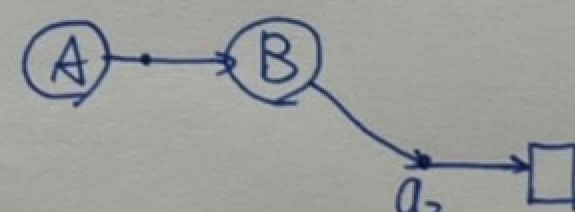
當其他所有(state, action)都不動，只對其中一個 state s 取出更好(or 維持)的 action 並更新，則整体的 policy 只有可能更好(or 維持)。

以此題來說 

沒有其他路線。



承上題(i)，選 a_2 會是較好的選擇。

因此 optimal policy 就是 

RL

2. (a)

exploration in DQN : ε-greedy : $\begin{cases} 1-\varepsilon \rightarrow \text{choose the greedy (best) action} \\ \varepsilon \rightarrow \text{randomly choose other action} \end{cases}$

DDPG : select action $a_t = \mu(s_t|\theta^{\mu}) + \underbrace{N_t}_{\text{add a noise}}$

(b)

DQN 中使用 target network 是為了避免 behavior network 每次更新，取出來的值會一直浮動。target network 只取值不做 gradient update，可以讓取出來的值更加穩定。