According to the graphical model $p(X_{1:T}|Z_{1:T})$ as $\prod_{t=1}^{T} p(X_t|\dots)$? Apply d-separation: $p(x_1,x_2,x_3,x_4|z_1,z_2,z_3,z_4)=p(x_1|z_1)p(x_2|x_1,z_2)p(x_3|x_2,z_3)p(x_4|x_3,z_4)$

↳ $q(X_{1:T}|Z_{1:T})$ as $\prod_{t=1}^{T} q(Z_t|\dots)$ $\Rightarrow q(z_1,z_2,z_3,z_4|x_1,x_2,x_3,x_4)=q(z_1|x_1)\cdot q(z_2|x_2,z_1)\cdot q(z_3|x_3,z_1,z_2)\cdot q(z_4|x_4,z_1,z_2,z_3)=\prod_{t=1}^{T} q(z_t|x_{t-1},z_{t-1})$

(a) factorize $p(Z_{1:T}|X_{1:T}) \Rightarrow p(Z_1|X_1)\cdot p(Z_2|Z_1,X_1)\cdot p(Z_3|Z_1,Z_2,X_3)\cdot p(Z_4|Z_1,Z_2,Z_3,X_4)$ (b) joint distribution $p(Z_{1:T})=p(Z_1)p(Z_2|Z_1)\cdot p(Z_3|Z_1,Z_2)p(Z_4|Z_1Z_2Z_3)$

Describe how to evaluate the probability $p(X_1,X_2,X_3)$ of ... $p(x_1,x_2,x_3)=p(z_1)\left|\frac{\partial z_1}{\partial x_1}\right| p(z_2|z_1)\left|\frac{\partial z_2}{\partial x_2}\right| p(z_3|z_2)\left|\frac{\partial z_3}{\partial x_3}\right|$

How would train the flow? maximum log likelihood $L=E_x[\log p(x)]$ or find $\arg\min_\theta E_x[-\log p_\theta(x)] = E_x[-\log p(f_\theta(x))-\log\det\left|\frac{\partial f_\theta(x)}{\partial x}\right|]$

Sample x_i from z_i using flow model $\Rightarrow$ use trained inversed function f^{-1} to generate x_i from z_i ① $x_3,z_3=f_3^{-1}(z_3)$ ② $x_2,z_2=f_2^{-1}(z_2)$ ③ $x_1=f_1^{-1}(z_1)$

Design an invertible mapping from $x_i \to z_i$: flow model use invertible $f_1 \to z_1=f_1(x_1)$, $f_2 \to z_2=f_2(x_2,z_2)$, $f_3 \to z_3=f(x_3,z_3)$

Design an encoding distribution $q(z_{1:T}|x_{1:T})$ to approx $p(z_{1:T}|x_{1:T})$, z_t based on $x_{\le t}, z_{\le t}$, factorize of $q(z_{1:T}|x_{1:T}) = \prod_{t=1}^{T} q(z_t|x_{1:t},z_{<t},x_{<t})$

Train this latent model by maximum ELBO. Describe network CNN for encoding, decoding for prior distribution/training object

↳ Encoder = CNN $x_{1:T} \to z_{1:T}$, Decoder $z_{1:T} \to x_{1:T}$ objective = Maximum ELBO, prior distribution = Gaussian distribution, ELBO: $E_{q(z_{1:T}|x_{1:T})}[\log p(x_{1:T}|z_{1:T})] - KL(q(z_{1:T}|x_{1:T})\|p(x_{1:T}))$

dropout = 隨機棄用一些 hidden unit 不參與計算。是一種 regularization 手段

gradient vanish problem: Cause ① activation 導數太小 ex: sigmoid 在極端值的導數 $\to 0$ solve 使用 ReLU activation, batch normalization Residual Network

② 網路太深，前面梯度極小

pooling = 用於減少 feature graph 的維度，從而降低計算複雜度，同時保持每特徵

CNN contribution = ① Local Connectivity: CNN 利用局部連接來捕捉局部的特徵

② Parameter Sharing: 卷積層中權重在不同的空間位置共享，減少參數量，提高效率

Training VAE

try to maximize a ELBO. Explain main idea, variational lower bound: $L: E_{z\sim g(z|x;\theta')}[\log p(x|z;\theta)] - KL(g(z|x;\theta')\|p(z))$ (Reconstruction loss, regularization loss)

$KL(q(z|x)\|p(z)) = E_{q(z|x)}\left[\log \frac{q(z|x)}{\sum_{k=1}^{K}\pi_k \mathcal{N}(z|\mu_k,\Sigma_k)}\right]$

What distribution approximate $q(z|x)$: Gaussian distribution, this is an assumption

Explain the notion of re-param trick: 將隨機變量 $z\sim g(z|x)$ 重寫成一個確定性變量的函數 $z=g(\varepsilon,x)$，ε 是從高斯分佈抽樣

Posterior $q(z|x)=p(z)$? ans: False. when maximum variational lower bound, $g(z|x)$ shouldn't be identity to $p(z)$, However it should catch the posterior distribution

How would you evaluate KL divergence if prior $p(z)$ is replace with Gaussian Mix... $KL(q(z|x)\|p(z)) = E_{q(z|x)}\left[\log \frac{q(z|x)}{p(z)}\right]$ substitute $p(z)$ with GMM

VAE	stochastic	maximum	variational lower bound	✓	✓ Reconstruction loss, KL Divergence
GAN	stochastic	minimax	loss	✗	✗ Fréchet Inception Distance (FID)
WGAN	stochastic	minimize	Earth Mover's Distance	✗	✗ Wasserstein distance
FLOW	deterministic	maximum	log likelihood	✓	✓ log likelihood
Diffusion	stochastic	minimize	loss that combine reconstruction at each step	✓	✓ FID
AE	deterministic	minimize	reconstruction loss	✗	✗ Reconstruction loss

↳ generating latent code of specific data sample

↳ Evaluate Probability of spectic data sample

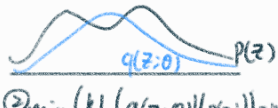
In evaluating KL divergence between ground-truth $p(z)$ and learned distribution $q(z)$ ... while $p(z)$ has 2 peak.

GT	learned
$p(z)$	$q(z;\theta)$
↓ two peak	↑ unimodal distribution: one peak

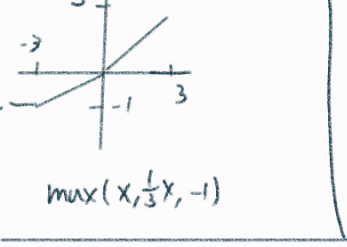
$KL(p\|q) = -\int p(x)\log\left[\frac{q(x)}{p(x)}\right]dx \cong E_{x\sim p}[\log p(x) - \log q(x)]$

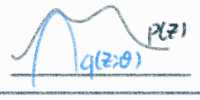
① $\min(KL(p(z)\|q(z;\theta))) = \min_{z\sim p}(E[\log p(z) - \log q(z;\theta)]) = \min_{z\sim p}\left(E\left[\log\left(\frac{p(z)}{q(z;\theta)}\right)\right]\right)$, KL Divergence ≥ 0

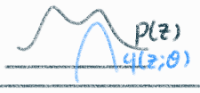
$\frac{p(z)}{q(z;\theta)}$ 越接近 1，取 log 後越接近 0



② $\min(KL(q(z;\theta)\|p(z))) = \min_{z\sim q}(E[\log q(z;\theta) - \log p(z)]) = \min_{z\sim q}\left(E\left[\log\left(\frac{q(z;\theta)}{p(z)}\right)\right]\right)$. 因 q 做 sample 只能局部接近 p 沒辦法知道 p 的整體







train VAE prior distribution $N(0,I)$ Reparameterization for end-to-end training

$E_{z\sim q(z|x;\theta')}[\log p(x|z;\theta)]$

$-KL(q(z|x;\theta')\|p(z))$

Reconstruction loss (Encoder-Decoder 為 RNN 因為 z 有連續性)

ReLU ✗ 差量 $x<0$ 輸出 0，無梯度

LeakyReLU ✗ 一條很小斜率的直線

ELU ✗ 幅度小的曲線

Sigmoid ✗ $\frac{1}{1+e^{-x}}$

Conv feature map (K_h, K_w), stride (S_h, S_w), padding (P_h, P_w)

$H_{out} = \left\lfloor \frac{H_{in} - K_h + 2\times P_h}{S_h} \right\rfloor + 1$

$W_{out} = \left\lfloor \frac{W_{in} - K_w + 2\times P_w}{S_w} \right\rfloor + 1$

parameters $= K_h \cdot K_w \cdot \text{Input}_C \cdot \text{Output}_C$

Compare RNN-based, VAE, GAN, flow model, diffusion model

allow exact evaluation of likelihood: VAE、Flow-based、RNN

training 的方：RNN：最大化下個時間步驟的條件機率

VAE：最大化 ELBO

GAN：最小化生成器和 discriminator 的對抗損失，最大化 JS-divergence

Flow-base：maximum log likelihood

Diffusion：最小化每一步 Diffusion 的重構誤差，通常也包含 KL-divergence

in terms of generation process, describe how to draw sample

RNN-based auto regressive：依賴前 timestep 的 data 對成下一個 timestep 的 data

VAE：在 latent space 中采樣，生成真實 data 的圖片

Flow-based：通過可逆將 latent space 樣本轉為 data 樣本，這個過程是確定性的

Diffusion：逐步降噪生成圖片

major advantages of diffusion as compared to GAN

Diffusion 可以生成更高質量的圖，訓練不依賴於 generator 和 discriminator 更穩定

過大的步長使每一步的 noise 變化過大，導致逆向難以準確重構

GAN vs WGAN

	Critic
GAN → Maximum JS-divergence	相似
WGAN → Minimize Wasserstein Distance	

Transformer vs. RNN

Transformer 是一種基於注意力機制的組跨架構，特別適用於處理序列資料

RNN 區別：Transformer 可以並行處理，RNN 要逐個 timestep 處理

應用例子：Transformer 架構在 NLP 任務中非常成功，如 GPT

Dropout vs L_1/L_2 regularization：L1, L2 正則化對 loss function 做修改，Dropout 直接對網路做隨機忽略

應用例子：CNN 中 Fully Connect 之後使用 Dropout 減少 overfitting.

padding：在輸入的圖邊界添加 extra pixel 以控制輸出 feature map 大小，for instance：Kernel size (3,3) Img_sz (5,5)

Input sz=Output sz Same padding Output_sz (5,5)

Thin padding. No padding Output_sz (3,3)

attention

↳ self-attention 計算輸入的元素和其他元素之間的關係來重新加權

↳ Multi-head-attention 將 self-attention 利用多次，並將結果整合，以得到更豐富的特徵

例子：NLP 中 attention 可以使 model 專注於句子中當前單字最相關的部分

對比：CNN 通常用於圖像處理，但 attention 也能用於捕捉相同區域間的依賴性

RNN：attention 有利於捕捉更長距依賴的關係