Variational approximation(Continued)

Let $L(Q,\lambda) = Q(q,g) + \lambda(\sum_z q_z - 1)$, then

$$\frac{\partial L(Q,\lambda)}{\partial q_j} = \log g(y,j) - \log q_j - 1 + \lambda$$

$$\implies q_j^* \equiv \alpha g(y,j)$$

$$\sum_z q_z = 1 : q_j^* = \frac{g(y,j)}{\sum_z g(y,z)}$$

Substituting q^* in the original maximization equation, we get $\log \sum_{z=1}^{k} g(y,z)$.
Hence LHS=RHS.
For all other q,

$$\log \sum_{z=1}^{k} g(y,z) \geq \sum_{z=1}^{k} q_z \log g(y,z) - \sum_z q_z \log q_z$$

Applying variational approximation

$$\theta^{ML} \equiv arg\max_{\theta} \sum_{i=1}^{N} \log \sum_{\mathbf{z}:z_1,\ldots,m} P(\mathbf{y}^i, \mathbf{z}|\theta, \mathbf{x}^i)$$

$$\equiv arg\max_{\theta} \sum_{i=1}^{N} \max_{q_{i,\mathbf{z}}:\sum_{\mathbf{z}} q_{i,\mathbf{z}}=1} \sum_{\mathbf{z}} q_{i,\mathbf{z}} \log P(\mathbf{y}^i, \mathbf{z}|\theta, \mathbf{x}^i) - \sum_{\mathbf{z}} q_{i,\mathbf{z}} \log q_{i,\mathbf{z}}$$

Solve by alternating optimization.

EM algorithm

E-Step: Solve for $q_{i\mathbf{z}}$ keeping θ fixed at θ^t

$$q^t_{i,\mathbf{z}} \equiv \frac{P(\mathbf{y}^i, \mathbf{z}|\theta^t, \mathbf{x}^i)}{\sum_{\mathbf{z}} P(\mathbf{y}^i, \mathbf{z}|\theta^t, \mathbf{x}^i)}$$
$$\equiv P(\mathbf{z}|\theta^t, \mathbf{x}^i, \mathbf{y}^i)$$

Hence $q_{i\vec{z}}$ is the posterior distribution of hidden variable at time t.
M-Step: Solve for θ, keeping $q_{i,\mathbf{z}}$ fixed to $q^t_{i,\mathbf{z}}$ The problem becomes

$$\max_{\theta} \sum_i \sum_{\mathbf{z}} q^t_{i,\mathbf{z}} \log P(\mathbf{y}^i, \mathbf{z}|\theta, \mathbf{x}^i)$$

which is concave in θ and can be often solved in closed form.(Example:HMM)

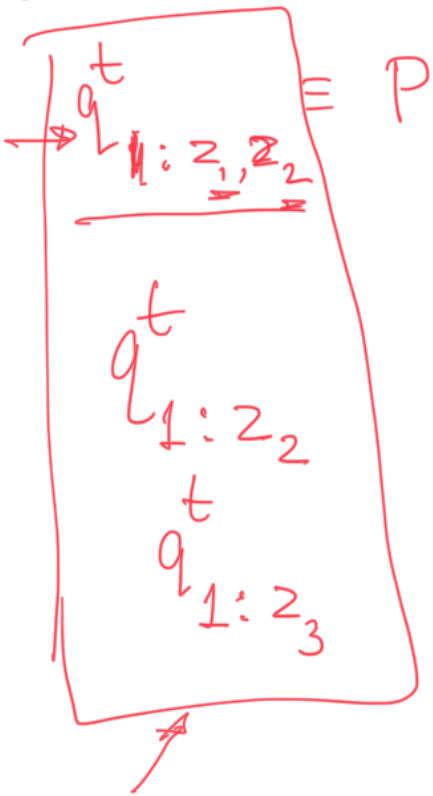
EM algorithm for graphical models

Above formulation may lead to lot of computations for calculating $q^t_{i,\mathbf{z}}$ depending on the possible values of $\mathbf{z}$. In graphical models, we don't need to directly compute all $q_{i,\mathbf{z}}$.

M-Step:

$$
\begin{aligned}
& \max_\theta \sum_i \sum_{\mathbf{z}} q^t_{i\mathbf{z}} \log P(\mathbf{y}^i, \mathbf{z}|\theta, \mathbf{x}^i) \\
= & \max_\theta \sum_i \sum_{\mathbf{z}} q^t_{i,\mathbf{z}} (\log \exp(\sum_C F_\theta(\mathbf{y}^i_C, \mathbf{z}_C|\mathbf{x}^i)) - \log Z_\theta(\mathbf{x}^i)) \\
= & \max_\theta \sum_i \sum_{\mathbf{z}} q^t_{i,\mathbf{z}} (\sum_C F_\theta(\mathbf{y}^i_C, \mathbf{z}_C|\mathbf{x}^i)) - \sum_i \log Z_\theta(\mathbf{x}^i) \sum_{\mathbf{z}} q^t_{i,\mathbf{z}} \\
= & \max_\theta \sum_i \sum_C \sum_{\mathbf{z}_C} F_\theta(\mathbf{y}^i_C, \mathbf{z}_C|\mathbf{x}^i) \sum_{\mathbf{z}-\mathbf{z}_C} q^t_{i,\mathbf{z}} - \sum_i \log Z_\theta(\mathbf{x}^i) \\
= & \max_\theta \sum_i \sum_C \sum_{\mathbf{z}_C} q^t_{i,\mathbf{z}_C} F_\theta(\mathbf{y}^i_C, \mathbf{z}_C|\mathbf{x}^i) - \sum_i \log Z_\theta(\mathbf{x}^i)
\end{aligned}
$$

Example: CRFs



$\equiv P(z_1, z_2 | \theta^t, y_3, y_4, y_6)$

$i=1$

z_1	z_2	y_3	y_4	z_3	y_6
Hostel	II,	IIT	Bombay,	Powai,	Mumbai
x_1	x_2	x_3	x_4	x_5	x_6

In In City

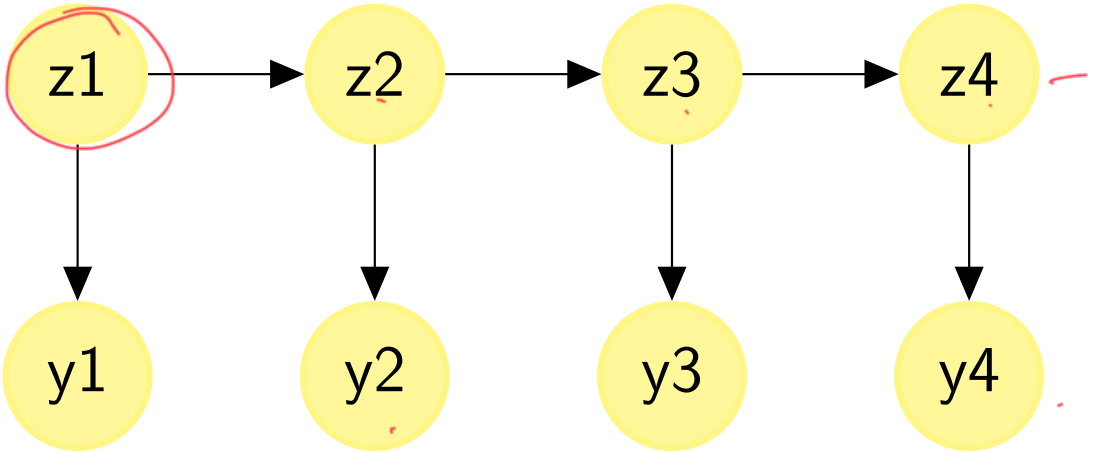
z_1 — z_2 — y_3 — y_4 — z_3 — y_6

(z_1, z_2) (z_2, y_3) . . .

$y \in \{$ House#, Inst Name, Area, City, other $\}$

Example: CRF continued

Example: HMM training



Parameters θ of the network:

- $P(z_j|z_{j-1})$, $z_j \in 1, \ldots, m$ are table potentials(m^2 entries). Since they are independent of j, they can be written as $P(z|z')$.
- $P(y_j|z_j)$, $y_j \in 1, \ldots, n$ are table potentials(nm entries). Since they are independent of j, they can be written as $P(y|z)$.
- $P(z)$ has m entries.

For HMM, $\mathbf{x}$ is not present.

Example: HMM training(continued)

There are two different kinds of $F_\theta(\mathbf{y}^i_C, \mathbf{z}_C)$

- $\log P(\mathbf{y}^i_p | z_p)$, $C \equiv (y_p, z_p)$
- $\log P(z_p | z_{p-1})$, $C \equiv (z_p, z_{p-1})$

$$\max_\theta \sum_i \sum_C \sum_{\mathbf{z}_C} q^t_{i,\mathbf{z}_C} F_\theta(\mathbf{y}^i_C, \mathbf{z}_C)$$
$$= \sum_{i=1}^N \sum_{p=1}^n \sum_{z_p} q^t_{i,z_p} \log P(y_{ip} | z_p) +$$
$$\sum_{p=2}^n \sum_{z_p, z_{p-1}} q^t_{i,z_p,z_{p-1}} \log P(z_p | z_{p-1})$$

Call $P(y|z)$ as θ_{yz} and $P(z|z')$ as $\theta_{zz'}$. In the M-step, we calculate

$$\theta^{t+1}_{yz} = \frac{\sum_{i=1}^N \sum_{p=1}^n q^t_{i,z_p=z} [[\mathbf{y}^i_p = y]]}{\sum_{i=1}^N \sum_{p=1}^n q^t_{i,z_p=z}}$$

$$\theta^{t+1}_{zz'} = \frac{\sum_{i=1}^N \sum_{p=2}^n q^t_{i,z_p=z,z_{p-1}=z'}}{\sum_{i=1}^N \sum_{p=2}^n q^t_{i,z_{p-1}=z'}}$$

Example: HMM training(continued)

In E-Step, we calculate:

$$q_{i,z_p=z} = P(z_p = z|y_i, \theta^t)$$

and

$$q_{i,z_p=z,z_{p-1}=z'} = P(z_p = z, z_{p-1} = z'|y_i, \theta^t)$$

Could be calculated using junction trees.
This algorithm is called **Baum Welch algorithm**.

Overall training algorithm for BNs with hidden variables

EM Algorithm

Input: Graph G, Data D with observed subset of variables $\mathbf{y}$ and hidden variables $\mathbf{z}$.

Initially ($t = 0$): Assign random variables of parameters $\theta^t = \Pr(y_j|\mathbf{y}_{Pa(j)})^t$

for $= 1, \ldots, T$ **do**

E-step

for $i = 1, \ldots, N$ **do**

Use inference in G to estimate conditionals $q^t_{i,\mathbf{z}_C} = \Pr(\mathbf{z}_c|\mathbf{y}^i, \theta^t)$ for all variable subsets $c \subseteq (j, pa(j))$ involving any hidden variable.

end for

M-step

When j is observed: $\Pr(y_j|pa(y_j) = \mathbf{z}_c)^t = \frac{\sum_{i=1}^N q^t_{i,\mathbf{z}_C}[[y^i_j == y_j]]}{\sum_{i=1}^N q^t_{i,\mathbf{z}_C}}$

When j is not observed:

$\Pr(z_j|pa(y_j) = \mathbf{z}_{c-j})^t = \frac{\sum_{i=1}^N q^t_{i,\mathbf{z}_C}}{\sum_{i=1}^N \sum_{z_j=1}^m q^t_{i,[\mathbf{z}_{c-j}, z_j]}}$

end for

Take Aways for Learning with Hidden variables

1. When not variables are observed in each training sample, learning of potentials expressing their interaction becomes complicated.
2. EM algorithm is an efficient option: it alternates between estimating probability distribution over hidden variables and solving for potentials with weighted instances.
3. Variational approximation provides a formal justification for the EM algorithm and is more broadly applicable as we will see next.
4. For graphical models: the distribution over hidden variables can be factorized.

Variational Auto Encoders(VAEs)

VAEs: useful for generating high dimensional objects with latent variables. E.g. generate images with latent factor capturing its properties.

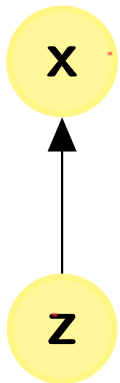
Hidden variables $\mathbf{z}$ are continuous & multidimensional. $\mathbf{z} = z_1 \ldots z_k$.

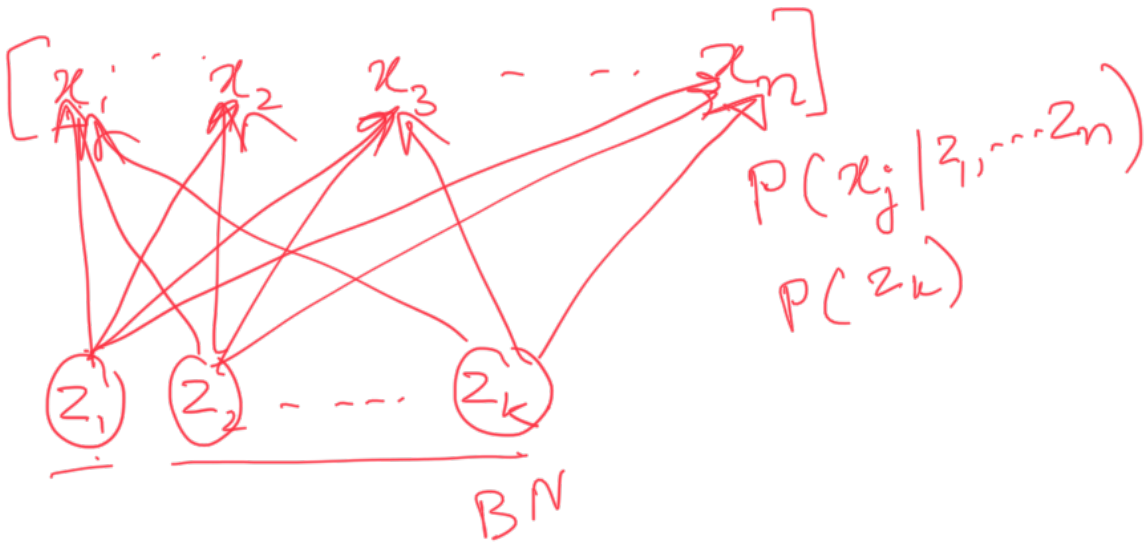
Observed variables are continuous. We denote them as $\mathbf{x} = x_1, \ldots, x_n$ to be consistent with paper on the topic.

All $\mathbf{z}$ are parents of each observed variable which are also continuous.

$P_\theta(\mathbf{x}|\mathbf{z})$

Prior $P_\theta(\mathbf{z})$



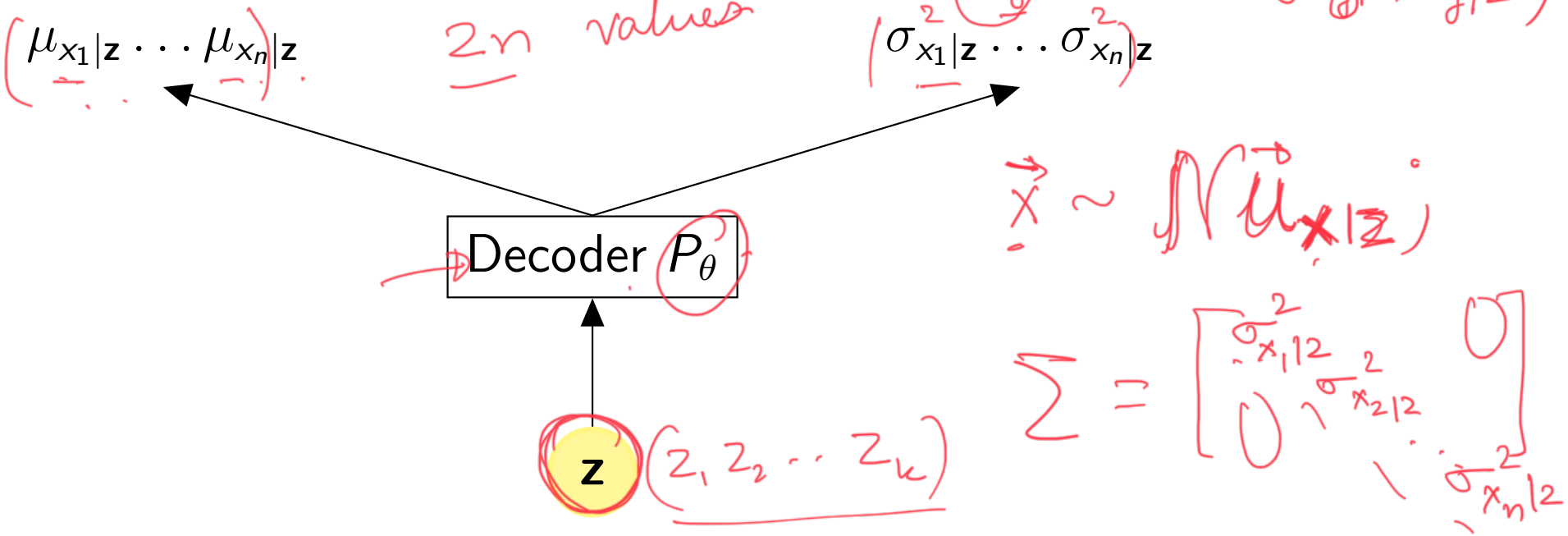


Potentials in VAE

$P(z_j)$ is usually simple like Gaussian. This provides prior distributions over the latent variables.

The $P(\mathbf{x}|\mathbf{z})$ distribution is complicated (it has to generate complex objects, example images, time series, etc). We choose to make it a neural network. The parameters of the neural network are called θ.

Example NN with Gaussian output layer.



Training VAEs

Training data $D = \{\mathbf{x}^i\}_{i=1}^N$ is observed and $\mathbf{z}$ is hidden. Marginalize over $\mathbf{z}$ to compute $P(\mathbf{x})$.

$$\max_\theta \sum_{i=1}^N \log P(\mathbf{x}^i|\theta) = \max_\theta \sum_{i=1}^N \log \int P_\theta(\mathbf{x}|\mathbf{z}) P_\theta(\mathbf{z}) dz$$

$\mathbf{z} \in R^k, \mathbf{x} \in R^d$ We can apply EM algorithm to train with hidden variables.

Training VAEs(Continued)

$$\sum_{i=1}^{N} \log P(\mathbf{x}^i|\theta) = \sum_{i=1}^{N} \log \int_z P_\theta(\mathbf{x}^i|\mathbf{z}) P_\theta(\mathbf{z}) dz$$

$$= \sum_{i=1}^{N} \max_{q_{i,\mathbf{z}}} \int_z (q_{i,\mathbf{z}} \log P_\theta(\mathbf{x}^i|\mathbf{z}) P_\theta(\mathbf{z}) - q_{i,\mathbf{z}} \log q_{i,\mathbf{z}}) dz$$

$$s.t. q_{i,\mathbf{z}} \geq 0, \int_z q_{i,\mathbf{z}} = 1 \quad \forall i$$

But unlike in the usual applications of EM, for calculating optimal $q^t_{i,\mathbf{z}}$, we have to compute

$$q^t_{i,\mathbf{z}} = P_{\theta=\theta^t}(\mathbf{z}|\mathbf{x}^i) = \frac{P_{\theta^t}(\mathbf{x}^i|\mathbf{z}) P_{\theta^t}(\mathbf{z})}{\int_z P_{\theta^t}(\mathbf{x}^i|\mathbf{z}) P_{\theta^t}(\mathbf{z})}$$

intractable ... Using EM as-is is not promising.

But since $\mathbf{z}$ is continuous, calculating $q^t_{i,\mathbf{z}}$ is difficult.

Training VAE(Continued)

To compute $q_{i,\mathbf{z}}$ we can use another neural network $q_\phi(\mathbf{z}|\mathbf{x}^i)$.

$$\sum_{i=1}^{N} \max_{q_{i,\mathbf{z}}} \int_z (q_{i,\mathbf{z}} \log P_\theta(\mathbf{x}^i|\mathbf{z}) P_\theta(\mathbf{z}) - q_{i,\mathbf{z}} \log q_{i,\mathbf{z}}) dz$$

$$= \max_{\phi} \sum_{i=1}^{N} \int_z (q_\phi(\mathbf{z}|\mathbf{x}^i) \log P_\theta(\mathbf{x}^i|\mathbf{z}) - \int_z q_\phi(\mathbf{z}|\mathbf{x}^i) \log \frac{q_\phi(\mathbf{z}|\mathbf{x}^i)}{P_\theta(z)}) dz$$

Estimating the first term

We use sampling to estimate the integral

$$\max_{\theta,\phi} \sum_{i=1}^{N} \int_{z} (q_\phi(\mathbf{z}|\mathbf{x}^i) \log P_\theta(\mathbf{x}^i|\mathbf{z}) dz = \max_{\theta,\phi} \sum_{i=1}^{N} E_{q_\phi(\mathbf{z}|\mathbf{x}^i)}[\log P_\theta(\mathbf{x}^i|\mathbf{z})]$$

approximating expectation with sampling.

$$\approx \max_{\theta,\phi} \sum_{i=1}^{N} \frac{1}{r} \sum_{j=1}^{r} \log P_\theta(\mathbf{x}^i|\mathbf{z}_j)$$

where $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_r \sim q_\phi(\mathbf{z}|\mathbf{x}^i)$ main constraint.

How to sample from q_ϕ while still allowing the gradients to flow through parameters ϕ?

Approximating expected value with average on a sample.

$P(x)$ $g(x)$ $E_{P(x)}[g(x)] = \int_x P(x) g(x) dx \approx \frac{1}{r} \sum_{j=1}^{r} g(x^j)$

$x^1, x^2, \ldots, x^r \sim P(x)$

Reparameterization trick

Assume $q_\phi(\mathbf{z}|\mathbf{x}^i) = \prod_k q_\phi(z_k|\mathbf{x}^i)$

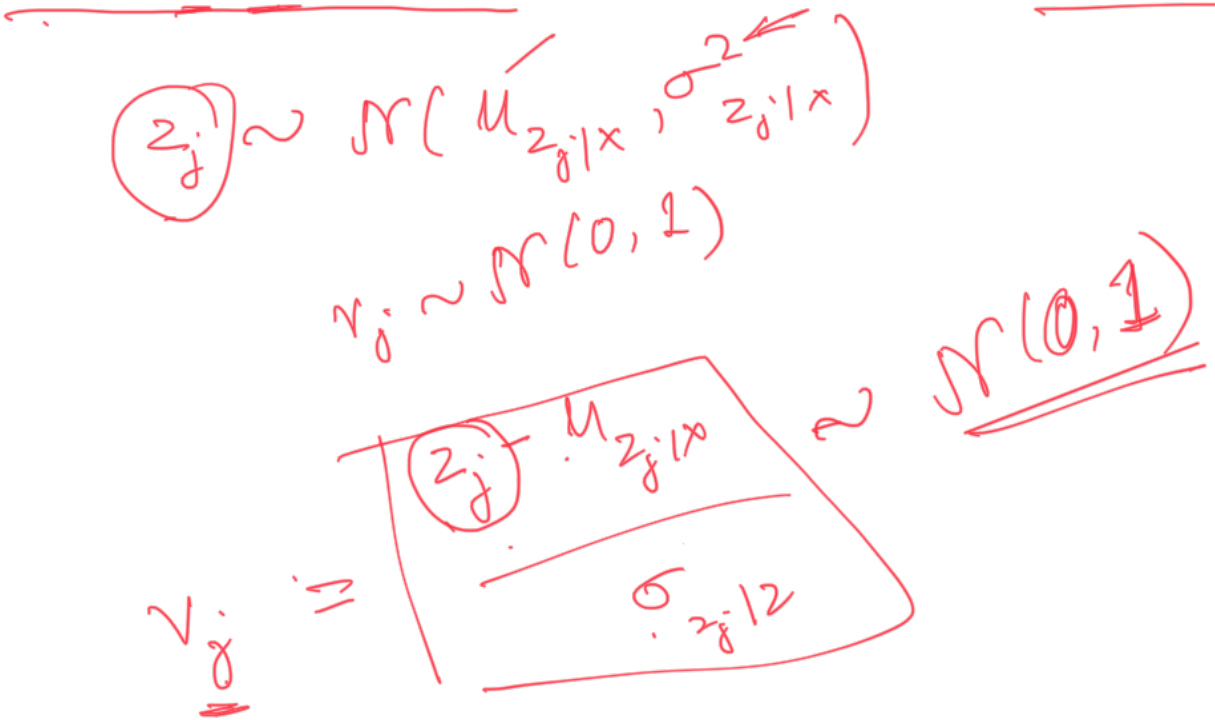
Let $q_\phi(z_k|\mathbf{x}^i) \sim \mathcal{N}(\mu_{z_k|\mathbf{x}^i}, \sigma^2_{z_k|\mathbf{x}^i})$

To sample from $\mathcal{N}(\mu_{\mathbf{z}|\mathbf{x}^i}, \sigma^2_{\mathbf{z}|\mathbf{x}^i})$, we use **Reparameterization trick**.

Sample from a parameterless distribution. Here, we choose $\mathcal{N}(0, I)$.

Let the samples be $v_1 \ldots v_r$

$z_j = v_j \sigma_{z|\mathbf{x}^i} + \mu_{z|\mathbf{x}^i}$ are samples from $\mathcal{N}(\mu_{z|\mathbf{x}^i}, \sigma_{z|\mathbf{x}^i})$.

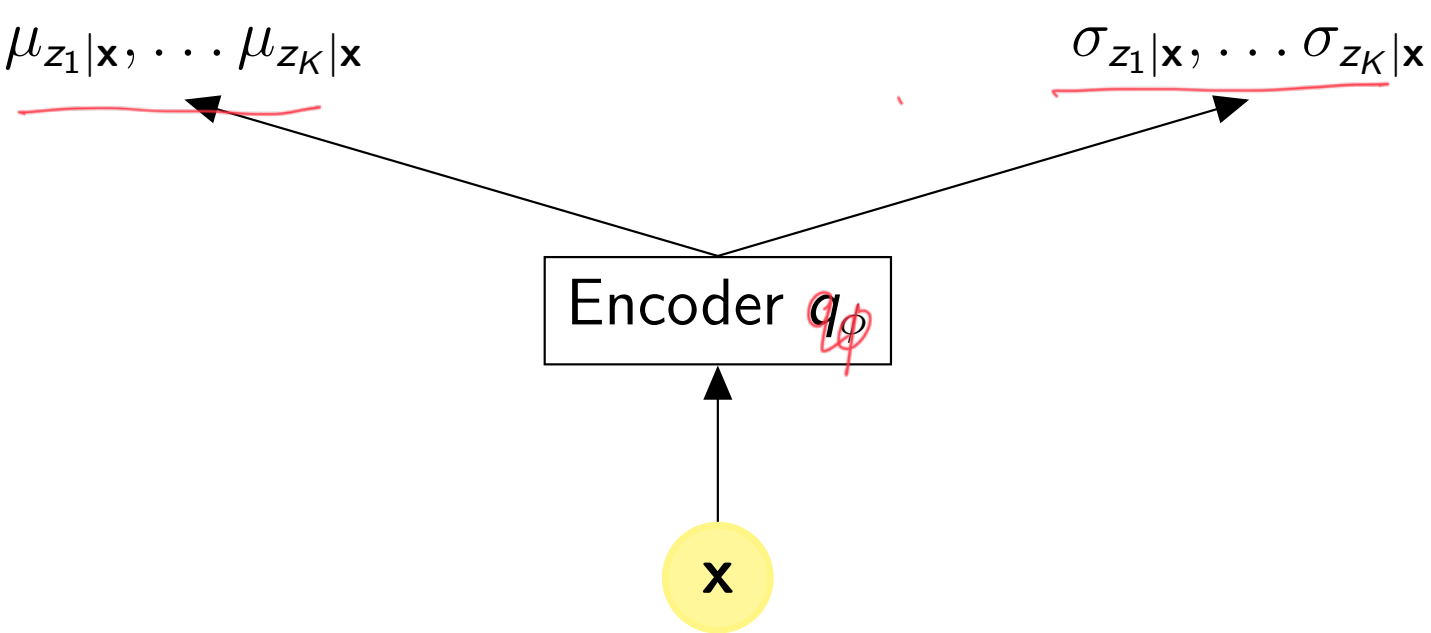


Training VAE(Continued)

$$\max_{\theta,\phi} \sum_{i=1}^{N} \frac{1}{r} \sum_{j=1}^{r} \log P_\theta(\mathbf{x}^i|\mathbf{z}_j) = \max_{\theta,\phi} \sum_{i=1}^{N} \frac{1}{r} \sum_{j=1}^{r} \log P_\theta(\mathbf{x}^i|\mu_{\mathbf{z}|\mathbf{x}^i} + \sigma_{\mathbf{z}|\mathbf{x}^i} * \mathbf{v}_j)$$

depend on ϕ

r: samples
j do not involve ϕ
$\sim N(0, I)$



Calculating second term: KL distance

$\int_{\mathbf{z}} q_\phi(\mathbf{z}|\mathbf{x}^i) \log \frac{q_\phi(\mathbf{z}|\mathbf{x}^i)}{P_\theta(z)}$ is the KL distance between $q_\phi(\mathbf{z}|\mathbf{x}^i)$ and $P_\theta(z)$.

KL distance in closed form

Assume $P_\theta(z) = \mathcal{N}(0,1), q(z|x^i) = \mathcal{N}(\mu_i, \sigma_i^2)$

$$KL(q(z|x), P(z)) = \int_z q(z|x) \log \frac{q(z|x^i)}{P(z)}$$

$$= \int_z \mathcal{N}(\mu_i, \sigma_i^2) 0.5[-\frac{(z-\mu_i)^2}{\sigma_i^2} - \log \sigma_i^2 + z^2]$$

$$= 0.5[-\frac{\sigma_i^2}{\sigma_i^2} - \log \sigma_i^2 + \int_z z^2 \mathcal{N}(\mu_i, \sigma_i^2)]$$

$$= 0.5[-1.0 - \log \sigma_i^2 + \mu_i^2 + \sigma_i^2]$$