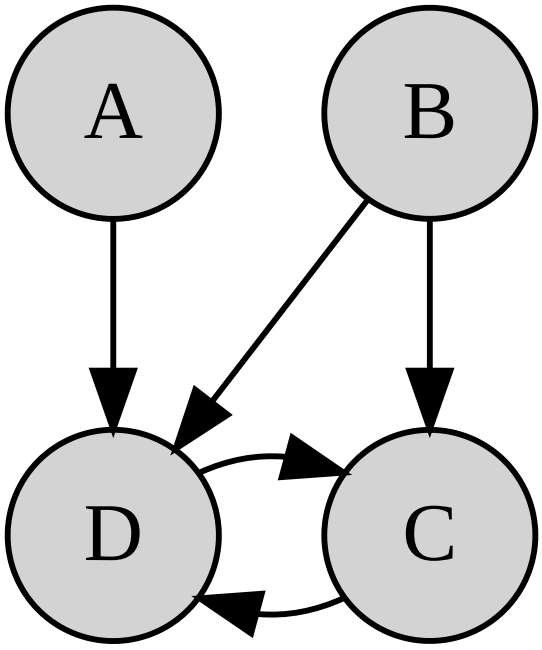
MGM: VAE

Category	
Files	
Created	@May 2, 2023 4:24 PM
Reminder	
Status	Open
URL	
Updated	@May 5, 2023 2:02 PM

Problem-1

(a)

Probabilistic graphical model as PGM is a joint probabilistic distribution that uses a graph structure to encode conditional independence assumptions.

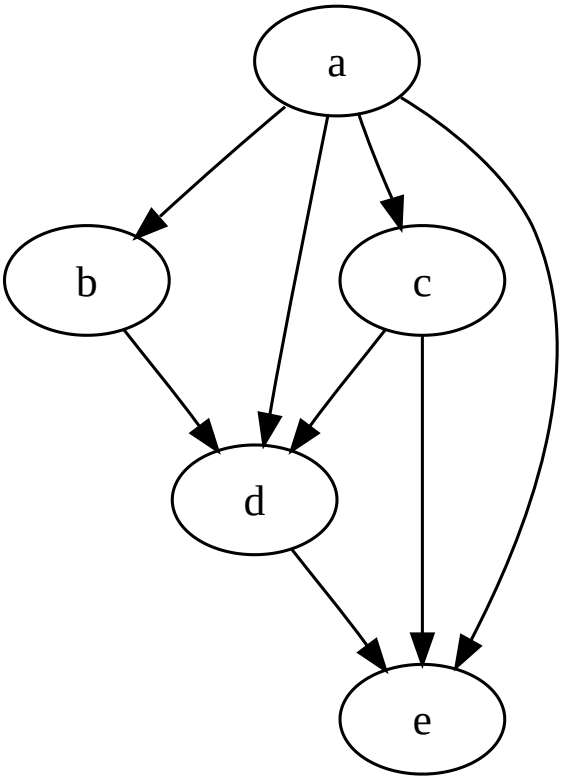


PGMs are composed of two main components: a graph and a set of probability distributions. The graph represents the structure of the relationships between the variables in the model, while the probability distributions define the probabilities of those relationships. PGMs can be divided into two main classes: directed graphical models (also known as Bayesian networks) and undirected graphical models (also known as Markov random fields).

PGMs are used in a variety of applications, such as medical diagnosis, speech recognition, and image segmentation. They allow for efficient inference and learning of complex probabilistic relationships, even in the presence of noisy or incomplete data. By modelling the uncertainty in a system, PGMs provide a powerful tool for decision-making and prediction in a wide range of domains.

(b)

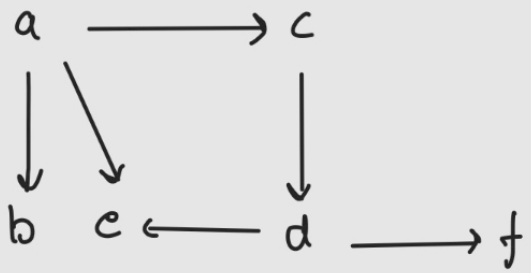
DAG stands for Directed Acyclic Graph, which is a type of graph that has directed edges between its nodes, but does not contain any directed cycles.



(c)

(c)

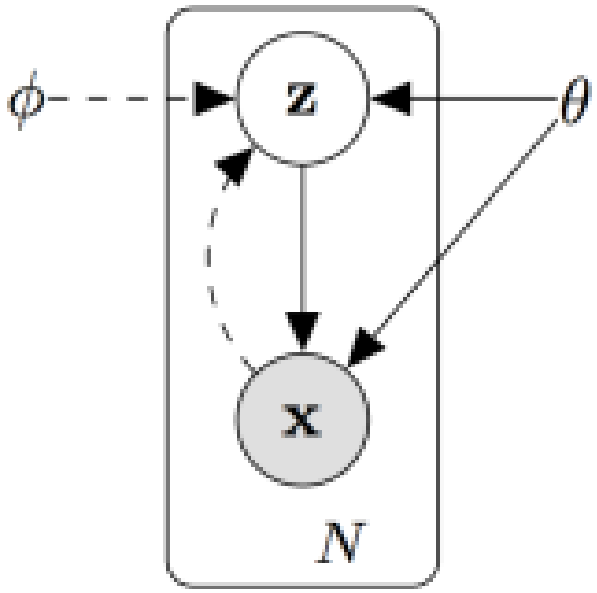
$\{a, b, c, d, e\}$



$$P(a,b,c,d,e) = \prod_{i=1}^{D} P(x_i \mid \text{parent}(x_i))$$

$$= p(a) \cdot p(b|a) \cdot p(c|a) \cdot p(d|c) \cdot p(e|a) \cdot p(e|d) \cdot p(f|d)$$

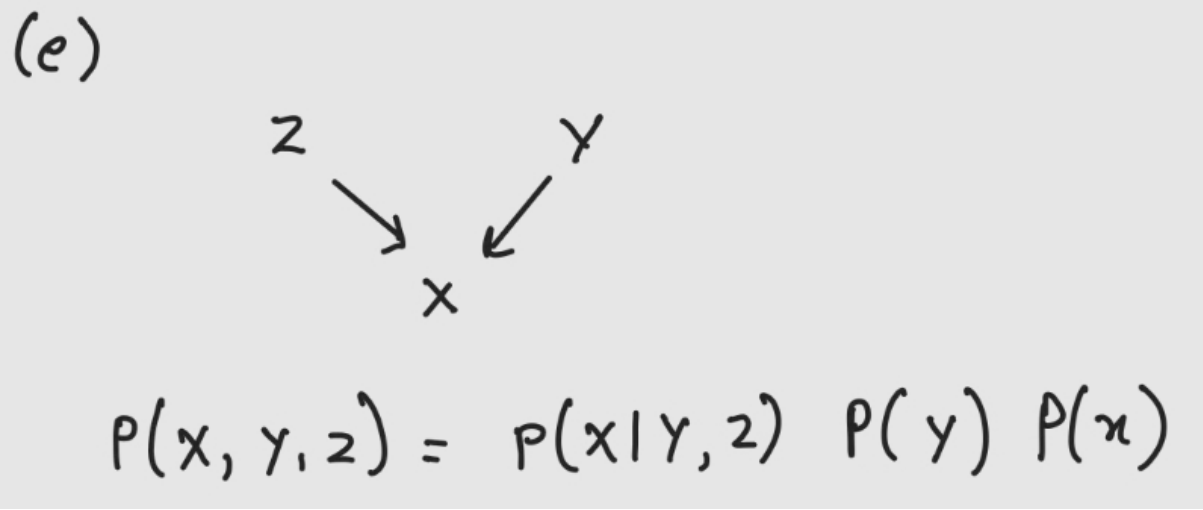
(d)



The model used by VAE :

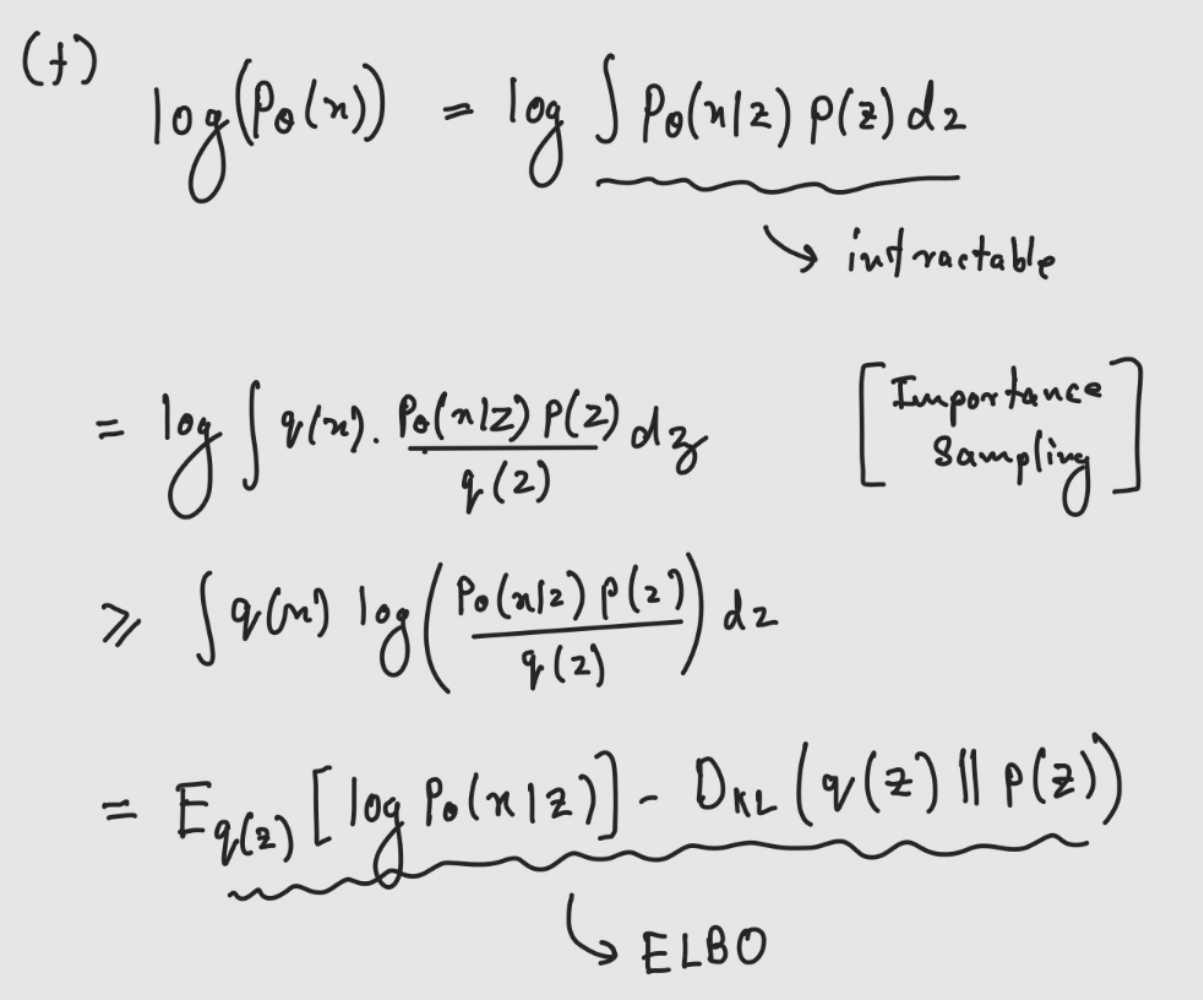
$$p(x, z) = p(x|z) \cdot p(z)$$

(e)



(f)

In VAE, importance sampling is used to estimate the expectations of the posterior distribution of the latent variables. The posterior distribution of the latent variables is intractable to compute, so importance sampling is used to obtain a lower bound that approximates it.



(g)

One common choice for the importance sampling distribution is the proposal distribution, which is typically a simpler and more tractable distribution than the true posterior. The proposal distribution is used to generate samples from the latent space, which are then used to estimate the expectations needed to compute the loss function and perform inference.

One common choice for the proposal distribution in VAE is the multivariate Gaussian distribution, which is easy to sample from and is often a good approximation to the true posterior distribution.

(h)

Generalized Jensen's Inequality

If a function $g(x)$ maps inputs to scalar outputs in R and $f : R \rightarrow R$ is a convex function, then for any distribution $P_X(x)$:

$$E_{P_X(x)}[f(g(x))] \geq f(E_{P_X(x)}[g(x))$$

Proof:

(h)

Proof

Say $y = g(x)$,

then by LOTUS:

$$E_{p_X(x)}[f(g(x))] = E_{p_Y(y)}[f(y)]$$

$$\geq f\left(E_{p_Y(y)}[y]\right) \quad \left[\begin{array}{l}\text{By}\\ \text{Jensen's}\\ \text{inequality}\end{array}\right]$$

$$= f\left(E_{p_X(x)}[g(x)]\right) \quad \left[\begin{array}{l}\text{Substituting}\\ g(x) \text{ back}\\ \text{by LOTUS}\end{array}\right]$$

For a concave function 'f' the inequality is as follows:

$$E_{p_X(x)}[f(g(x))] \leq f\left(E_{p_X(x)}[g(x)]\right)$$

(i)

A latent variable is a variable that is not directly observed but is inferred from other variables that are observed. Latent variables are used to represent underlying

factors or concepts that cannot be directly measured or observed, but are believed to have an effect on the observed variables.

Latent variables can be used in various models, such as factor analysis, latent class analysis, and latent variable models, to explain the relationships between observable variables and to capture the unobservable factors that influence them.

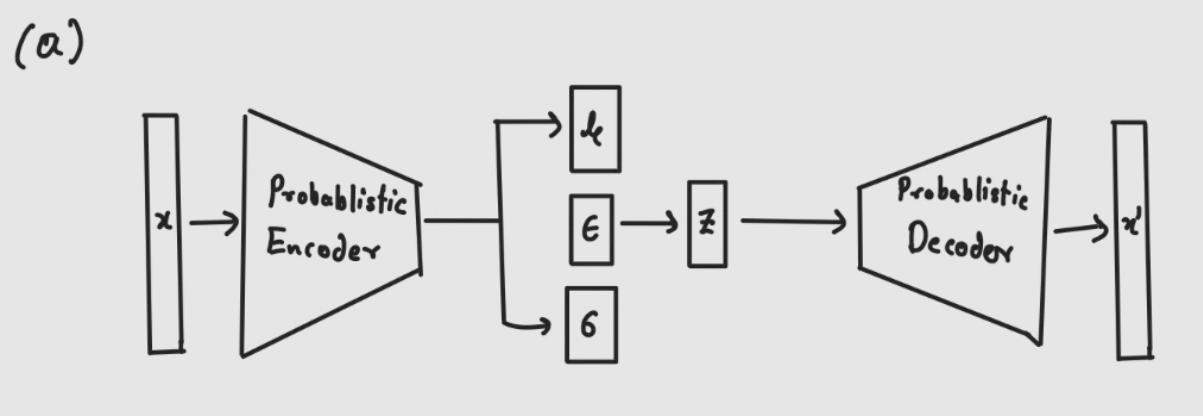
(j)

The use of a latent variable in VAE has several benefits:

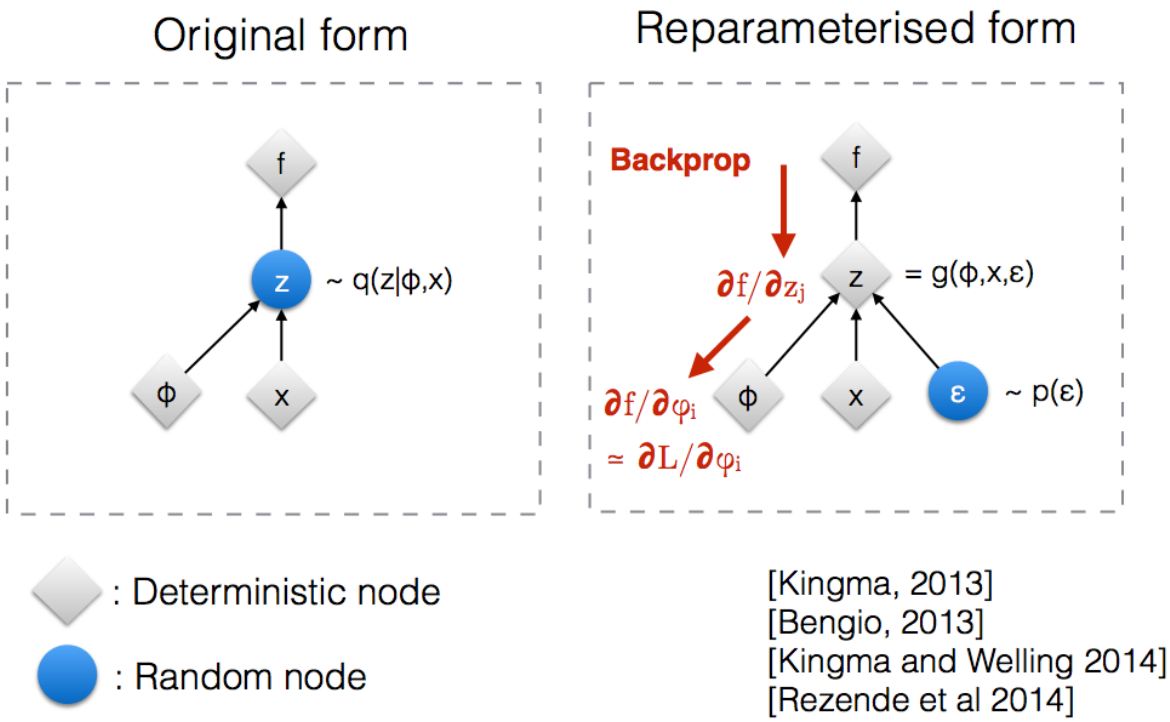
1. Dimensionality reduction: By representing the data in a low-dimensional latent space, VAE can reduce the dimensionality of the data, making it easier to process and analyse.
2. Continuous representations: VAE can generate a continuous representation of the data in the latent space, allowing for smoother interpolation between different data points.
3. Generative modelling: The decoder network in VAE can be used to generate new data points by sampling from the latent space, which is represented by a distribution.
4. Regularization: By introducing a prior distribution over the latent space, VAE can regularize the latent variable, preventing overfitting and improving generalization.

Problem-2

(a)



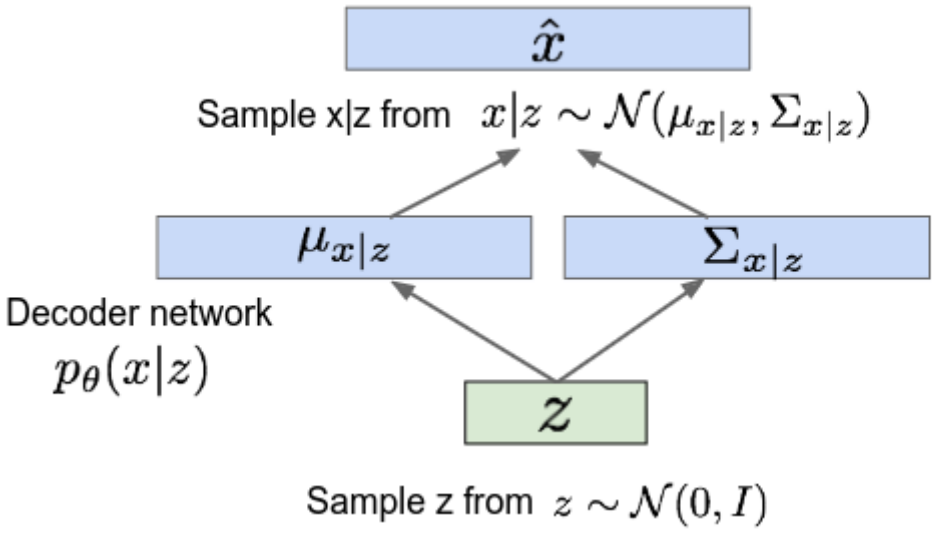
(b)



The reparametrization trick is a technique used in Variational Autoencoder (VAE) models to enable backpropagation through the stochastic layer, which is typically represented by a random sample from a normal distribution.

The reparametrization trick separates the random sampling operation from the model parameters, allowing gradients to flow through the stochastic layer and enabling efficient training via stochastic gradient descent.

(c)



Once a Variational Autoencoder (VAE) is trained, it can be used to generate new images by sampling from the learned latent space. The generation process typically involves the following steps:

1. Sampling a point from the latent space
2. Decoding the latent point back to the original input space, generating a new image.
3. Repeat the above steps to generate multiple new images by sampling different points from the learned latent space.

(d)

(d)

Data likelihood: $p_\theta(x) = \int \underbrace{p_\theta(z)\, p_\theta(x|z)}_{\text{intractable}}\, dz$

$p_\theta(x)$: we want to maximize this

$\Rightarrow$ Using variational inference to approximate the unknown posterior distribution from only the observed data.

$$\log(p_\theta(x)) = \mathop{E}_{z\sim q_\phi(z|x)}\left[\log p_\theta(x)\right] \qquad \left(p_\theta(x) \text{ doesn't depend on } z\right)$$

$$= \mathop{E}_{z}\left[\log\left(\frac{p_\theta(x|z)\, p_\theta(z)}{p_\theta(z|x)}\right)\right] \qquad (\text{Bayes})$$

$$= \mathop{E}_{z}\left[\log\left(\frac{p_\theta(x|z)\, p_\theta(z)}{p_\theta(z|x)} \cdot \frac{q_\phi(z|x)}{q_\phi(z|x)}\right)\right] \qquad (\text{importance sampling})$$

$$= \mathop{E}_{z}\left[\log(p_\theta(x|z))\right] - \mathop{E}_{z}\left[\log\left(\frac{q_\phi(z|x)}{p_\theta(z)}\right)\right] + \mathop{E}_{z}\left[\log\left(\frac{q_\phi(z|x)}{p_\theta(z|x)}\right)\right]$$

$$= \mathop{E}_{z}\left[\log p_\theta(x|z)\right] - D_{KL}\left[q_\phi(z|x) \,\|\, p_\theta(z)\right] + \underbrace{D_{KL}\left[q_\phi(z|x) \,\|\, p_\theta(z|x)\right]}_{\geq 0} \quad (\text{intractable})$$

$$\geq \underbrace{\mathop{E}_{z}\left[\log p_\theta(x|z)\right] - D_{KL}\left[q_\phi(z|x) \,\|\, p_\theta(z)\right]}_{L(x,\theta,\phi)}$$

$L(x,\theta,\phi)$: Tractable lower Bound (ELBO)

(e)

Monte Carlo estimation is used in two places in a Variational Autoencoder (VAE):

1. Computation of ELBO: To compute the ELBO, Monte Carlo integration is used to estimate the expected log-likelihood under the approximate posterior distribution over the latent variables. Specifically, a set of samples is drawn from the posterior distribution, and the log-likelihood of the data is estimated by averaging the log-likelihood over the samples.
2. Generating of new samples: Once the VAE is trained, it can be used to generate new samples by sampling from the learned latent space. To do this, Monte Carlo sampling is used to generate a set of samples from the prior distribution over the latent variables.

(f)

(f)

Generalize to 2 multivariate Normal Distributions

$p(x) = N(x; \mu_1, \Sigma_1)$ $\quad \mu_1, \mu_2 \rightarrow$ Means — given

$q(x) = N(x; \mu_2, \Sigma_2)$ $\quad \Sigma_1, \Sigma_2 \rightarrow$ Covariance Matrices

$$D_{KL}(p(x) \| q(x)) = \frac{1}{2}\left[\log \frac{|\Sigma_2|}{|\Sigma_1|} - d + tr(\Sigma_2^{-1}\Sigma_1) + (\mu_2 - \mu_1)^T \Sigma_2^{-1} (\mu_2 - \mu_1)\right]$$

Proof

$$KL(p \| q) = \sum_x p \log\left(\frac{p}{q}\right)$$

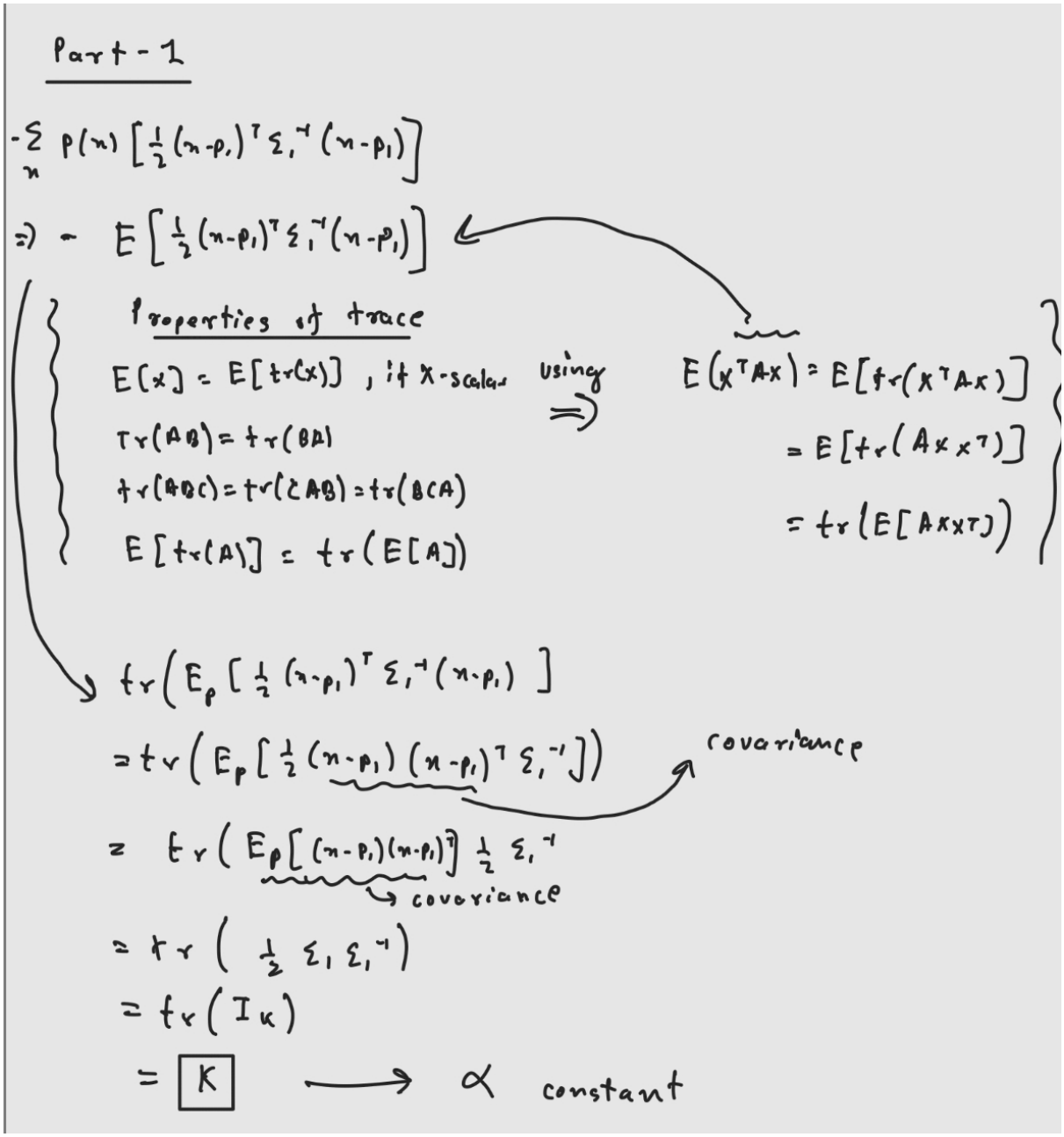
$$= \sum_x p (\log p - \log q)$$

Multivariate Normal Density

$$N(x; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^K |\Sigma|}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right)$$

$x \rightarrow \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_K \end{bmatrix}$

$$KL(p \| q) = \sum_x p(x) \left\{ \frac{1}{2} \log\left(\frac{|\Sigma_2|}{|\Sigma_1|}\right) - \frac{1}{2}(x-\mu_1)^T \Sigma_1^{-1} (x-\mu_1) + \frac{1}{2}(x-\mu_2)^T \Sigma_2^{-1} (x-\mu_2) \right\}$$


Part - 1
$-\sum_x p(x)\left[\frac{1}{2}(x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1)\right]$
$\Rightarrow -E\left[\frac{1}{2}(x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1)\right]$
Properties of trace
$E[x] = E[tr(x)]$, if x-scalar
$tr(AB) = tr(BA)$
$tr(ABC) = tr(CAB) = tr(BCA)$
$E[tr(A)] = tr(E[A])$
using
$\Rightarrow$
$E(x^TAx) = E[tr(x^TAx)]$
$= E[tr(Axx^T)]$
$= tr(E[Axx^T])$
$tr\left(E_p\left[\frac{1}{2}(x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1)\right]\right)$
$= tr\left(E_p\left[\frac{1}{2}(x-\mu_1)(x-\mu_1)^T\Sigma_1^{-1}\right]\right)$
covariance
$= tr\left(E_p[(x-\mu_1)(x-\mu_1)^T]\frac{1}{2}\Sigma_1^{-1}\right)$
covariance
$= tr\left(\frac{1}{2}\Sigma_1\Sigma_1^{-1}\right)$
$= tr(I_K)$
$= K$
$\propto$ constant

Part - 2

$$\sum_x p(x)\left[\frac{1}{2}(x-\mu_2)^T \Sigma_2^{-1}(x-\mu_2)\right]$$

$$\rightarrow \sum_x p(x)\left[\frac{1}{2}\left((x-\mu_1)+(\mu_1-\mu_2)\right)^T \Sigma_2^{-1}\left((x-\mu_1)+(\mu_1-\mu_2)\right)\right]$$

$$(A+B)^T \Sigma_2^{-1}(A+B)$$

$$\Rightarrow (A^T+B^T)\Sigma_2^{-1}(A+B)$$

$$\Rightarrow A^T\Sigma_2^{-1}A + B^T\Sigma_2^{-1}B + \underbrace{\left(B^T\Sigma_2^{-1}A + A^T\Sigma_2^{-1}B\right)}_{\text{same}}$$

similar to prev.

$$\Rightarrow E\left[\frac{1}{2}(x-\mu_1)^T\Sigma_2^{-1}(x-\mu_1)\right] + E_p\left[\underbrace{(x-\mu_1)^T\Sigma_2^{-1}(\mu_1-\mu_2)}_{\rightarrow 0 \text{ (proof?)}}\right]$$

$$+ E\left[\underbrace{(\mu_1-\mu_2)^T\Sigma_2^{-1}(\mu_1-\mu_2)}_{\text{constant}}\right]$$

$$\Rightarrow tr\left\{\frac{\Sigma_2^{-1}\Sigma_1}{2}\right\} + \underbrace{(\mu_1-\mu_2)^T\Sigma_2^{-1}(\mu_1-\mu_2)}_{E[\text{const.}] = \text{const.}} + 0$$

$$\rightarrow \beta_{\text{const.}}$$

$$E_p\left[(x-\mu_1)^T\Sigma_2^{-1}(\mu_2-\mu_1)\right]$$
$$= \left[(E_p[x]-\mu_1)^T\Sigma_2^{-1}(\mu_2-\mu_1)\right]$$
$$= \left[(\mu_1-\mu_1)\cdots\right] = \boxed{0}$$

(g)

Autoencoder:

- Non-probabilistic model
- No assumption about the data distribution
- Encoder maps input to a fixed-length latent representation
- No explicit way to sample from the learned latent space

Variational Autoencoder (VAE):

- Probabilistic model
- Assumes a prior distribution over the latent space
- Encoder maps input to a distribution over the latent representation
- Allows for explicit sampling from the learned latent space