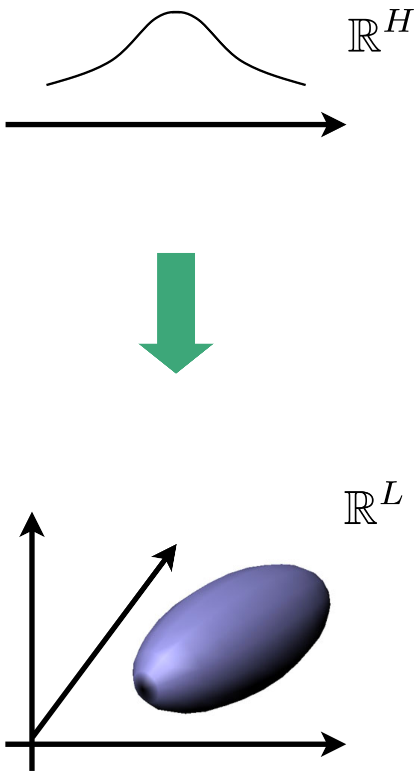
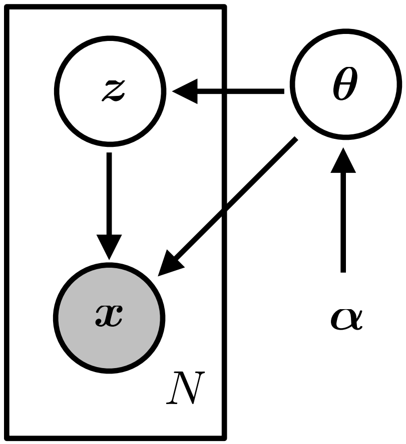
Variational autoencoder (VAE)

Probabilistic PCA as generative model



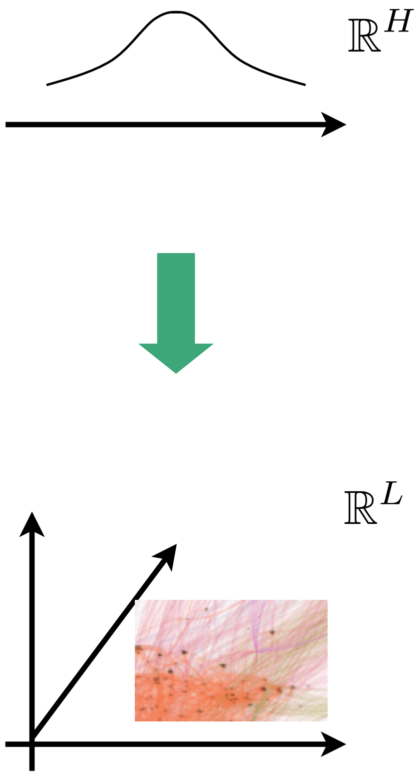
$$\boldsymbol{z} \sim \mathrm{Gauss}_H(\boldsymbol{0}, \boldsymbol{I}),$$

$$\boldsymbol{x} \sim \mathrm{Gauss}_L(\underbrace{\boldsymbol{B}\boldsymbol{z}}, \sigma^2 \boldsymbol{I}),$$



Linear projection just gives Gaussian distribution in high dimensional space

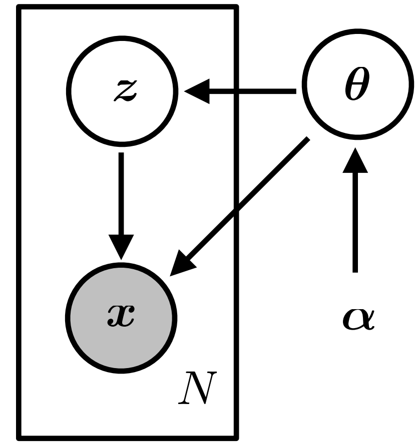
Deep non-linear probabilistic PCA



$$\boldsymbol{z} \sim \mathrm{Gauss}_H(\mathbf{0}, \boldsymbol{I}),$$

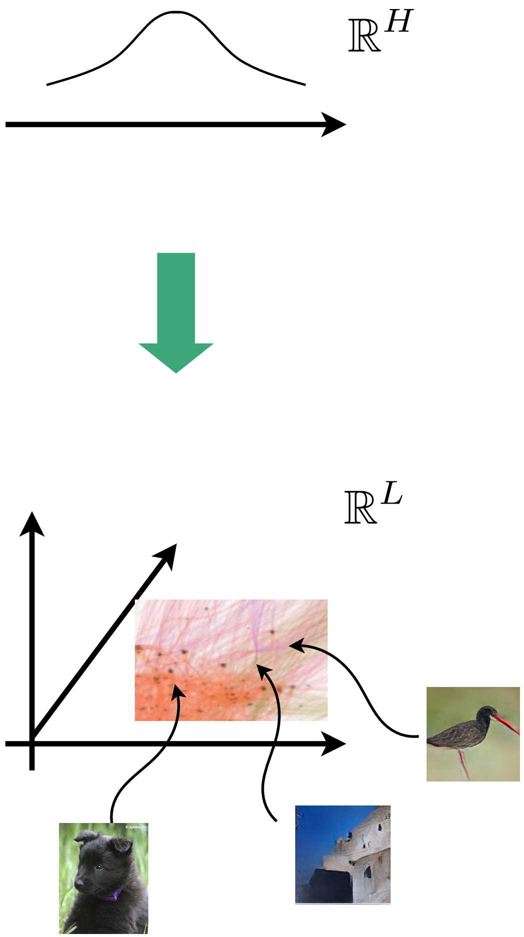
$$\boldsymbol{x} \sim \mathrm{Gauss}_L \underbrace{\left(\boldsymbol{\mu}_\theta(\boldsymbol{z}), \mathbf{Diag}(\boldsymbol{\sigma}^2_\theta(\boldsymbol{z}))\right)}$$

where $(\boldsymbol{\mu}_\theta, \boldsymbol{\sigma}^2_\theta) = \mathrm{DNN}_\theta(\boldsymbol{z}),$

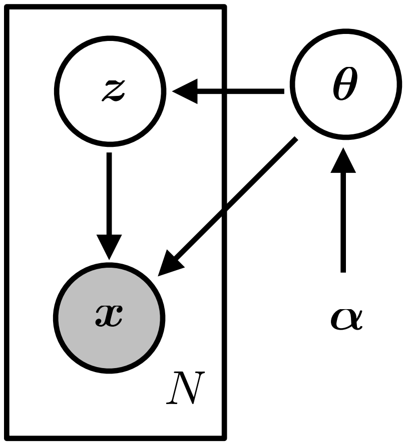


Deep neural network could transfer Gaussian into complicated distribution!

Deep non-linear probabilistic PCA



$$\boldsymbol{z} \sim \mathrm{Gauss}_H(\mathbf{0}, \boldsymbol{I}),$$

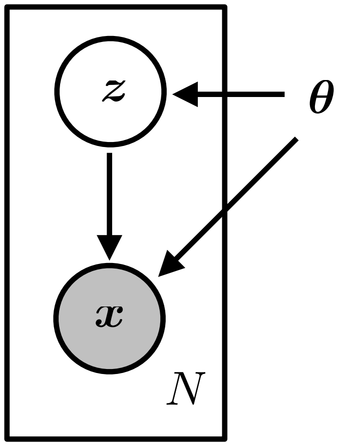


$$\boldsymbol{x} \sim \mathrm{Gauss}_L\left(\underbrace{\boldsymbol{\mu}_\theta(\boldsymbol{z}), \mathbf{Diag}(\boldsymbol{\sigma}_\theta^2(\boldsymbol{z}))}\right)$$

where $\quad (\boldsymbol{\mu}_\theta, \boldsymbol{\sigma}_\theta^2) = \mathrm{DNN}_\theta(\boldsymbol{z}),$

Realistic data generation from Gaussian variables?

Variational autoencoder [Kingma&Welling:2014,Rezende+:2014]



Model parameter: $\boldsymbol{\theta}$

Latent variable: $\boldsymbol{z} \sim p_\theta(\boldsymbol{z}) \quad = \mathrm{Gauss}_H(\boldsymbol{z}; \mathbf{0}, \boldsymbol{I}),$

Observed variable: $\boldsymbol{x} \sim p_\theta(\boldsymbol{x}|\boldsymbol{z}) = \mathrm{Gauss}_L\left(\boldsymbol{x}; \boldsymbol{\mu}_\theta(\boldsymbol{z}), \mathbf{Diag}(\boldsymbol{\sigma}_\theta^2(\boldsymbol{z}))\right)$

where $(\boldsymbol{\mu}_\theta, \boldsymbol{\sigma}_\theta^2) = \mathrm{DNN}_\theta(\boldsymbol{z}),$

– Model parameter is deterministic (EM inference!)

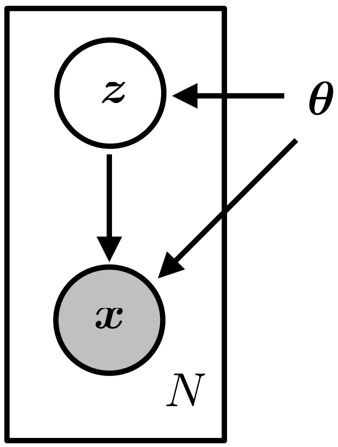
Variational autoencoder [Kingma&Welling:2014,Rezende+:2014]

Model parameter: $\boldsymbol{\theta}$

Latent variable: $\boldsymbol{z} \sim p_\theta(\boldsymbol{z}) = \mathrm{Gauss}_H(\boldsymbol{z}; \mathbf{0}, \boldsymbol{I}),$

Observed variable: $\boldsymbol{x} \sim p_\theta(\boldsymbol{x}|\boldsymbol{z}) = \mathrm{Gauss}_L\left(\boldsymbol{x}; \boldsymbol{\mu}_\theta(\boldsymbol{z}), \mathbf{Diag}(\boldsymbol{\sigma}_\theta^2(\boldsymbol{z}))\right)$

where $(\boldsymbol{\mu}_\theta, \boldsymbol{\sigma}_\theta^2) = \mathrm{DNN}_\theta(\boldsymbol{z}),$



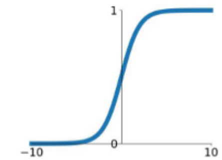
(Conditional) conjugacy?

Conjugacy of (non-linear) neural networks

$$p_\theta(x|z) \propto \exp\left(-\frac{(x - {\color{red}\rho}(\theta_1 z + \theta_2))^2}{2}\right)$$

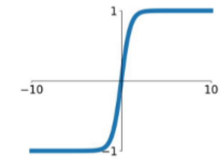
sigmoid:

$\rho(y) = 1/(1 + e^{-y})$



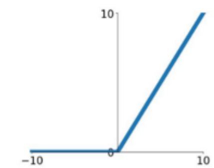
tanh:

$\rho(y) = \tanh(y)$



ReLu:

$\rho(y) = \max(0, y)$



Conjugacy of (non-linear) neural networks

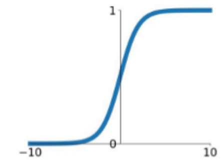
$$p_\theta(x|z) \propto \exp\left(-\frac{(x-\rho(\theta_1 z+\theta_2))^2}{2}\right)$$

No distribution with such function form of *z.*

-> no conjugate prior!

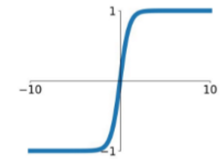
sigmoid:

$\rho(y) = 1/(1+e^{-y})$



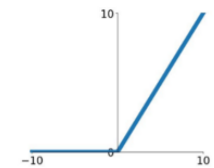
tanh:

$\rho(y) = \tanh(y)$



ReLu:

$\rho(y) = \max(0, y)$

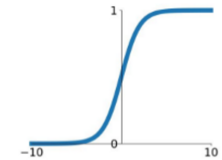


Conjugacy of (non-linear) neural networks

$$p_\theta(x|z) \propto \exp\left(-\frac{(x-\rho(\theta_1 z+\theta_2))^2}{2}\right)$$

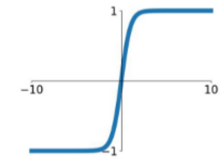
sigmoid:

$\rho(y) = 1/(1+e^{-y})$



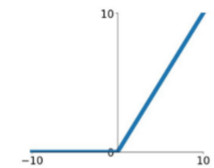
tanh:

$\rho(y) = \tanh(y)$



ReLu:

$\rho(y) = \max(0, y)$



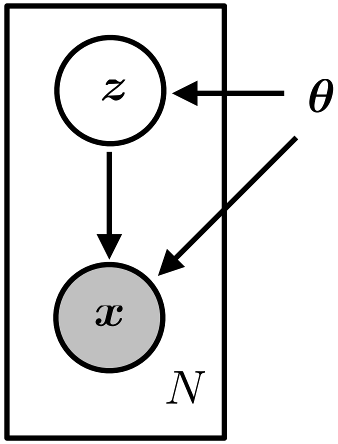
No distribution with such function form of *z.*
-> no conjugate prior!

Furthermore, even if restricting function-form, we can't analytically compute expectations of, e.g.,

$$\langle \log p_\theta(x|z)\rangle_{\mathrm{Gauss}(\boldsymbol{z};\boldsymbol{z},\boldsymbol{I})} = -\left\langle \frac{(x-\rho(\theta_1 z+\theta_2))^2}{2}\right\rangle_{\mathrm{Gauss}(\boldsymbol{z};\boldsymbol{z},\boldsymbol{I})}$$

Some tricks are required!

Variational autoencoder [Kingma&Welling:2014,Rezende+:2014]



Model parameter: $\boldsymbol{\theta}$

Latent variable: $\boldsymbol{z} \sim p_\theta(\boldsymbol{z}) \quad = \mathrm{Gauss}_H(\boldsymbol{z}; \mathbf{0}, \boldsymbol{I}),$

Observed variable: $\boldsymbol{x} \sim p_\theta(\boldsymbol{x}|\boldsymbol{z}) = \mathrm{Gauss}_L\left(\boldsymbol{x}; \boldsymbol{\mu}_\theta(\boldsymbol{z}), \mathbf{Diag}(\boldsymbol{\sigma}_\theta^2(\boldsymbol{z}))\right)$

where $(\boldsymbol{\mu}_\theta, \boldsymbol{\sigma}_\theta^2) = \mathrm{DNN}_\theta(\boldsymbol{z}),$

Approx. posterior: $q_\phi(\boldsymbol{z}|\boldsymbol{x}) = \mathrm{Gauss}_H\left(\boldsymbol{z}; \underbrace{\boldsymbol{\mu}_\phi(\boldsymbol{x}), \mathbf{Diag}(\boldsymbol{\sigma}_\phi^2(\boldsymbol{x}))}_{\text{how shall we design?}}\right),$

Let us restrict the posterior to be Gaussian.

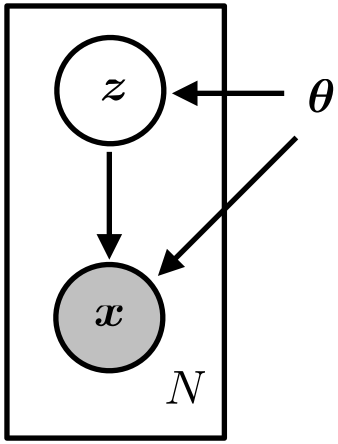
Variational autoencoder [Kingma&Welling:2014,Rezende+:2014]

Model parameter: $\boldsymbol{\theta}$

Latent variable: $\boldsymbol{z} \sim p_\theta(\boldsymbol{z}) \quad = \mathrm{Gauss}_H(\boldsymbol{z}; \mathbf{0}, \boldsymbol{I}),$

Observed variable: $\boldsymbol{x} \sim p_\theta(\boldsymbol{x}|\boldsymbol{z}) = \mathrm{Gauss}_L\left(\boldsymbol{x}; \boldsymbol{\mu}_\theta(\boldsymbol{z}), \mathbf{Diag}(\boldsymbol{\sigma}_\theta^2(\boldsymbol{z}))\right)$

where $(\boldsymbol{\mu}_\theta, \boldsymbol{\sigma}_\theta^2) = \mathrm{DNN}_\theta(\boldsymbol{z}),$



Approx. posterior: $q_\phi(\boldsymbol{z}|\boldsymbol{x}) = \mathrm{Gauss}_H\left(\boldsymbol{z}; \boldsymbol{\mu}_\phi(\boldsymbol{x}), \mathbf{Diag}(\boldsymbol{\sigma}_\phi^2(\boldsymbol{x}))\right),$

where $(\boldsymbol{\mu}_\phi, \boldsymbol{\sigma}_\phi^2) = \mathrm{DNN}_\phi(\boldsymbol{x}),$

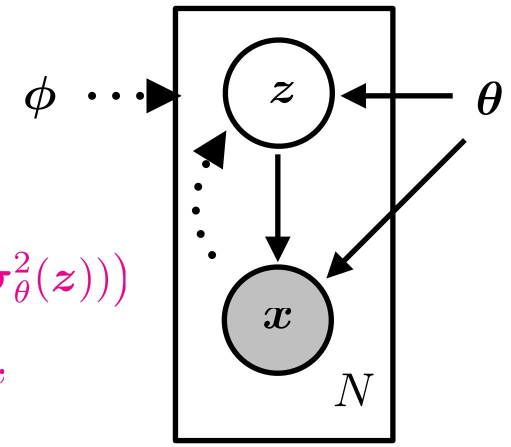
Variational autoencoder [Kingma&Welling:2014,Rezende+:2014]

Model parameter: θ

Latent variable: $\boldsymbol{z} \sim p_\theta(\boldsymbol{z}) \quad = \mathrm{Gauss}_H(\boldsymbol{z}; \boldsymbol{0}, \boldsymbol{I}),$

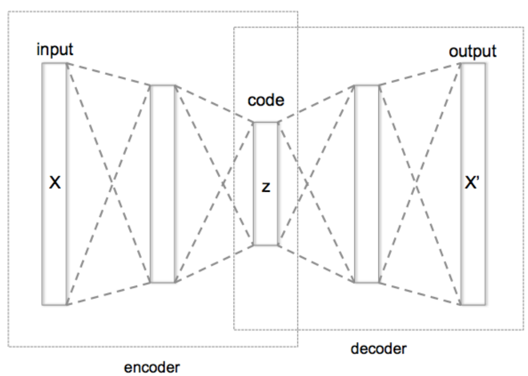
Observed variable: $\boldsymbol{x} \sim p_\theta(\boldsymbol{x}|\boldsymbol{z}) = \mathrm{Gauss}_L\left(\boldsymbol{x}; \boldsymbol{\mu}_\theta(\boldsymbol{z}), \mathbf{Diag}(\boldsymbol{\sigma}_\theta^2(\boldsymbol{z}))\right)$

where $(\boldsymbol{\mu}_\theta, \boldsymbol{\sigma}_\theta^2) = \mathrm{DNN}_\theta(\boldsymbol{z}),$



Approx. posterior: $q_\phi(\boldsymbol{z}|\boldsymbol{x}) = \mathrm{Gauss}_H\left(\boldsymbol{z}; \boldsymbol{\mu}_\phi(\boldsymbol{x}), \mathbf{Diag}(\boldsymbol{\sigma}_\phi^2(\boldsymbol{x}))\right),$

where $(\boldsymbol{\mu}_\phi, \boldsymbol{\sigma}_\phi^2) = \mathrm{DNN}_\phi(\boldsymbol{x}),$



Generative model + inference model form probabilistic autoencoder structure!