From Autoencoders to Variational Autoencoders: The Encoder

Issues with vanilla autoencoders

- The latent space plot isn’t symmetrical around the origin
- Some labels are represented over small areas, others over large ones
- There are discontinuities in the latent space

From Autoencoders to Variational Autoencoders

- Modify encoder component
- Modify loss function

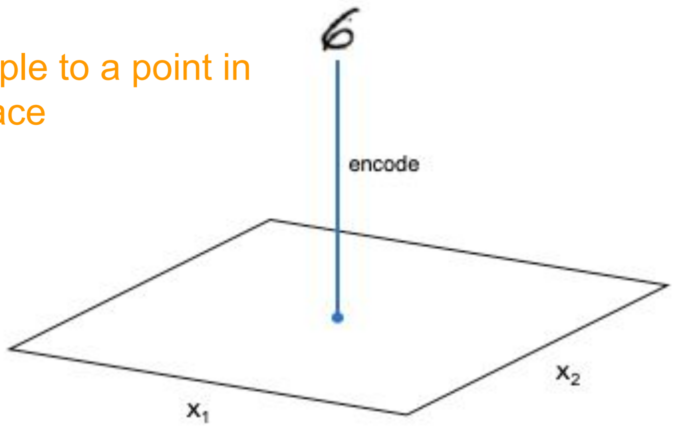
From Autoencoders to Variational Autoencoders

- Modify encoder component
- Modify loss function

Encoder mapping: AE vs VAE

Encoder mapping: AE vs VAE

Map sample to a point in latent space

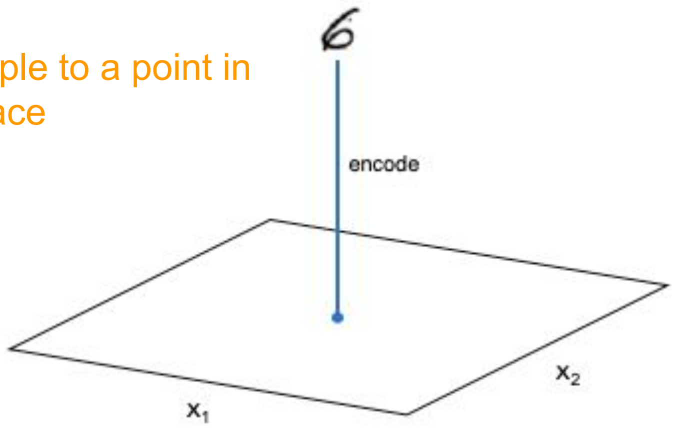


autoencoder

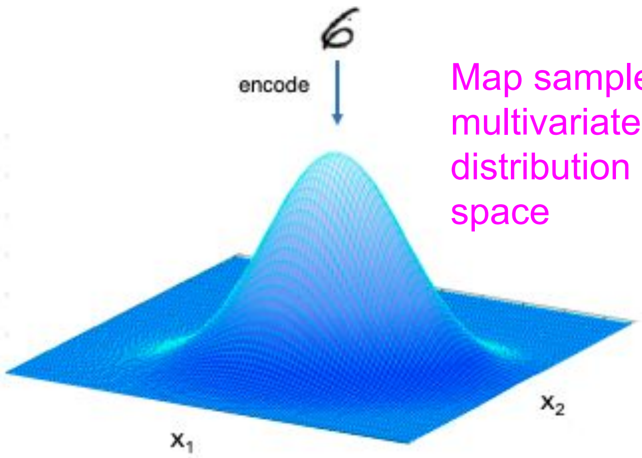
*Images taken from *Generative Deep Learning* by David Foster

Encoder mapping: AE vs VAE

Map sample to a point in latent space



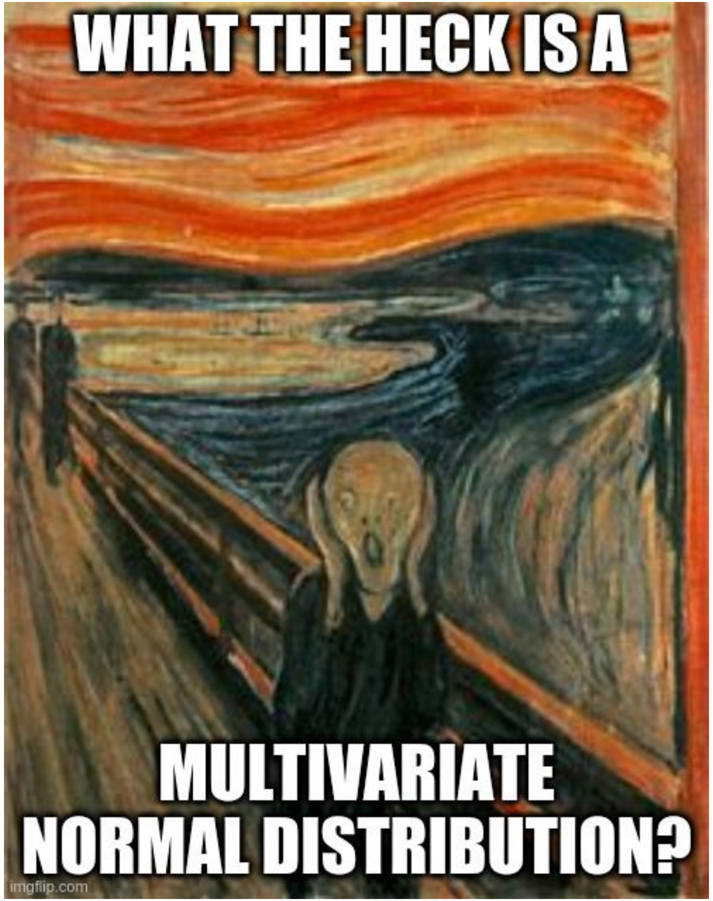
autoencoder



variational autoencoder

Map sample to a multivariate normal distribution in latent space

*Images taken from *Generative Deep Learning* by David Foster


WHAT THE HECK IS A
MULTIVARIATE
NORMAL DISTRIBUTION?
imgflip.com

Defining the normal distribution in 1d

- AKA Gaussian distribution

Defining the normal distribution in 1d

- AKA Gaussian distribution
- Probability distribution with bell curve shape

Defining the normal distribution in 1d

- AKA Gaussian distribution
- Probability distribution with bell curve shape
- Describe real-valued random variables (e.g., population height, weight)

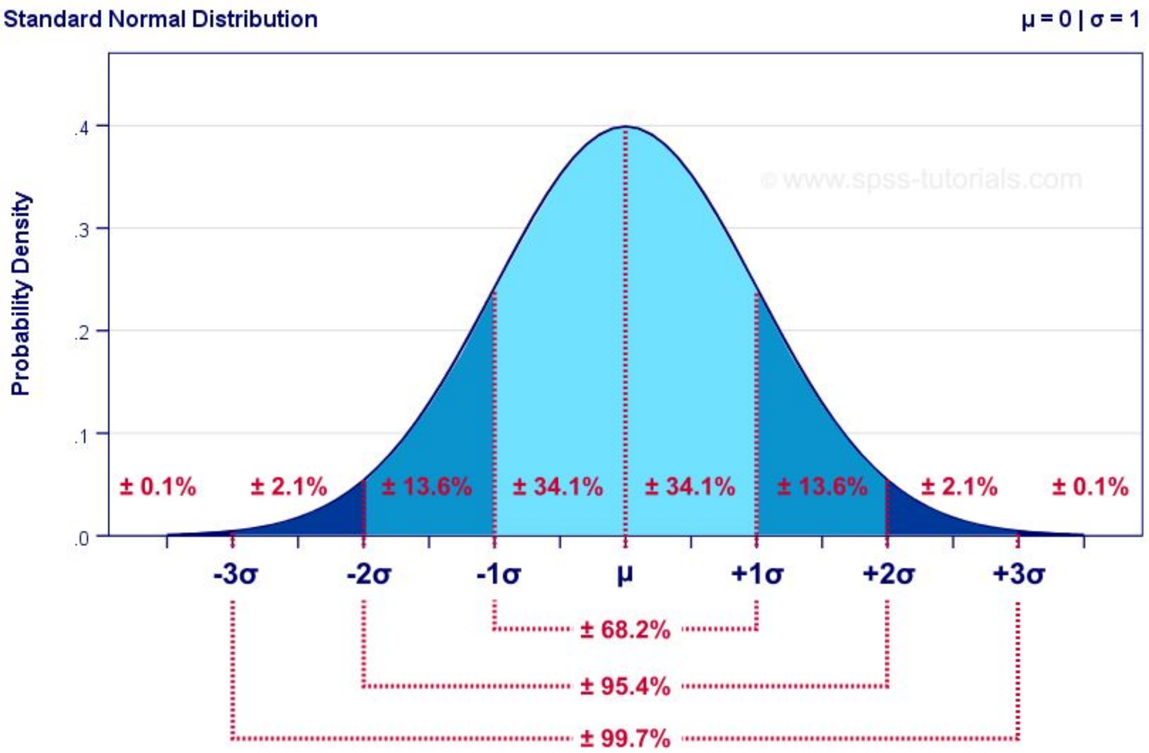
Defining the normal distribution in 1d

- AKA Gaussian distribution
- Probability distribution with bell curve shape
- Describe real-valued random variables (e.g., population height, weight)
- Defined by 2 variables:
    - μ - mean value
    - σ - standard deviation

Defining the normal distribution in 1d

- AKA Gaussian distribution
- Probability distribution with bell curve shape
- Describe real-valued random variables (e.g., population height, weight)
- Defined by 2 variables:
  - μ - mean value -> centre of the distribution
  - σ - standard deviation -> variability of the distribution

Visualising the (standard) normal distribution



Probability density function

$$f(x|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

Defining the normal distribution in 1d

- The greater σ the flatter the curve
- Changing μ shifts the curve right and left
- Let’s play around with the normal distro!

Sampling a point from a normal distribution

$$z = \mu + \sigma \varepsilon$$

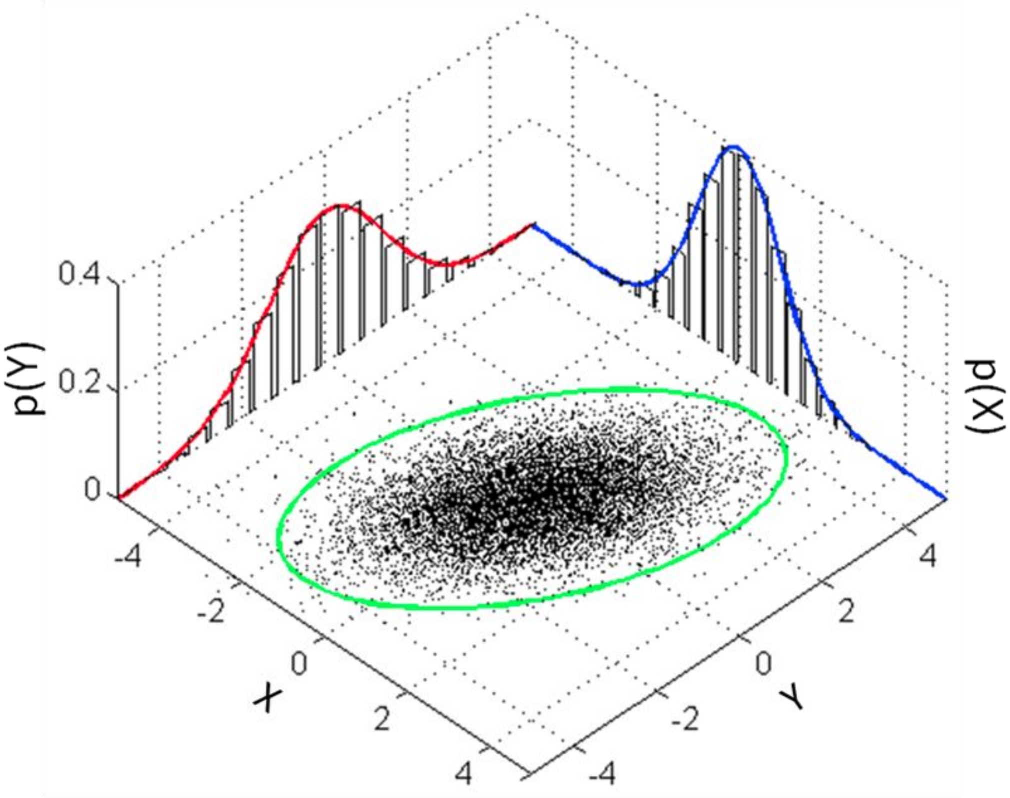
Sampling a point from a normal distribution

$$z = \mu + \sigma\boxed{\varepsilon}$$

Sampled point from
standard normal distribution


LET'S GENERALISE A NORMAL DISTRIBUTION TO MORE THAN 1 DIMENSION
MULTIVARIATE NORMAL DISTRIBUTION
imgflip.com

Visualising the multivariate normal distribution



Formalising the multivariate normal distribution

$$f(x_1, \ldots, x_k) = \frac{e^{-\frac{1}{2}(\vec{x}-\vec{\mu})^T \Sigma^{-1} (\vec{x}-\vec{\mu})}}{\sqrt{(2\pi)^k |\Sigma|}}$$

Formalising the multivariate normal distribution

$$f(x_1, \ldots, x_k) = \frac{e^{-\frac{1}{2}\boxed{(\vec{x}-\vec{\mu})^T \Sigma^{-1} (\vec{x}-\vec{\mu})}}}{\sqrt{(2\pi)^k |\Sigma|}}$$

∝ Mahalanobis distance:

Distance between **x** and **μ**

Formalising the multivariate normal distribution

$$f(x_1, \ldots, x_k) = \frac{e^{-\frac{1}{2}(\vec{x}-\boxed{\vec{\mu}})^T \Sigma^{-1} (\vec{x}-\boxed{\vec{\mu}})}}{\sqrt{(2\pi)^k |\Sigma|}}$$

$$\boxed{\vec{\mu} = \begin{pmatrix} \mu_x \\ \mu_y \end{pmatrix}}$$

Formalising the multivariate normal distribution

$$f(x_1, \dots, x_k) = \frac{e^{-\frac{1}{2}(\vec{x}-\boxed{\vec{\mu}})^T \boxed{\Sigma}^{-1}(\vec{x}-\boxed{\vec{\mu}})}}{\sqrt{(2\pi)^k \boxed{|\Sigma|}}}$$

$$\boxed{\vec{\mu} = \begin{pmatrix} \mu_x \\ \mu_y \end{pmatrix}}$$

$$\boxed{\Sigma = \begin{pmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{pmatrix}}$$

Covariance matrix

Formalising the multivariate normal distribution

$$f(x_1, \ldots, x_k) = \frac{e^{-\frac{1}{2}(\vec{x}-\boxed{\vec{\mu}})^T \boxed{\Sigma}^{-1}(\vec{x}-\boxed{\vec{\mu}})}}{\sqrt{(2\pi)^k \boxed{|\Sigma|}}}$$

$$\boxed{\vec{\mu} = \begin{pmatrix} \mu_x \\ \mu_y \end{pmatrix}}$$

$$\boxed{\Sigma = \begin{pmatrix} \sigma_x^2 & \boxed{\rho}\sigma_x\sigma_y \\ \boxed{\rho}\sigma_x\sigma_y & \sigma_y^2 \end{pmatrix}}$$

Covariance matrix

Correlation between x, y

VAE and multivariate normal distribution

- VAE assumes all dimensions are independent (ρ = 0)

VAE and multivariate normal distribution

- VAE assumes all dimensions are independent (ρ = 0)

$$\Sigma = \begin{pmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{pmatrix} \Rightarrow \begin{pmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{pmatrix}$$

VAE and multivariate normal distribution

- VAE assumes all dimensions are independent (ρ = 0)
- Encoder maps each input to a mean vector and a variance vector in the latent space

VAE and multivariate normal distribution

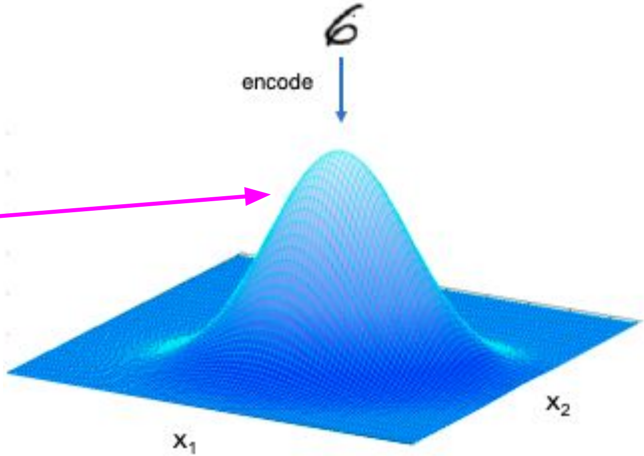
- VAE assumes all dimensions are independent (ρ = 0)
- Encoder maps each input to a mean vector and a variance vector in the latent space

$$\vec{\mu} = \begin{pmatrix} \mu_x \\ \mu_y \end{pmatrix} \quad \vec{\sigma}^2 = \begin{pmatrix} \sigma_x^2 \\ \sigma_y^2 \end{pmatrix}$$

VAE and multivariate normal distribution

- VAE assumes all dimensions are independent (ρ = 0)
- Encoder maps each input to a mean vector and a variance vector in the latent space

$$\vec{\mu} = \begin{pmatrix}\mu_x \\ \mu_y\end{pmatrix} \quad \vec{\sigma}^2 = \begin{pmatrix}\sigma_x^2 \\ \sigma_y^2\end{pmatrix}$$



Encoding data into a point in latent space

$$z = \vec{\mu} + \Sigma\varepsilon$$

Encoding data into a point in latent space

$$z = \vec{\mu} + \Sigma \boxed{\varepsilon}$$

Sampled point from **standard** normal distribution

Encoding data into a point in latent space

$$z = \vec{\mu} + \boxed{\Sigma}\boxed{\varepsilon}$$

Sampled point from **standard** normal distribution

Encoding data into a point in latent space

$$z = \vec{\mu} + \boxed{\Sigma}\boxed{\varepsilon}$$

$$\boxed{\Sigma = e^{\frac{log(\vec{\sigma}^2)}{2}}}$$

Sampled point from **standard** normal distribution

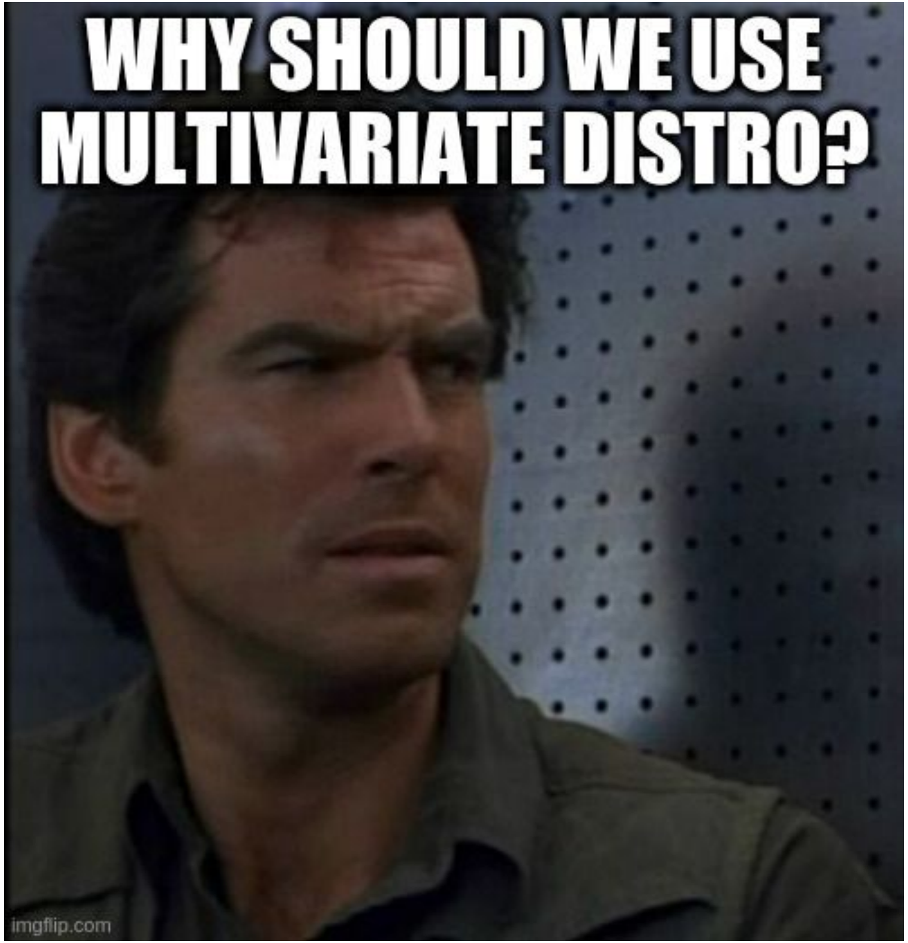
Encoding data into a point in latent space

$$z = \vec{\mu} + \boxed{\Sigma}\boxed{\varepsilon}$$

Sampled point from **standard** normal distribution

$$\boxed{\Sigma = e^{\frac{log(\vec{\sigma}^2)}{2}}}$$

Log can take any value in - / + infinity


WHY SHOULD WE USE
MULTIVARIATE DISTRO?
imgflip.com

A major autoencoder drawback

There are discontinuities
in the latent space

⇩

Some generated images
will be poor

The multivariate normal distro fix

- We’re sampling a random point around the mean value

The multivariate normal distro fix

- We’re sampling a random point around the mean value
- Points in the same area produce very similar images when decoded

The multivariate normal distro fix

- We’re sampling a random point around the mean value
- Points in the same area produce very similar images when decoded
- Reconstruction loss is small

The multivariate normal distro fix

- We’re sampling a random point around the mean value
- Points in the same area produce very similar images when decoded
- Reconstruction loss is small
- VAE ensures a quasi-continuous latent space

The multivariate normal distro fix

- We’re sampling a random point around the mean value
- Points in the same area produce very similar images when decoded
- Reconstruction loss is small
- VAE ensures a quasi-continuous latent space
- We can sample any point in the latent space, expecting the decoder to create a well formed image

What next?

- Discuss techniques to fix ulterior AEs’ issues
- Focus on loss function for VAEs