UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

> Reinventa el mundo <

# Sesión 4.2

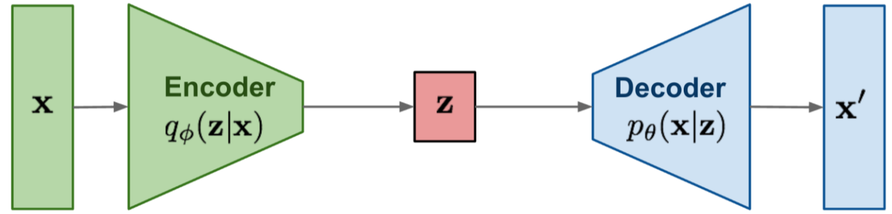
## *Variational autoencoder*

***Variational inference. ELBO.***

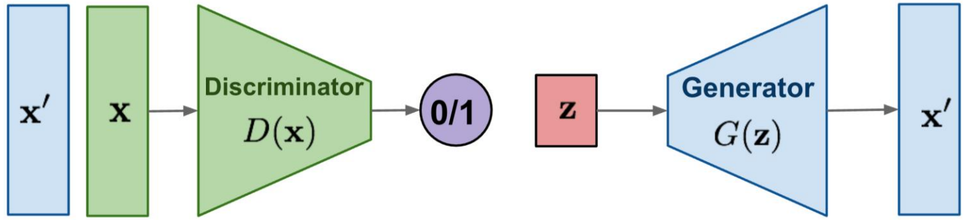
TRANSFORMATEC

# Generative *model*

**VAE:** maximize variational lower bound

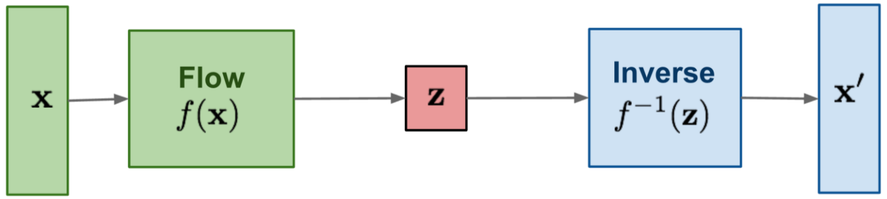



**GAN:** Adversarial training




**Diffusion models:** Gradually add Gaussian noise and then reverse


**Flow-based models:** Invertible transform of distributions

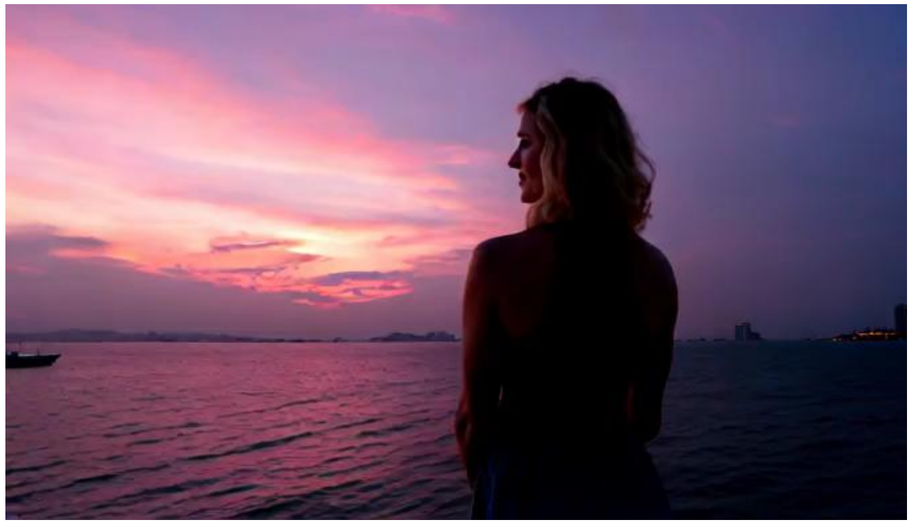


# Generative *model*

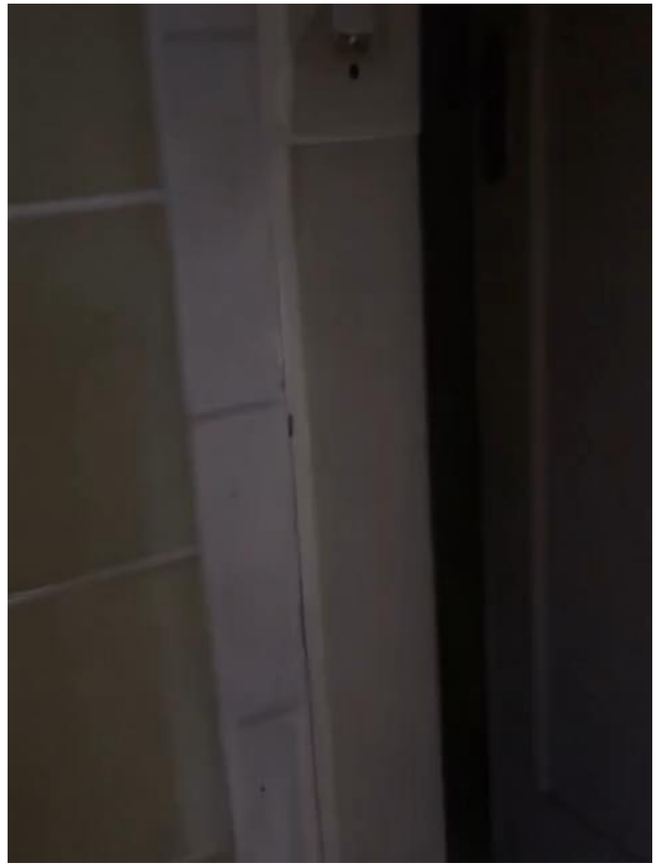
2014

2024

# Generative *model*



CineMaster (2025)



Pika (2025)

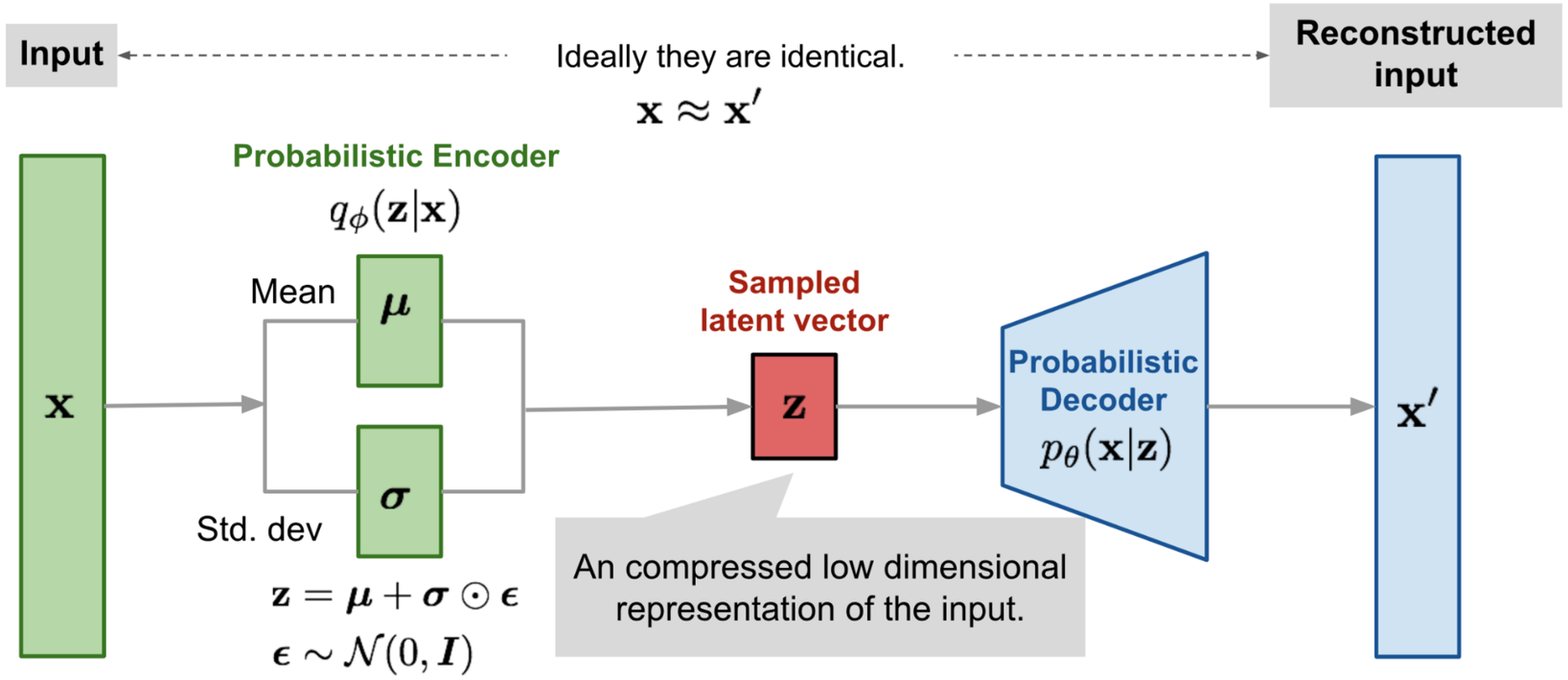
UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA


# 1.

## Variational *Autoencoder*

TRANSFORMATEC

# VAE

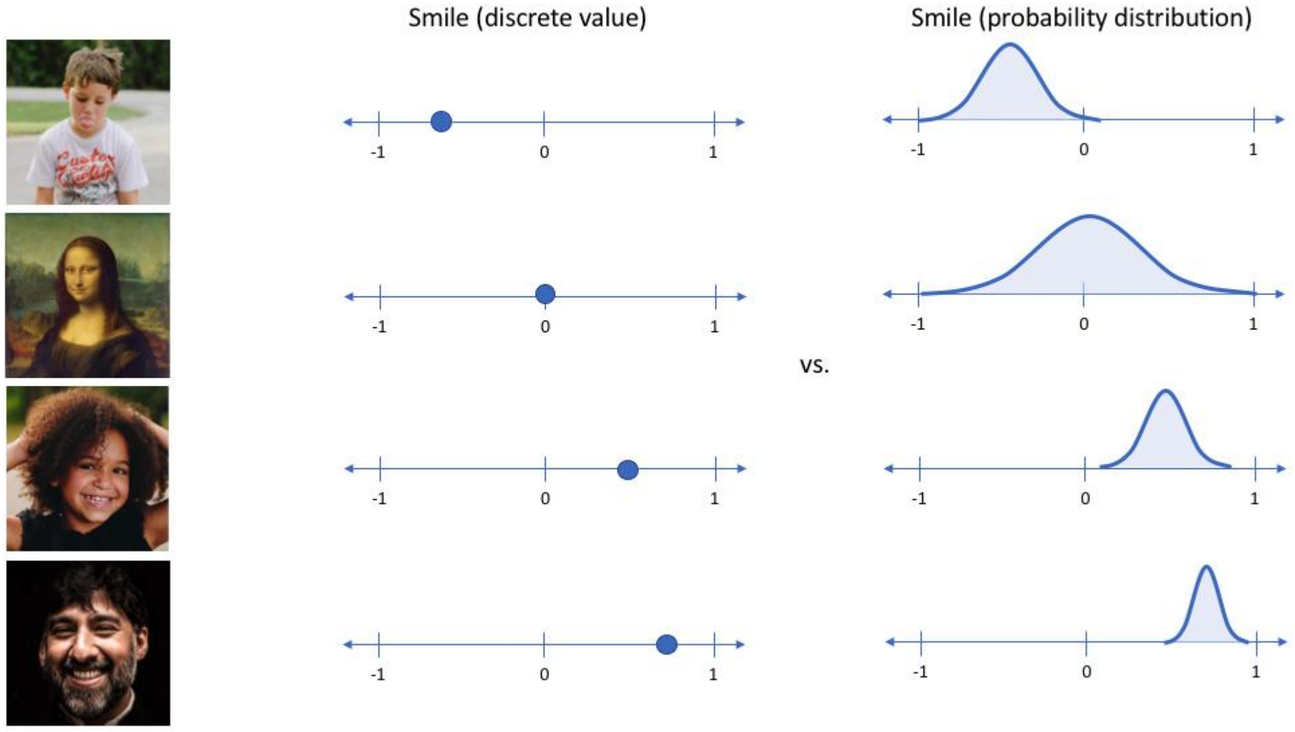





Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

A diferencia de los autoencoders tradicionales que aprenden una función de codificación determinística, los VAEs introducen una distribución probabilística en el espacio latente

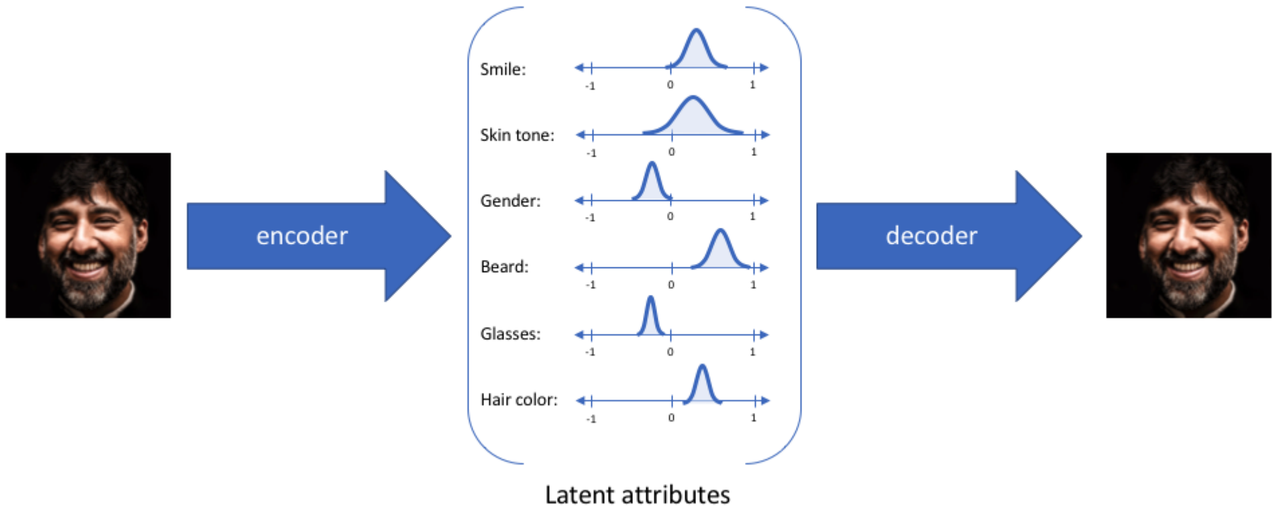





Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

A diferencia de los autoencoders tradicionales que aprenden una función de codificación determinística, los VAEs introducen una distribución probabilística en el espacio latente

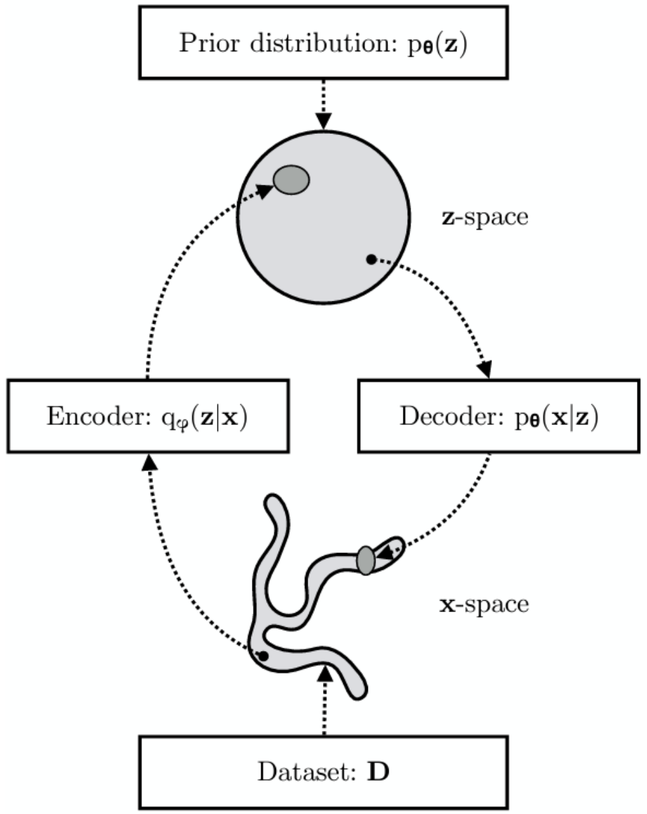





Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

El **encoder**, parametrizado por $\phi$, transforma los datos de entrada en una distribución probabilística sobre el espacio latente.
El **decoder**, parametrizado por $\theta$, utiliza estas representaciones latentes para reconstruir los datos de entrada.

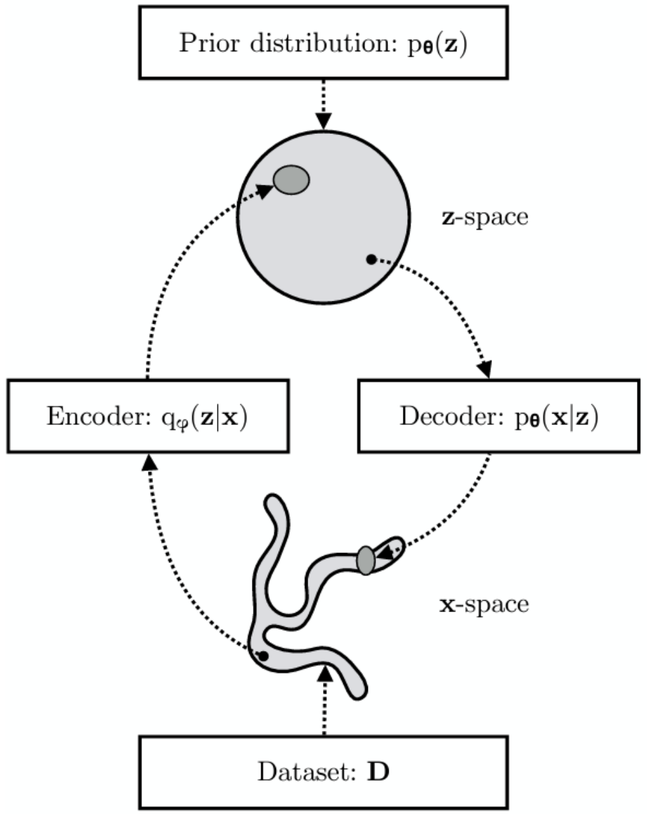




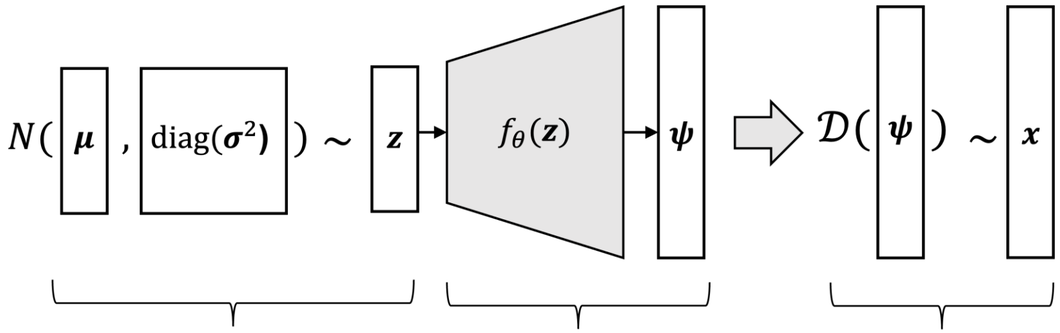

Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

El **encoder**, parametrizado por $\phi$, transforma los datos de entrada en una distribución probabilística sobre el espacio latente.
El **decoder**, parametrizado por $\theta$, utiliza estas representaciones latentes para reconstruir los datos de entrada.




Supongamos que se nos da un conjunto de datos $x_1, \ldots, x_n \in \mathbb{R}^D$ generados por una VAE.




Sample $\boldsymbol{z}$ from a normal distribution parameterized by $\boldsymbol{\mu}$ and a diagonal covariance matrix with the values in $\boldsymbol{\sigma}^2$ along the diagonal

Decode $\boldsymbol{z}$ into the parameters, $\boldsymbol{\psi}$, used to sample $\boldsymbol{x}$

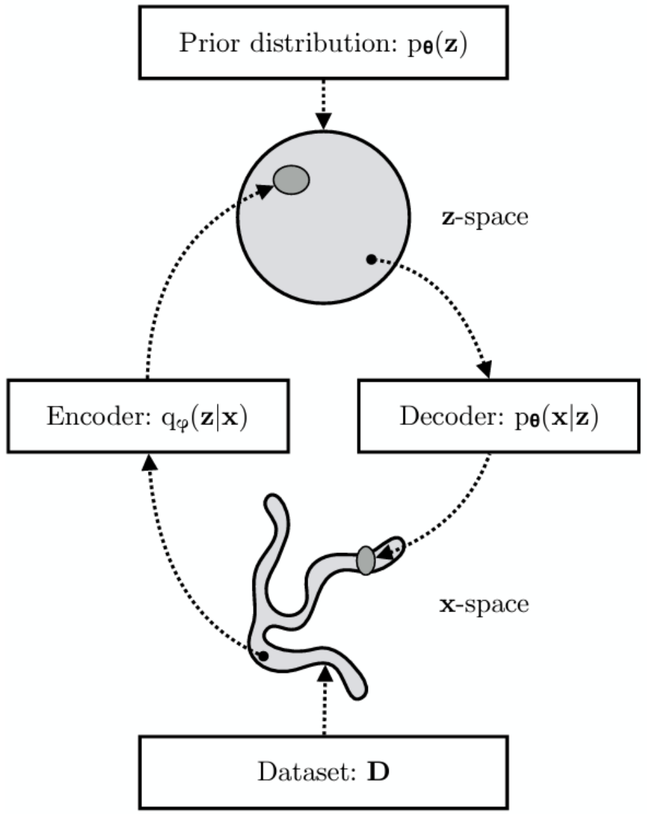
Sample $\boldsymbol{x}$



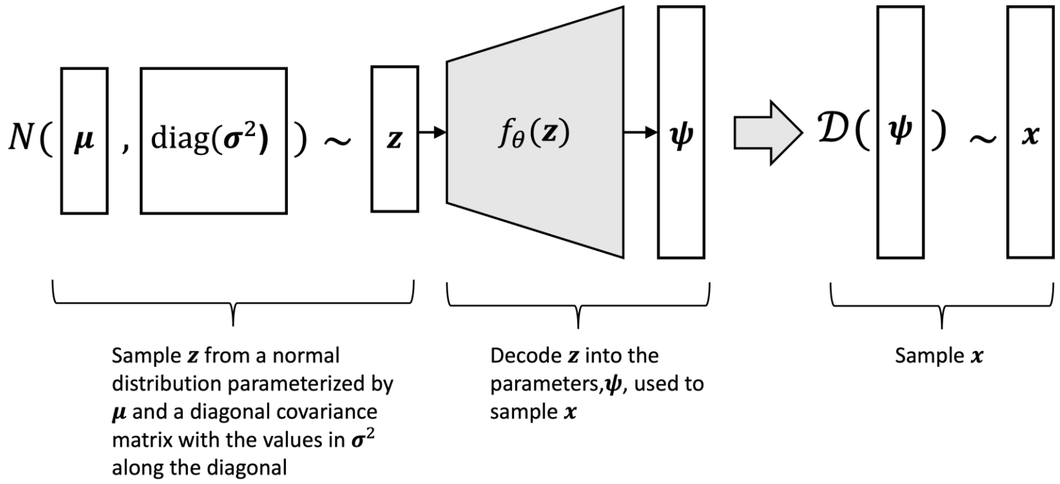
Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

El **encoder**, parametrizado por $\phi$, transforma los datos de entrada en una distribución probabilística sobre el espacio latente.
El **decoder**, parametrizado por $\theta$, utiliza estas representaciones latentes para reconstruir los datos de entrada.




Supongamos que se nos da un conjunto de datos $x_1, \ldots, x_n \in \mathbb{R}^D$ generados por una VAE.




**Posterior distribution** representa cómo los datos observados actualizan la distribución de las variables latentes.

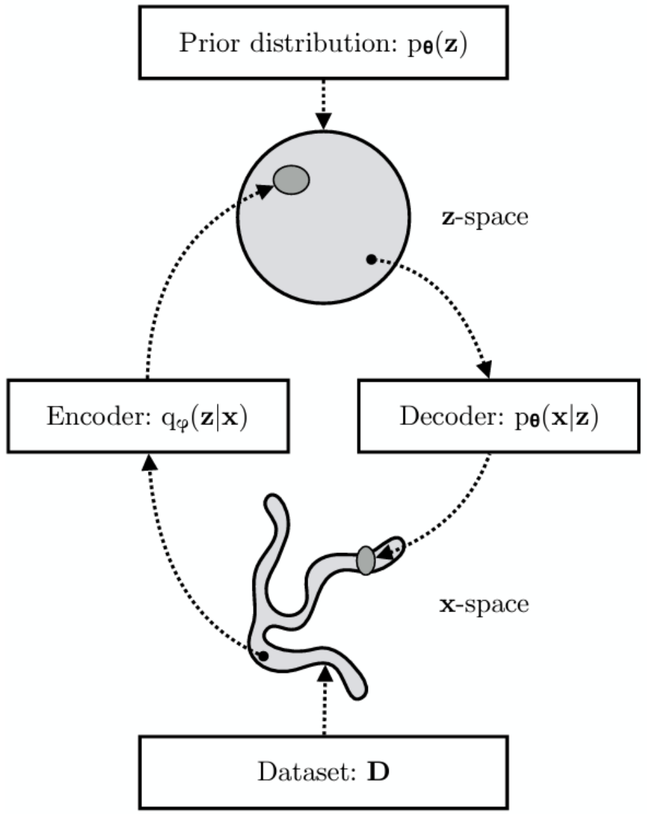
$$p_\theta(z_i \mid x_i) = \frac{p_\theta(x_i \mid z_i)p(z_i)}{\int p_\theta(x_i \mid z_i)p(z_i)\, dz_i}$$

**Bayes Theorem**

Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

El **encoder**, parametrizado por $\phi$, transforma los datos de entrada en una distribución probabilística sobre el espacio latente.
El **decoder**, parametrizado por $\theta$, utiliza estas representaciones latentes para reconstruir los datos de entrada.




Sin embargo, calcular la verdadera posterior distribution es intratable debido a la **integral en el denominador.**

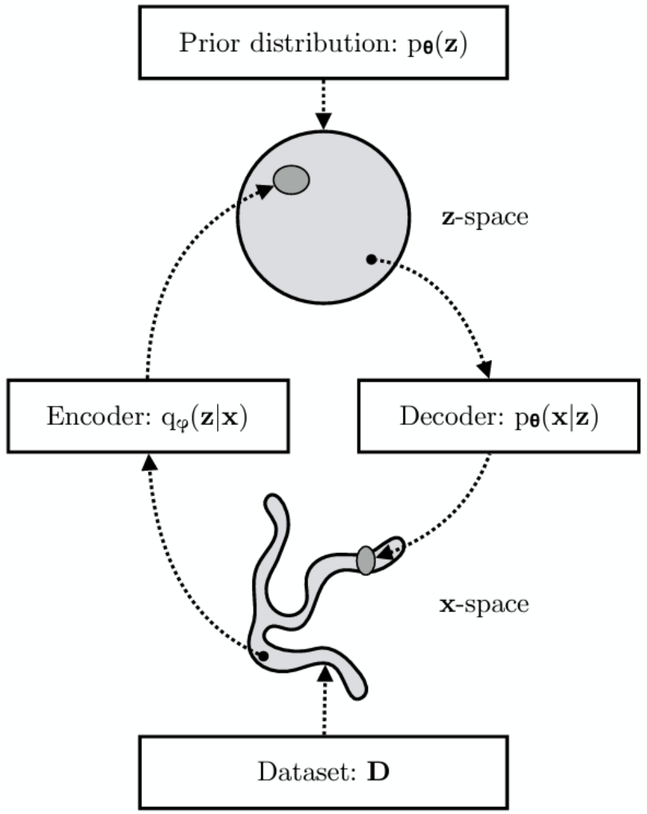
$$p_\theta(z_i \mid x_i) = \frac{p_\theta(x_i \mid z_i)p(z_i)}{\int p_\theta(x_i \mid z_i)p(z_i)\, dz_i}$$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

El **encoder**, parametrizado por $\phi$, transforma los datos de entrada en una distribución probabilística sobre el espacio latente.
El **decoder**, parametrizado por $\theta$, utiliza estas representaciones latentes para reconstruir los datos de entrada.




Sin embargo, calcular la verdadera posterior distribution es intratable debido a la **integral en el denominador.**

$$p_\theta(z_i \mid x_i) = \frac{p_\theta(x_i \mid z_i)p(z_i)}{\int p_\theta(x_i \mid z_i)p(z_i)\, dz_i}$$

Otra solución es **maximum likelihood estimation**:

$$\hat{\theta} := \text{argmax}_\theta \prod_{i=1}^{n} p_\theta(x_i) = \text{argmax}_\theta \prod_{i=1}^{n} \int p_\theta(x_i \mid z_i)p(z_i)\, dz_i$$

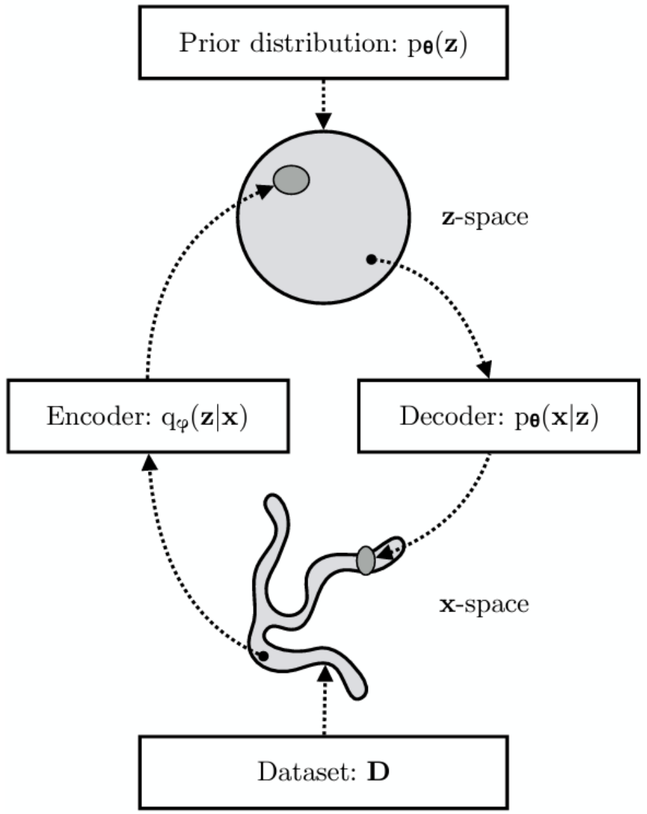
Sin embargo, también necesitaremos resolver dicha integral. :'c



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

El **encoder**, parametrizado por $\phi$, transforma los datos de entrada en una distribución probabilística sobre el espacio latente.
El **decoder**, parametrizado por $\theta$, utiliza estas representaciones latentes para reconstruir los datos de entrada.




Sin embargo, calcular la verdadera posterior distribution es intratable debido a la **integral en el denominador.**

$$p_\theta(z_i \mid x_i) = \frac{p_\theta(x_i \mid z_i)p(z_i)}{\int p_\theta(x_i \mid z_i)p(z_i)\, dz_i}$$

Otra solución es **maximum likelihood estimation**:

$$\hat{\theta} := \text{argmax}_\theta \prod_{i=1}^{n} p_\theta(x_i) = \text{argmax}_\theta \prod_{i=1}^{n} \int p_\theta(x_i \mid z_i)p(z_i)\, dz_i$$

Sin embargo, también necesitaremos resolver dicha integral. :'c

**Aquí entra Variational Inference para solucionar el problema :D**



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# ELBO

Definimos la **evidencia** como $\log p(x;\theta)$

Es la probabilidad marginal de los datos observados bajo el modelo actual.
Cuantifica cuán bien el modelo con parámetros $\theta$ explica los datos observados, es decir es la probabilidad de que los datos hayan sido generados por el modelo tal como está parametrizado

Nosotros podemos probar que la evidencia está acotada por el **ELBO** (Evidence Lower BOund):

$$\log p(x;\theta) \geq E_{Z\sim q}\left[\log \frac{p(x,Z;\theta)}{q(Z)}\right] \qquad ELBO := E_{Z\sim q}\left[\log \frac{p(x,Z;\theta)}{q(Z)}\right]$$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# ELBO

Definimos la **evidencia** como $\log p(x;\theta)$

Es la probabilidad marginal de los datos observados bajo el modelo actual.
Cuantifica cuán bien el modelo con parámetros $\theta$ explica los datos observados, es decir es la probabilidad de que los datos hayan sido generados por el modelo tal como está parametrizado

Nosotros podemos probar que la evidencia está acotada por el **ELBO** (Evidence Lower BOund):

$$\log p(x;\theta) \geq E_{Z\sim q}\left[\log \frac{p(x,Z;\theta)}{q(Z)}\right] \qquad ELBO := E_{Z\sim q}\left[\log \frac{p(x,Z;\theta)}{q(Z)}\right]$$

La demostración es directa:

$$\begin{aligned}\log p(x;\theta) &= \log \int p(x,z;\theta)\, dz \\ &= \log \int p(x,z;\theta)\frac{q(z)}{q(z)}\, dz \\ &= \log E_{Z\sim q}\left[\frac{p(x,Z)}{q(z)}\right] \geq E_{Z\sim q}\left[\log \frac{p(x,Z)}{q(z)}\right]\end{aligned}$$

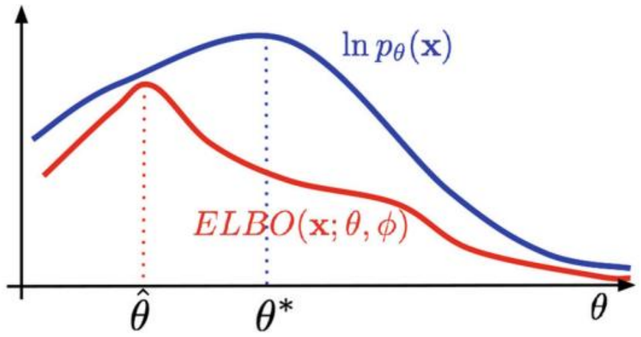
**Jensen's Inequality**
**:D**



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# ELBO

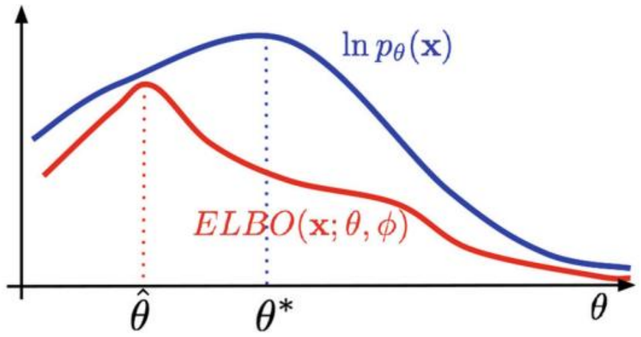
$$\text{ELBO}(\phi, \theta) = \sum_{i=1}^{n} E_{z_i \sim q} \left[ \log p_\theta(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i) \right]$$






Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# ELBO

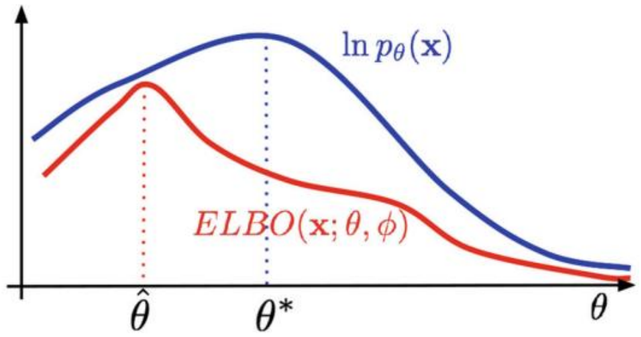



$$\begin{aligned}\text{ELBO}(\phi,\theta) &= \sum_{i=1}^{n} E_{z_i\sim q}\left[\log p_\theta(\boldsymbol{x}_i,\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right] \\ &= \sum_{i=1}^{n} \int q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\left[\log p_\theta(\boldsymbol{x}_i,\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right]\ d\boldsymbol{z}_i\end{aligned}$$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# ELBO

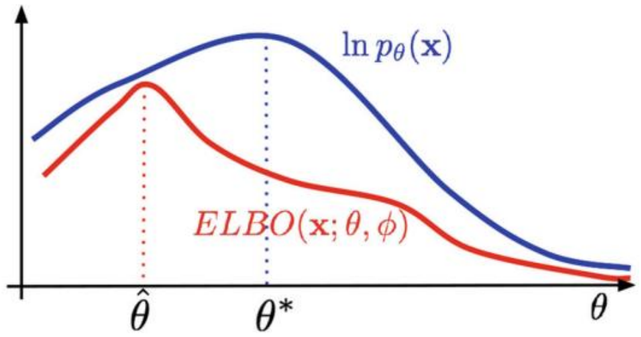



$$
\begin{aligned}
\text{ELBO}(\phi,\theta) &= \sum_{i=1}^{n} E_{z_i\sim q}\left[\log p_\theta(\boldsymbol{x}_i,\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right] \\
&= \sum_{i=1}^{n} \int q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\left[\log p_\theta(\boldsymbol{x}_i,\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right]\, d\boldsymbol{z}_i \\
&= \sum_{i=1}^{n} \int q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\left[\log p_\theta(\boldsymbol{x}_i \mid \boldsymbol{z}_i) + \log p(\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right]\, d\boldsymbol{z}_i
\end{aligned}
$$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# ELBO

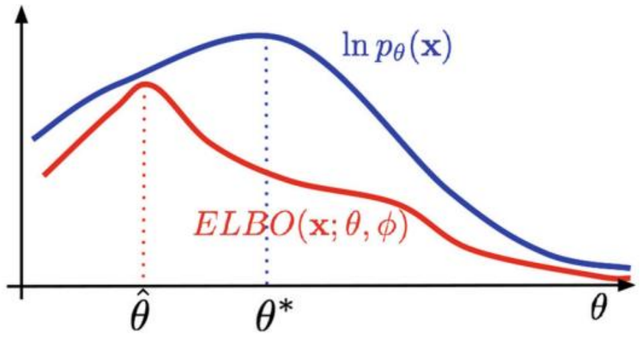



$$
\begin{aligned}
\text{ELBO}(\phi,\theta) &= \sum_{i=1}^{n} E_{z_i\sim q}\left[\log p_\theta(\boldsymbol{x}_i,\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right] \\
&= \sum_{i=1}^{n} \int q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\left[\log p_\theta(\boldsymbol{x}_i,\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right]\, d\boldsymbol{z}_i \\
&= \sum_{i=1}^{n} \int q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\left[\log p_\theta(\boldsymbol{x}_i \mid \boldsymbol{z}_i) + \log p(\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right]\, d\boldsymbol{z}_i \\
&= \sum_{i=1}^{n} E_{z_i\sim q}\left[\log p_\theta(\boldsymbol{x}_i \mid \boldsymbol{z}_i)\right] + \sum_{i=1}^{n} \int q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\left[\log p(\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right]\, d\boldsymbol{z}_i
\end{aligned}
$$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# ELBO

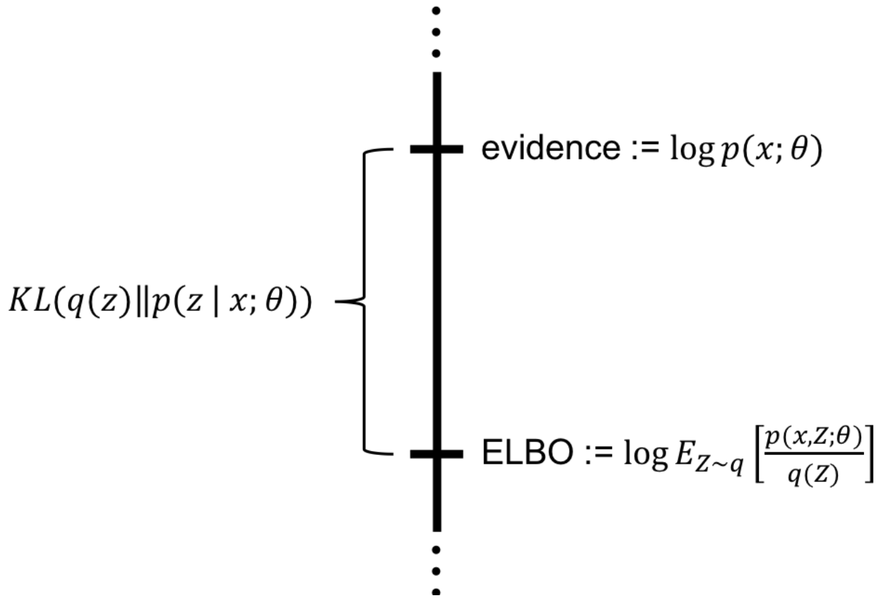



$$
\begin{aligned}
\text{ELBO}(\phi,\theta) &= \sum_{i=1}^{n} E_{z_i\sim q}\left[\log p_\theta(\boldsymbol{x}_i,\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right] \\
&= \sum_{i=1}^{n} \int q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\left[\log p_\theta(\boldsymbol{x}_i,\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right]\, d\boldsymbol{z}_i \\
&= \sum_{i=1}^{n} \int q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\left[\log p_\theta(\boldsymbol{x}_i \mid \boldsymbol{z}_i) + \log p(\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right]\, d\boldsymbol{z}_i \\
&= \sum_{i=1}^{n} E_{z_i\sim q}\left[\log p_\theta(\boldsymbol{x}_i \mid \boldsymbol{z}_i)\right] + \sum_{i=1}^{n} \int q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\left[\log p(\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right]\, d\boldsymbol{z}_i \\
&= \sum_{i=1}^{n} E_{z_i\sim q}\left[\log p_\theta(\boldsymbol{x}_i \mid \boldsymbol{z}_i)\right] + \boxed{\sum_{i=1}^{n} E_{z_i\sim q}\left[\log \frac{p(\boldsymbol{z}_i)}{q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)}\right]}
\end{aligned}
$$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# ELBO



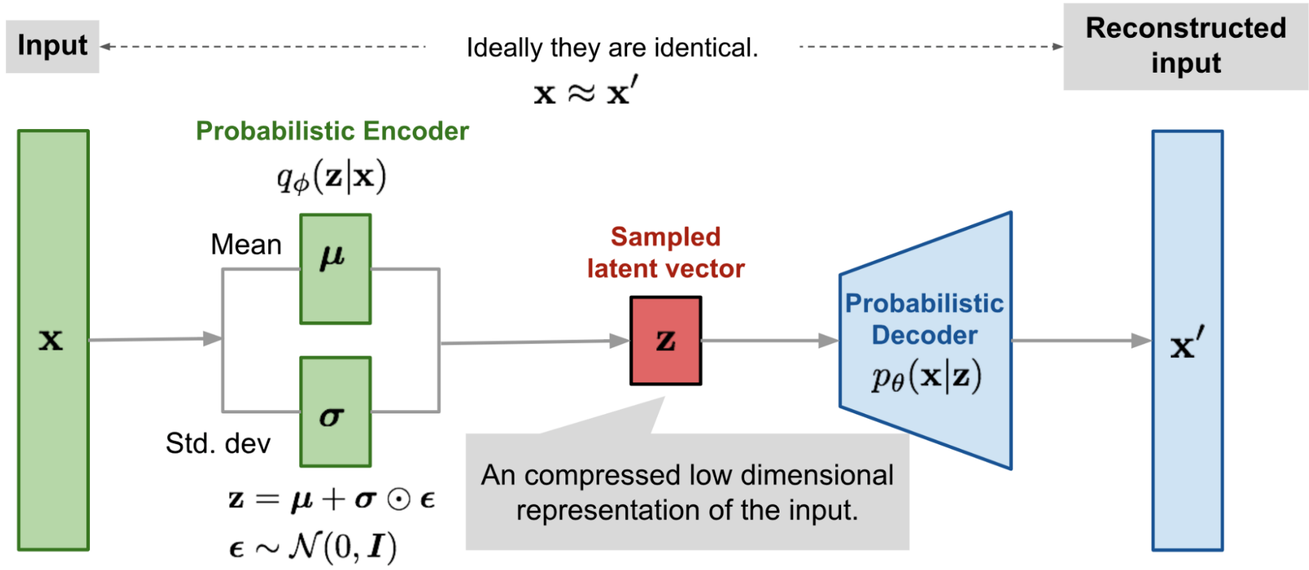

$$
\begin{aligned}
\text{ELBO}(\phi,\theta) &= \sum_{i=1}^{n} E_{z_i\sim q}\left[\log p_\theta(\boldsymbol{x}_i,\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right] \\
&= \sum_{i=1}^{n} \int q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\left[\log p_\theta(\boldsymbol{x}_i,\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right]\, d\boldsymbol{z}_i \\
&= \sum_{i=1}^{n} \int q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\left[\log p_\theta(\boldsymbol{x}_i \mid \boldsymbol{z}_i) + \log p(\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right]\, d\boldsymbol{z}_i \\
&= \sum_{i=1}^{n} E_{z_i\sim q}\left[\log p_\theta(\boldsymbol{x}_i \mid \boldsymbol{z}_i)\right] + \sum_{i=1}^{n} \int q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\left[\log p(\boldsymbol{z}_i) - \log q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\right]\, d\boldsymbol{z}_i \\
&= \sum_{i=1}^{n} E_{z_i\sim q}\left[\log p_\theta(\boldsymbol{x}_i \mid \boldsymbol{z}_i)\right] + \sum_{i=1}^{n} E_{z_i\sim q}\left[\log \frac{p(\boldsymbol{z}_i)}{q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)}\right] \\
&= \boxed{\sum_{i=1}^{n} E_{z_i\sim q}\left[\log p_\theta(\boldsymbol{x}_i \mid \boldsymbol{z}_i)\right] - KL(q(\boldsymbol{z}_i \mid \boldsymbol{x}_i)\,||\,p(\boldsymbol{z}_i))}
\end{aligned}
$$

**Function Loss**



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE




$$\max \mathcal{L}(\phi, \beta) = \mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) - D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right)$$



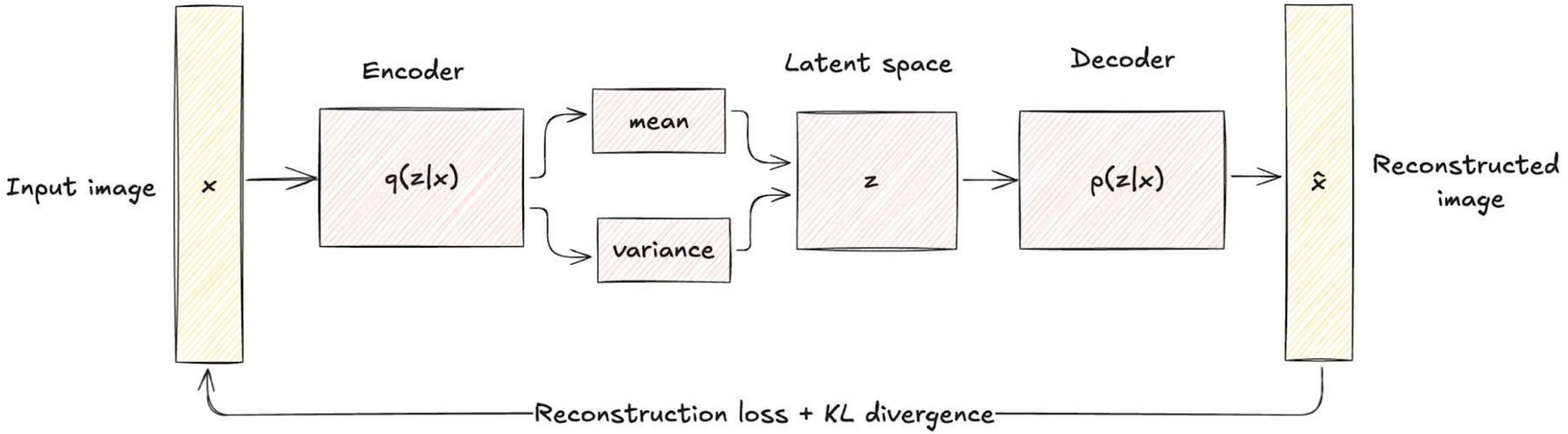
Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi, \beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**

La operación de muestreo es **no diferenciable**



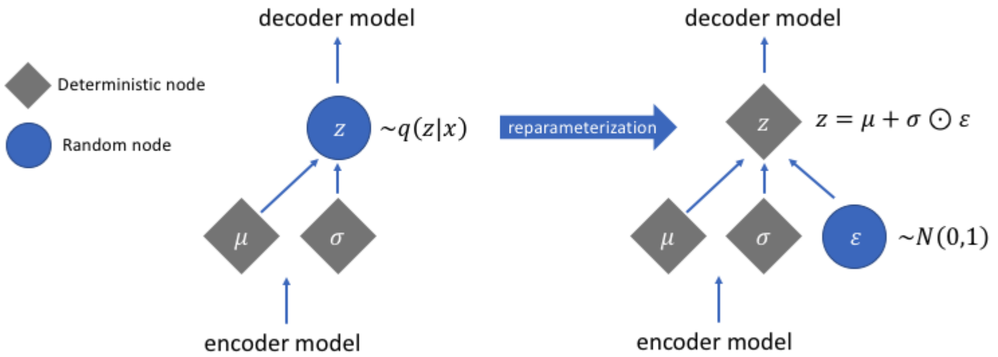



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi,\beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**




**Reparameterization trick**

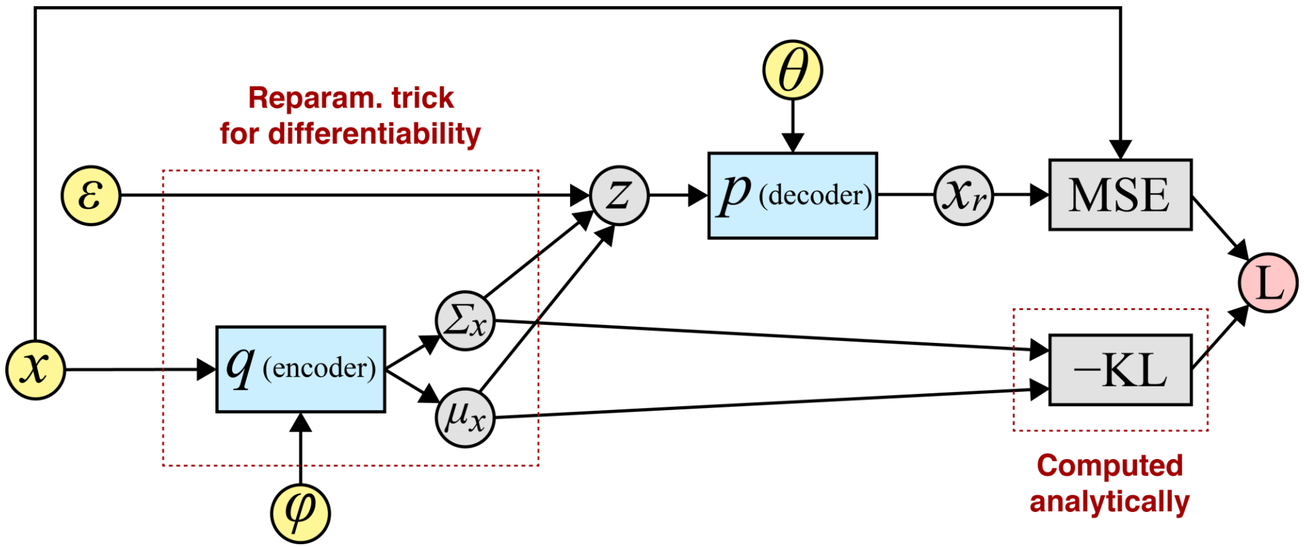


Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi,\beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

Decoder Encoder Fixed






Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi, \beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**

La divergencia KL entre $q(z|x)$ y $p(z)$ es: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) = \int q(z|x) \log \frac{q(z|x)}{p(z)} \partial z$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max\ \mathcal{L}_i(\phi,\beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**

La divergencia KL entre $q(z|x)$ y $p(z)$ es: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) = \int q(z|x)\log\frac{q(z|x)}{p(z)}\partial z$

Asumimos:

- $q(z|x) \sim \mathcal{N}(z;\mu,\sigma^2)$. Entonces:

$$q(z|x) = \frac{1}{(2\pi)^{d/2}\prod_{j=1}^{d}\sigma_j}\exp\left(-\frac{1}{2}\sum_{j=1}^{d}\frac{(z_j-\mu_j)^2}{\sigma_j^2}\right)$$

- $p(z) = \mathcal{N}(z;0,I)$, cuya densidad es:

$$p(z) = \frac{1}{(2\pi)^{d/2}}\exp\left(-\frac{1}{2}\sum_{j=1}^{d}z_j^2\right)$$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi,\beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**

La divergencia KL entre $q(z|x)$ y $p(z)$ es: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) = \int q(z|x)\log\frac{q(z|x)}{p(z)}\partial z$

Asumimos:

- $q(z|x) \sim \mathcal{N}(z;\mu,\sigma^2)$. Entonces:

$$q(z|x) = \frac{1}{(2\pi)^{d/2}\prod_{j=1}^{d}\sigma_j}\exp\left(-\frac{1}{2}\sum_{j=1}^{d}\frac{(z_j-\mu_j)^2}{\sigma_j^2}\right)$$

- $p(z) = \mathcal{N}(z;0,I)$, cuya densidad es:

$$p(z) = \frac{1}{(2\pi)^{d/2}}\exp\left(-\frac{1}{2}\sum_{j=1}^{d}z_j^2\right)$$

Desarrollamos:

$$\log\frac{q(z|x)}{p(z)} = \log\left(\frac{1}{(2\pi)^{d/2}\prod_{j=1}^{d}\sigma_j}\exp\left(-\frac{1}{2}\sum_{j=1}^{d}\frac{(z_j-\mu_j)^2}{\sigma_j^2}\right)\cdot(2\pi)^{d/2}\exp\left(\frac{1}{2}\sum_{j=1}^{d}z_j^2\right)\right) = -\sum_{j=1}^{d}\log\sigma_j - \frac{1}{2}\sum_{j=1}^{d}\frac{(z_j-\mu_j)^2}{\sigma_j^2} + \frac{1}{2}\sum_{j=1}^{d}z_j^2$$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max\ \mathcal{L}_i(\phi,\beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**

La divergencia KL entre $q(z|x)$ y $p(z)$ es: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) = \int q(z|x)\log\frac{q(z|x)}{p(z)}\partial z$

Asumimos:

- $q(z|x) \sim \mathcal{N}(z;\mu,\sigma^2)$. Entonces:

$$q(z|x) = \frac{1}{(2\pi)^{d/2}\prod_{j=1}^{d}\sigma_j}\exp\left(-\frac{1}{2}\sum_{j=1}^{d}\frac{(z_j-\mu_j)^2}{\sigma_j^2}\right)$$

- $p(z) = \mathcal{N}(z;0,I)$, cuya densidad es:

$$p(z) = \frac{1}{(2\pi)^{d/2}}\exp\left(-\frac{1}{2}\sum_{j=1}^{d}z_j^2\right)$$

Desarrollamos:

$$\log\frac{q(z|x)}{p(z)} = \log\left(\frac{1}{(2\pi)^{d/2}\prod_{j=1}^{d}\sigma_j}\exp\left(-\frac{1}{2}\sum_{j=1}^{d}\frac{(z_j-\mu_j)^2}{\sigma_j^2}\right)\cdot(2\pi)^{d/2}\exp\left(\frac{1}{2}\sum_{j=1}^{d}z_j^2\right)\right) = -\sum_{j=1}^{d}\log\sigma_j - \frac{1}{2}\sum_{j=1}^{d}\frac{(z_j-\mu_j)^2}{\sigma_j^2} + \frac{1}{2}\sum_{j=1}^{d}z_j^2$$

Reemplazando: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) = \mathbb{E}_{q(z|x)}\left[\log\frac{q(z|x)}{p(z)}\right] = -\sum_{j=1}^{d}\log\sigma_j - \frac{1}{2}\sum_{j=1}^{d}\mathbb{E}_{q(z|x)}\left[\frac{(z_j-\mu_j)^2}{\sigma_j^2}\right] + \frac{1}{2}\sum_{j=1}^{d}\mathbb{E}_{q(z|x)}[z_j^2]$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi, \beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**

La divergencia KL entre $q(z|x)$ y $p(z)$ es: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) = -\sum_{j=1}^{d} \log \sigma_j - \frac{1}{2}\sum_{j=1}^{d} \mathbb{E}_{q(z|x)}\left[\frac{(z_j - \mu_j)^2}{\sigma_j^2}\right] + \frac{1}{2}\sum_{j=1}^{d} \mathbb{E}_{q(z|x)}\left[z_j^2\right]$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi,\beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**

La divergencia KL entre $q(z|x)$ y $p(z)$ es: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) = -\sum_{j=1}^{d} \log \sigma_j - \frac{1}{2}\sum_{j=1}^{d} \mathbb{E}_{q(z|x)}\left[\frac{(z_j-\mu_j)^2}{\sigma_j^2}\right] + \frac{1}{2}\sum_{j=1}^{d} \mathbb{E}_{q(z|x)}\left[z_j^2\right]$

**Termino 1** $-\sum_{j=1}^{d} \log \sigma_j$ Este término es independiente de $z$, por lo que queda como está.



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi,\beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

Decoder Encoder Fixed

La divergencia KL entre $q(z|x)$ y $p(z)$ es: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) = -\sum_{j=1}^{d}\log\sigma_j - \frac{1}{2}\sum_{j=1}^{d}\mathbb{E}_{q(z|x)}\left[\frac{\left(z_j-\mu_j\right)^2}{\sigma_j^2}\right] + \frac{1}{2}\sum_{j=1}^{d}\mathbb{E}_{q(z|x)}\left[z_j^2\right]$

**Termino 1** $-\sum_{j=1}^{d}\log\sigma_j$ Este término es independiente de $z$, por lo que queda como está.

**Termino 2** $\mathbb{E}_{q(z|x)}\left[\frac{\left(z_j-\mu_j\right)^2}{\sigma_j^2}\right]$

Dado que $q(z|x) \sim \mathcal{N}\left(\mu_j,\sigma_j^2\right)$, la variable $z_j-\mu_j$ tiene distribución $\mathcal{N}\left(0,\sigma_j^2\right)$.

Entonces:

$$\mathbb{E}_{q(z|x)}\left[\frac{\left(z_j-\mu_j\right)^2}{\sigma_j^2}\right] = \frac{1}{\sigma_j^2}\mathbb{E}\left[\left(z_j-\mu_j\right)^2\right] = \frac{\sigma_j^2}{\sigma_j^2} = 1$$

Por lo tanto:

$$-\frac{1}{2}\sum_{j=1}^{d}\mathbb{E}_{q(z|x)}\left[\frac{\left(z_j-\mu_j\right)^2}{\sigma_j^2}\right] = -\frac{1}{2}\sum_{j=1}^{d}1 = -\frac{d}{2}$$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi,\beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

Decoder Encoder Fixed

La divergencia KL entre $q(z|x)$ y $p(z)$ es: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) = -\sum_{j=1}^{d}\log\sigma_j - \frac{1}{2}\sum_{j=1}^{d}\mathbb{E}_{q(z|x)}\left[\frac{(z_j-\mu_j)^2}{\sigma_j^2}\right] + \frac{1}{2}\sum_{j=1}^{d}\mathbb{E}_{q(z|x)}\left[z_j^2\right]$

**Termino 1** $-\sum_{j=1}^{d}\log\sigma_j$ Este término es independiente de $z$, por lo que queda como está.

**Termino 2** $\mathbb{E}_{q(z|x)}\left[\frac{(z_j-\mu_j)^2}{\sigma_j^2}\right]$

Dado que $q(z|x) \sim \mathcal{N}(\mu_j, \sigma_j^2)$, la variable $z_j - \mu_j$ tiene distribución $\mathcal{N}(0, \sigma_j^2)$.
Entonces:

$$\mathbb{E}_{q(z|x)}\left[\frac{(z_j-\mu_j)^2}{\sigma_j^2}\right] = \frac{1}{\sigma_j^2}\mathbb{E}\left[(z_j-\mu_j)^2\right] = \frac{\sigma_j^2}{\sigma_j^2} = 1$$

Por lo tanto:

$$-\frac{1}{2}\sum_{j=1}^{d}\mathbb{E}_{q(z|x)}\left[\frac{(z_j-\mu_j)^2}{\sigma_j^2}\right] = -\frac{1}{2}\sum_{j=1}^{d}1 = -\frac{d}{2}$$

**Termino 3** $\mathbb{E}_{q(z|x)}\left[z_j^2\right]$

Habíamos definido $z_j$ con media $\mu_j$ y varianza $\sigma_j^2$. Entonces:

$$\mathbb{E}\left[z_j^2\right] = \mathrm{var}(z_j) + \left(\mathbb{E}[z_j]\right)^2 = \sigma_j^2 + \mu_j^2$$

Por lo tanto:

$$\frac{1}{2}\sum_{j=1}^{d}\mathbb{E}_{q(z|x)}\left[z_j^2\right] = \frac{1}{2}\sum_{j=1}^{d}\left(\sigma_j^2 + \mu_j^2\right)$$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi,\beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**

La divergencia KL entre $q(z|x)$ y $p(z)$ es: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) = -\sum_{j=1}^{d} \log \sigma_j - \frac{1}{2}\sum_{j=1}^{d} 1 + \frac{1}{2}\sum_{j=1}^{d}\left(\sigma_j^2 + \mu_j^2\right)$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi,\beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**

La divergencia KL entre $q(z|x)$ y $p(z)$ es: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) = -\sum_{j=1}^{d} \log \sigma_j - \frac{1}{2}\sum_{j=1}^{d} 1 + \frac{1}{2}\sum_{j=1}^{d}(\sigma_j^2 + \mu_j^2)$

$$= \frac{1}{2}\sum_{j=1}^{d}\left[-2\log\sigma_j - 1 + \sigma_j^2 + \mu_j^2\right] = \frac{1}{2}\sum_{j=1}^{d}\left[-\log\sigma_j^2 - 1 + \sigma_j^2 + \mu_j^2\right]$$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi, \beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**

La divergencia KL entre $q(z|x)$ y $p(z)$ es: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) = -\sum_{j=1}^{d} \log \sigma_j - \frac{1}{2}\sum_{j=1}^{d} 1 + \frac{1}{2}\sum_{j=1}^{d}(\sigma_j^2 + \mu_j^2)$

$$= \frac{1}{2}\sum_{j=1}^{d}\left[-2\log\sigma_j - 1 + \sigma_j^2 + \mu_j^2\right] = \frac{1}{2}\sum_{j=1}^{d}\left[-\log\sigma_j^2 - 1 + \sigma_j^2 + \mu_j^2\right]$$

$$= \frac{1}{2}\sum_{j=1}^{d}\left[-\log \mathbf{var} - 1 + \exp(\log \mathbf{var}) + \mu_j^2\right]$$



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi,\beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**

La divergencia KL entre $q(z|x)$ y $p(z)$ es: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) = -\sum_{j=1}^{d}\log\sigma_j - \frac{1}{2}\sum_{j=1}^{d}1 + \frac{1}{2}\sum_{j=1}^{d}(\sigma_j^2+\mu_j^2)$

$$= \frac{1}{2}\sum_{j=1}^{d}\left[-2\log\sigma_j - 1 + \sigma_j^2 + \mu_j^2\right] = \frac{1}{2}\sum_{j=1}^{d}\left[-\log\sigma_j^2 - 1 + \sigma_j^2 + \mu_j^2\right]$$

$$= \frac{1}{2}\sum_{j=1}^{d}\left[-\log \mathbf{var} - 1 + \exp(\log \mathbf{var}) + \mu_j^2\right]$$

Preferimos usar $\log \mathbf{var}$ porque:

- **Garantizar la positividad:** La varianza debe ser estrictamente positiva. Al parametrizar la varianza como $\sigma^2 = \exp(\log \mathbf{var})$ nos aseguramos de que, sin importar el valor de $\log \mathbf{var}$ (que puede ser cualquier número real), $\sigma^2$ siempre será positivo.
- **Estabilidad numérica y optimización:** Trabajar en el espacio logarítmico puede mejorar la convergencia durante el entrenamiento, ya que los cambios relativos en la varianza se representan de forma más suave.



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi,\beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**

La divergencia KL entre $q(z|x)$ y $p(z)$ es: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) = -\sum_{j=1}^{d} \log \sigma_j - \frac{1}{2}\sum_{j=1}^{d} 1 + \frac{1}{2}\sum_{j=1}^{d}(\sigma_j^2 + \mu_j^2)$

$$= \frac{1}{2}\sum_{j=1}^{d}\left[-2\log\sigma_j - 1 + \sigma_j^2 + \mu_j^2\right] = \frac{1}{2}\sum_{j=1}^{d}\left[-\log\sigma_j^2 - 1 + \sigma_j^2 + \mu_j^2\right]$$

$$= \frac{1}{2}\sum_{j=1}^{d}\left[-\log \mathbf{var} - 1 + \exp(\log \mathbf{var}) + \mu_j^2\right]$$

**CODIGO:**

```
KLD = -0.5 * torch.sum(1 + logvar - mu.pow(2) - logvar.exp())
```

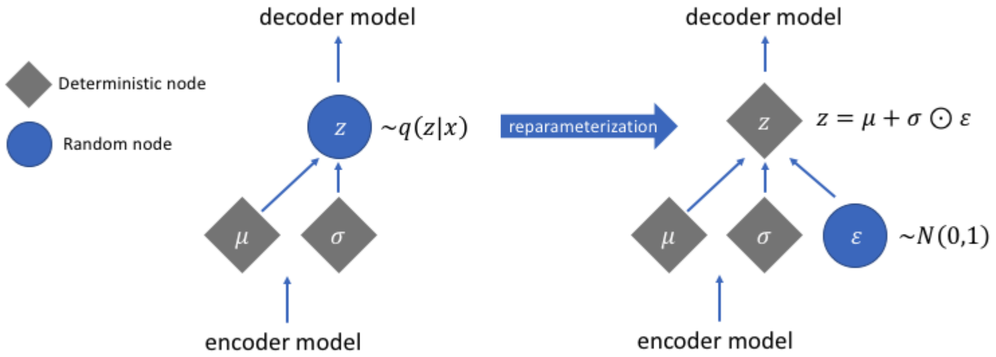


Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi, \beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**




Se quiere muestrear $z$ de la distribución $\mathcal{N}(\mu, \sigma^2)$. Reescribimos $z$ como:

$$z = \mu + \sigma \cdot \epsilon$$

donde $\epsilon \sim \mathcal{N}(0,1)$.

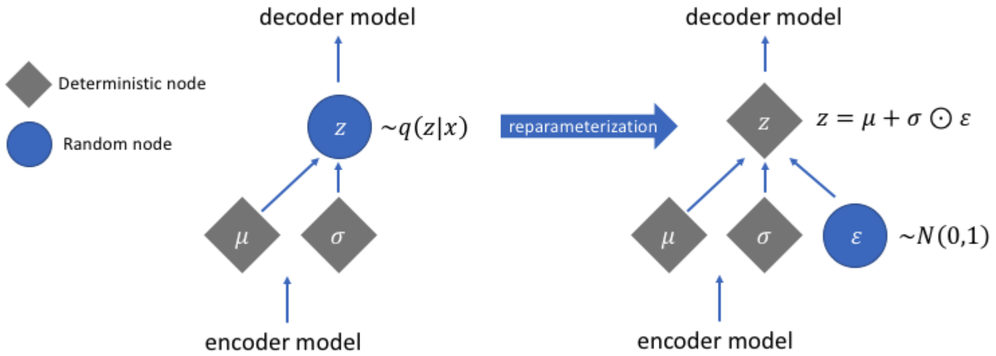


Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi, \beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**




Se quiere muestrear $z$ de la distribución $\mathcal{N}(\mu, \sigma^2)$. Reescribimos $z$ como:

$$z = \mu + \sigma \cdot \epsilon$$

donde $\epsilon \sim \mathcal{N}(0,1)$.

Reescribimos la varianza en función de **log var** :

$$\sigma = \sqrt{\sigma^2} = \sqrt{\exp(\log \textbf{var})} = \exp\left(\frac{1}{2}\log \textbf{var}\right)$$

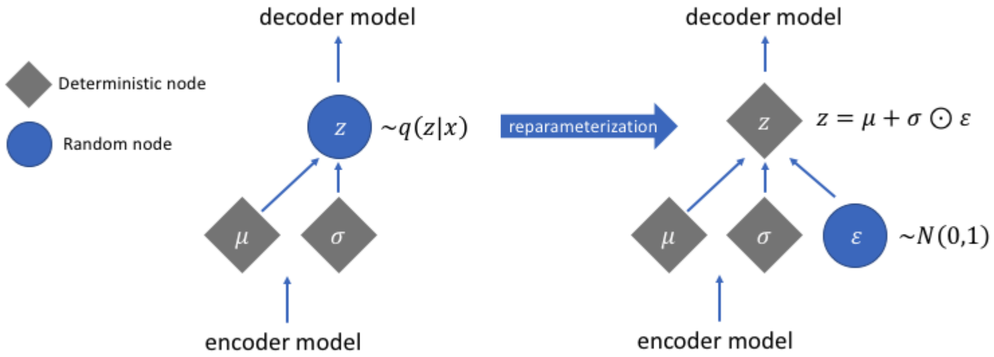


Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

# VAE

$$\max \ \mathcal{L}_i(\phi, \beta) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_l) - D_{\mathrm{KL}}\left(q_\phi(z|x_i)||p_\theta(z)\right)$$

**Decoder** **Encoder** **Fixed**




Se quiere muestrear $z$ de la distribución $\mathcal{N}(\mu, \sigma^2)$. Reescribimos $z$ como:

$$z = \mu + \sigma \cdot \epsilon$$

donde $\epsilon \sim \mathcal{N}(0,1)$.

Reescribimos la varianza en función de **log var** :

$$\sigma = \sqrt{\sigma^2} = \sqrt{\exp(\log \mathbf{var})} = \exp\left(\frac{1}{2}\log \mathbf{var}\right)$$

```
def reparameterize(self, mu, logvar):
    std = torch.exp(0.5 * logvar)
    eps = torch.randn_like(std)
    return mu + eps * std
```



Diederik P. Kingma and Max Welling (2013) "Auto-Encoding Variational Bayes". arXiv preprint arXiv:1312.6114

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

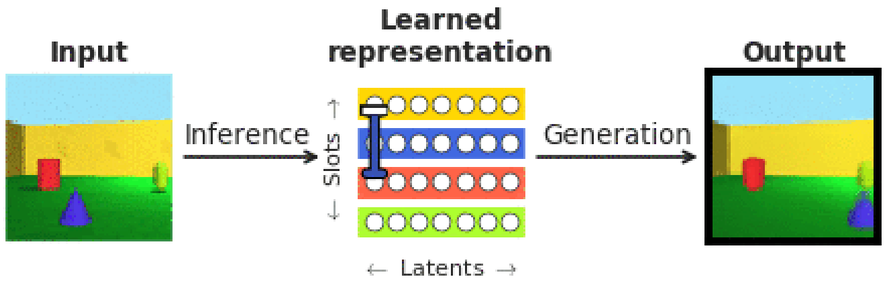

# 2.

## β-*VAE*

TRANSFORMATEC

# β-*VAE*

## Disentangled representation




La idea principal es aprender representaciones latentes donde cada variable capture de forma independiente uno de los factores generativos subyacentes a los datos, lo que puede facilitar tareas posteriores como clasificación, transfer learning o zero-shot inference.

Una representación bien desenredada debe tener variables que respondan de forma selectiva a cambios en factores concretos, por ejemplo, pose, iluminación o escala en imágenes.

Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

# β-*VAE*

Los autores proponen obtener representaciones latentes más disentangled (separadas e interpretables) haciendo: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) < \boldsymbol{C}$



Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

# β-*VAE*

Los autores proponen obtener representaciones latentes más disentangled (separadas e interpretables) haciendo: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) < \boldsymbol{C}$

La divergencia KL actúa como un information bottleneck, forzando a la red a utilizar su limitada capacidad de codificación de la manera más eficiente posible.



Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

# β-*VAE*

Los autores proponen obtener representaciones latentes más disentangled (separadas e interpretables) haciendo: $\mathrm{D}_{KL}\big(p(z|x)||p(z)\big) < \boldsymbol{C}$

La divergencia KL actúa como un information bottleneck, forzando a la red a utilizar su limitada capacidad de codificación de la manera más eficiente posible.

## Formalmente:

Sea $q_\phi(z|x)$ la distribución a posteriori y $p(z)$ el prior elegido, normalmente una gaussiana isotrópica factorizable: $p(z) = \prod_{m=1}^{M} p(z_m)$

Al imponer la condición $D_{KL}\left(q_\phi(z|x)|p(z)\right) < C$ estamos limitando el flujo de información entre $x$ y $z$.

Esta relación implica: $I(x;z) \leq E_{x\sim p_{data}(x)}\left[D_{KL}\left(q_\phi(z|x)|p(z)\right)\right] < C$

Así, un canal del espacio latente sólo puede transmitir, en promedio, menos de $C$ unidades de información (nats).

Si asumimos que la distribución a posteriori también se aproxima a una distribución que factorizable: $q_\phi(z|x) = \prod_{m=1}^{M} q_\phi(z_m|x)$

se tiene que $D_{KL}\left(q_\phi(z|x)|p(z)\right) = \sum_{m=1}^{M} D_{KL}\left(q_\phi(z_m|x)|p(z_m)\right)$

Esto significa que las $C$ unidades de información se reparte entre las $M$ dimensiones. En presencia de una restricción tan estricta, la única forma de usar eficientemente la capacidad es asignar cada unidad latente a capturar información de un único factor generativo.



Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

# β-*VAE*

Bajo una restricción de capacidad, la red debe seleccionar qué información preservar para poder reconstruir $x$ de forma adecuada. Si los factores generativos subyacentes son independientes, la asignación óptima en términos de eficiencia de información es que cada dimensión $z_m$ codifique preferentemente uno de esos factores.

Formalmente, si se intenta codificar información redundante o enredada (entangled), se desperdicia capacidad, ya que se estarían utilizando más de $C$ unidades para representar la misma variación. Así, la solución óptima se acerca a:

$$q_\phi(z|x) \approx \prod_{k=1}^{K} q_\phi(z_k|v_k)$$

donde cada $z_k$ está alineado con un factor generativo independiente $v_k$. Esto se traduce en una representación disentangled (desenredada).



Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

# β-*VAE*

Para forzar que el modelo use una capacidad limitada en el espacio latente y, por ende, promueva representaciones disentangled, se plantea la siguiente optimización restringida:

**Maximizar:** $\max_{\theta,\phi} \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]$

**Sujeto a:** $D_{KL}\left(q_\phi(z|x) \,|\, p(z)\right) \leq C$



Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

# β-*VAE*

Para forzar que el modelo use una capacidad limitada en el espacio latente y, por ende, promueva representaciones disentangled, se plantea la siguiente optimización restringida:

**Maximizar:** $\max_{\theta,\phi} \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]$

**Sujeto a:** $D_{KL}\left(q_\phi(z|x) \,|\, p(z)\right) \leq C$

Este problema lo solucionamos con una generalización a los multiplicadores de Lagrange: las **condiciones de KKT.**



Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

# Karush–Kuhn–Tucker *conditions (KKT)*

$$\min f(x) \quad \text{sujeto a } g_i(x) \leq 0, h_j(x) = 0$$

$$\mathcal{L}(x, \mu, \lambda) = f(x) + \mu^{\mathsf{T}} g(x) + \lambda^{\mathsf{T}} h(x)$$

**Stationarity** $\frac{\partial}{\partial x}\mathcal{L}(x, \mu, \lambda) = \frac{\partial}{\partial x} f(x) + \sum_i \mu_i \frac{\partial}{\partial x} g(x) + \sum_i \lambda_i \frac{\partial}{\partial x} g(x)$

**Primal feasibility** $g_i(x) \leq 0 \qquad h_i(x) = 0$

**Dual feasibility** $\lambda_i g_i(x) = 0$

**Complementary slackness** $\lambda_i \geq 0$

William Karush (1939) "Minima of functions of several variables with inequalities as side constraints". M. Sc. Dissertation. Dept. of Mathematics, Univ. of Chicago.

Harold W. Kuhn and Albert W. Tucker (1951) "Nonlinear programming". Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability (pp. 481–492). University of California Press.

# β-*VAE*

**Maximizar:** $\max_{\theta,\phi} \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]$

**Sujeto a:** $D_{KL}\left(q_\phi(z|x) \,|\, p(z)\right) \leq C$

Definimos la función Lagrangiana: $\mathcal{L}(\theta, \phi, \beta) = \mathbb{E}_{\boldsymbol{z}\sim q_\phi(\boldsymbol{z}|x)} \log p_\theta(x|z) - \beta\left(D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C\right)$

sujeto a: $g(\theta, \phi) = D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C \leq 0$



Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

# β-*VAE*

**Maximizar:** $\max_{\theta,\phi} \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]$

**Sujeto a:** $D_{KL}\left(q_\phi(z|x) \,|\, p(z)\right) \leq C$

Definimos la función Lagrangiana: $\mathcal{L}(\theta, \phi, \beta) = \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) - \beta\left(D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C\right)$

sujeto a: $g(\theta, \phi) = D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C \leq 0$

**Stationarity** $\nabla_{\theta,\phi}\mathcal{L}(\theta, \phi, \beta) = 0$

**Primal feasibility** $g(\theta, \phi) = D_{KL}\left(q_\phi(z|x) \,|\, p(z)\right) - C \leq 0$

**Dual feasibility** $\beta \geq 0$

**Complementary slackness** $\beta \cdot g(\theta, \phi) = \beta\left(D_{KL}\left(q_\phi(z|x) \,|\, p(z)\right) - C\right) = 0$



Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

# β-*VAE*

**Maximizar:** $\max_{\theta,\phi} \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]$

**Sujeto a:** $D_{KL}\left(q_\phi(z|x) \,|\, p(z)\right) \leq C$

Definimos la función Lagrangiana: $\mathcal{L}(\theta, \phi, \beta) = \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) - \beta\left(D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C\right)$

sujeto a: $g(\theta, \phi) = D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C \leq 0$

**Stationarity** $\nabla_{\theta,\phi}\mathcal{L}(\theta, \phi, \beta) = 0$

**Primal feasibility** $g(\theta, \phi) = D_{KL}\left(q_\phi(z|x) \,|\, p(z)\right) - C \leq 0$

**Dual feasibility** $\beta \geq 0$

**Complementary slackness** $\beta \cdot g(\theta, \phi) = \beta\left(D_{KL}\left(q_\phi(z|x) \,|\, p(z)\right) - C\right) = 0$

- Si la restricción es activa (es decir, $D_{KL} = C$), entonces se debe tener $\beta > 0$.
- Si la restricción es inactiva (es decir, $D_{KL} < C$), entonces $\beta = 0$.



Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

# β-*VAE*

**Maximizar:** $\max_{\theta,\phi} \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]$

**Sujeto a:** $D_{KL}\left(q_\phi(z|x) \,|\, p(z)\right) \leq C$

Definimos la función Lagrangiana:

$$\mathcal{L}(\theta, \phi, \beta) = \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) - \beta\left(D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C\right)$$

$$= \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) - \beta D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C\beta$$



Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

# β-*VAE*

**Maximizar:** $\max_{\theta,\phi} \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]$

**Sujeto a:** $D_{KL}\left(q_\phi(z|x) \,|\, p(z)\right) \leq C$

Definimos la función Lagrangiana:

$$\mathcal{L}(\theta, \phi, \beta) = \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) - \beta\left(D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C\right)$$

$$= \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) - \beta D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C\beta$$

Como nosotros vamos a optimizar $\theta, \phi$;
entonces $C, \beta$ son constantes



Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

# β-*VAE*

**Maximizar:** $\max_{\theta,\phi} \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]$

**Sujeto a:** $D_{KL}\left(q_\phi(z|x) \,|\, p(z)\right) \leq C$

Definimos la función Lagrangiana:

$$\mathcal{L}(\theta, \phi, \beta) = \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) - \beta\left(D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C\right)$$

$$= \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) - \beta D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C\beta$$

Entonces, la función a optimizar es:

$$\max_{\theta,\phi} \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) - \beta D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right)$$



Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

# β-*VAE*

**Maximizar:** $\max_{\theta,\phi} \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]$

**Sujeto a:** $D_{KL}\left(q_\phi(z|x) \,|\, p(z)\right) \leq C$

Definimos la función Lagrangiana:

$$\mathcal{L}(\theta, \phi, \beta) = \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) - \beta\left(D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C\right)$$

$$= \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) - \beta D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C\beta$$

Entonces, la función a optimizar es:

$$\min_{\theta,\phi} -\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) + \beta D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right)$$



Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

# β-*VAE*

**Maximizar:** $\max_{\theta,\phi} \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]$

**Sujeto a:** $D_{KL}\left(q_\phi(z|x) \,|\, p(z)\right) \leq C$

Definimos la función Lagrangiana: $\mathcal{L}(\theta, \phi, \beta) = \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) - \beta\left(D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C\right)$

$= \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) - \beta D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right) - C\beta$

Entonces, la función a optimizar es: $\min_{\theta,\phi} -\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|x)} \log p_\theta(x|z) + \beta D_{\mathrm{KL}}\left(q_\phi(z|x)||p_\theta(z)\right)$

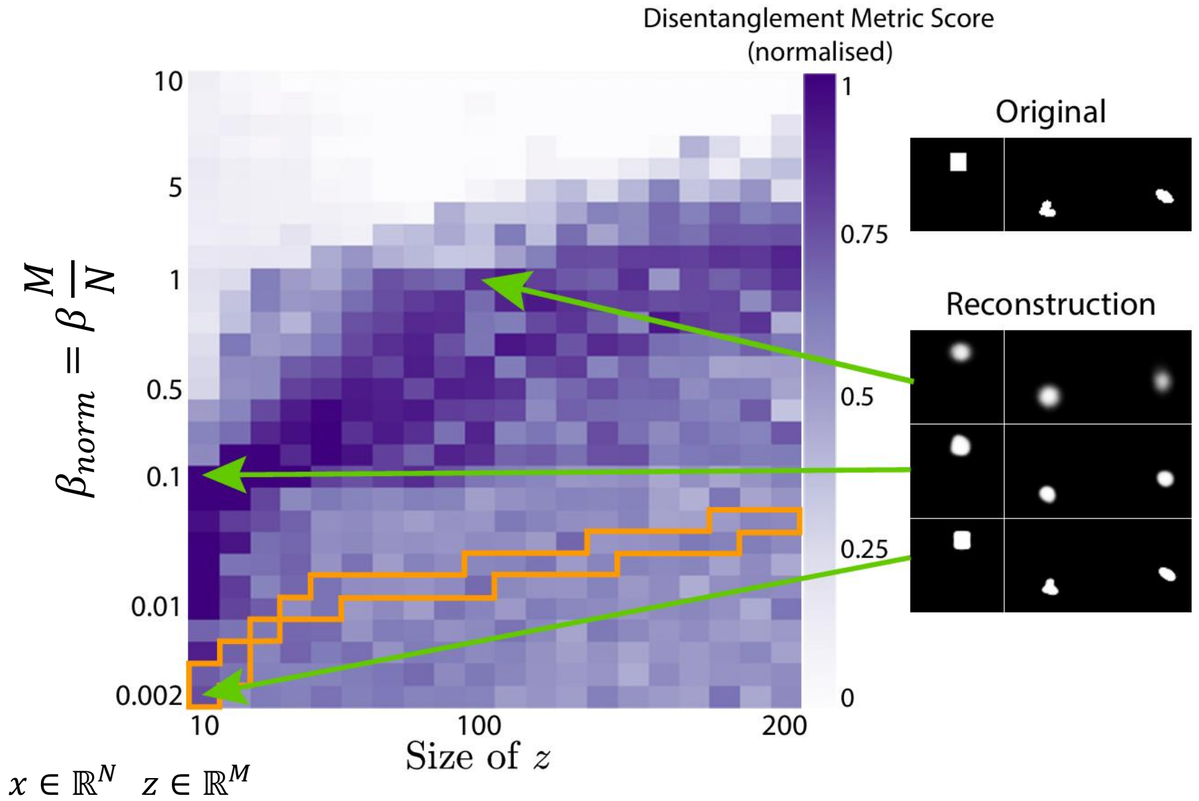
En el VAE clásico, el equilibrio entre la reconstrucción y la regularización se da para $\beta = 1$. Al aumentar β, se refuerza la penalización sobre el término de divergencia KL, lo que obliga a la distribución aproximada $q_\phi(z|x)$ a acercarse aún más a la distribución a priori $p(z)$. Con $\beta > 1$:

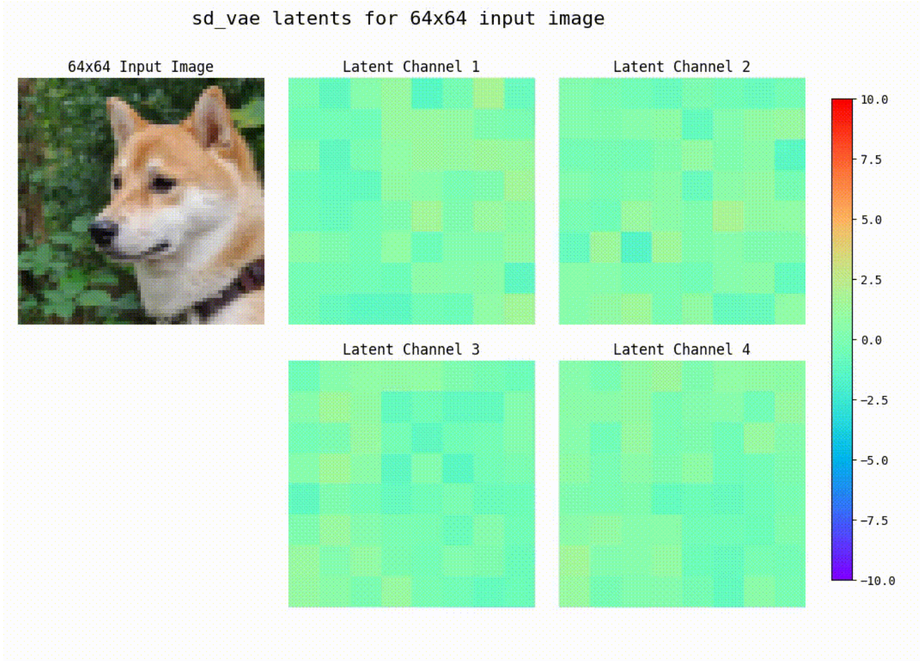
- **Reducción de capacidad latente:** Al imponer una penalización mayor sobre la divergencia KL, se restringe la cantidad de información que puede fluir en el espacio latente.
- **Disentanglement:** La limitación de la capacidad favorece que cada dimensión del latente capture factores de variación específicos y, en muchos casos, se observe que estos factores se vuelven más independientes entre sí.
- **Trade-off:** Un β demasiado alto puede degradar la calidad de la reconstrucción, ya que el modelo se centra excesivamente en cumplir la restricción de la divergencia KL.



Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

# β-*VAE*






Irina Higgins et al. (2017) "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". International Conference on Learning Representations (ICLR).

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

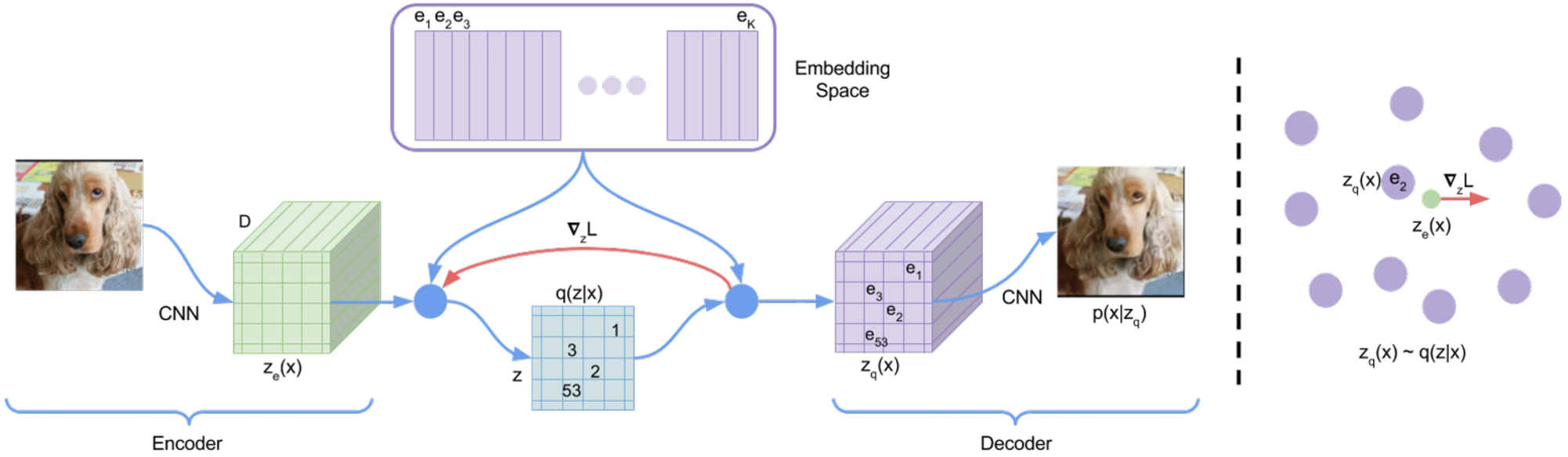

# 3.

## VQ-*VAE*

TRANSFORMATEC

# VQ-VAE

(Vector Quantised-Variational AutoEncoder)






Aaron van den Oord et al. (2018) "Neural Discrete Representation Learning". Advances in neural information processing systems, 30.

# VQ-VAE

(Vector Quantised-Variational AutoEncoder)

$$\mathcal{L} = \underbrace{\|x - \hat{x}\|^2}_{\text{Reconstruction Loss}} + \underbrace{\left\|\text{detach}\left[z_{e(x)}\right] - e\right\|^2}_{\text{Codebook Loss}} + \underbrace{\beta\left\|z_{e(x)} - \text{detach}[e]\right\|^2}_{\text{Commitment Loss}}$$

donde:

- $x$ es la entrada original.
- $\hat{x}$ es la reconstrucción generada por el decodificador.
- $z_{e(x)}$ es la salida del encoder.
- $e$ es el embedding del codebook más cercano.
- detach$[\cdot]$ representa a stop-gradient.
- $\beta$ es un hiperparámetro que modula la importancia de Commitment Loss.



Aaron van den Oord et al. (2018) "Neural Discrete Representation Learning". Advances in neural information processing systems, 30.

# VQ-VAE

(Vector Quantised-Variational AutoEncoder)

$$\mathcal{L} = \underbrace{\|x - \hat{x}\|^2}_{\text{Reconstruction Loss}} + \underbrace{\left\|\text{detach}\left[z_{e(x)}\right] - e\right\|^2}_{\text{Codebook Loss}} + \underbrace{\beta\left\|z_{e(x)} - \text{detach}[e]\right\|^2}_{\text{Commitment Loss}}$$

**Reconstruction Loss** Mide la capacidad del decodificador de reconstruir la entrada original a partir de la representación latente cuantizada.

**Codebook Loss** Actualiza los vectores del codebook (el conjunto de embeddings discretos) para que se acerquen a las salidas del encoder.

**Commitment Loss** Se quiere evitar que el encoder cambie arbitrariamente sus salidas, haciendo que sean consistentes con el embedding seleccionado.
Al penalizar la diferencia entre la salida del encoder y el embedding asignado, se compromete al encoder a trabajar en conjunto con el codebook, estabilizando así el proceso de aprendizaje.



Aaron van den Oord et al. (2018) "Neural Discrete Representation Learning". Advances in neural information processing systems, 30.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

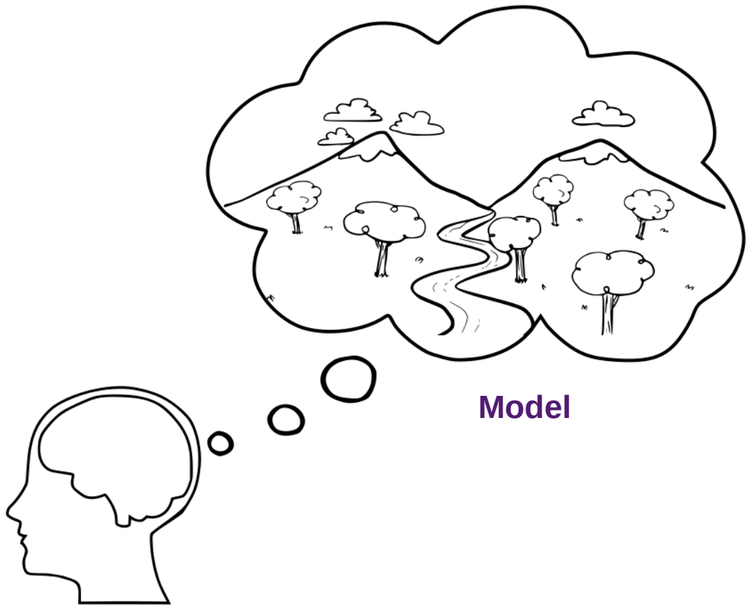
# 3.

## World *Models*


TRANSFORMATEC

# World *Models*

Complex world

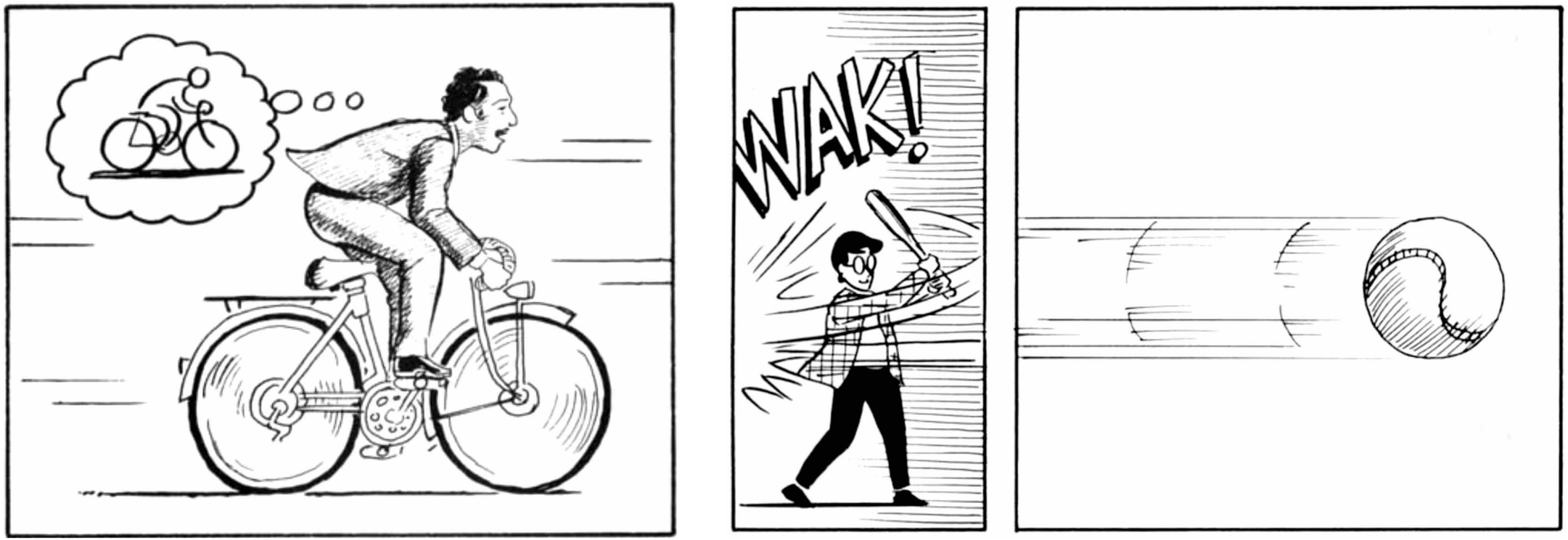


Model



David Ha and Jürgen Schmidhuber (2018) "World Models". arXiv preprint arXiv:1803.10122.

# World *Models*

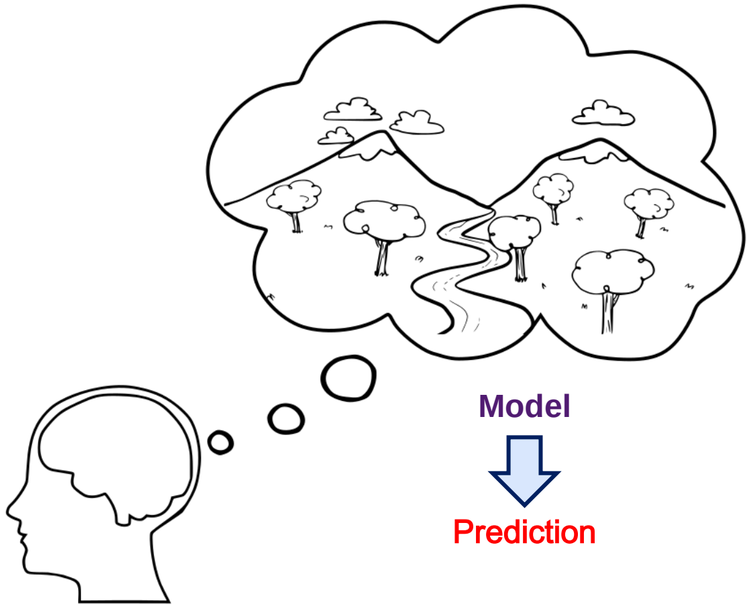





David Ha and Jürgen Schmidhuber (2018) "World Models". arXiv preprint arXiv:1803.10122.

# World *Models*

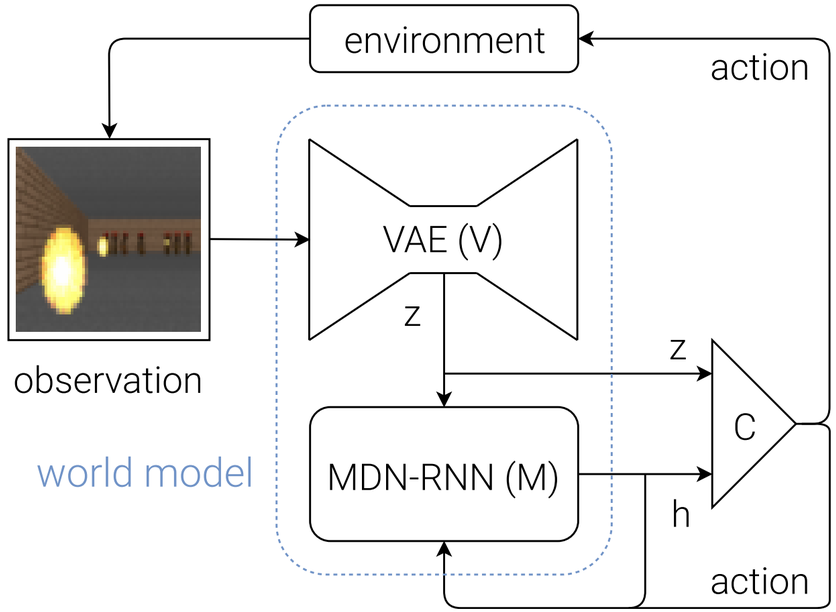
Complex world



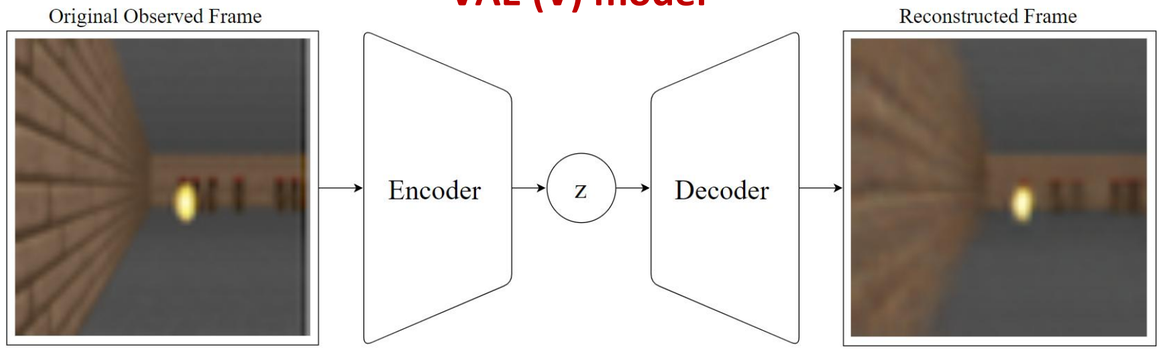



David Ha and Jürgen Schmidhuber (2018) "World Models". arXiv preprint arXiv:1803.10122.

# Agent *Model*



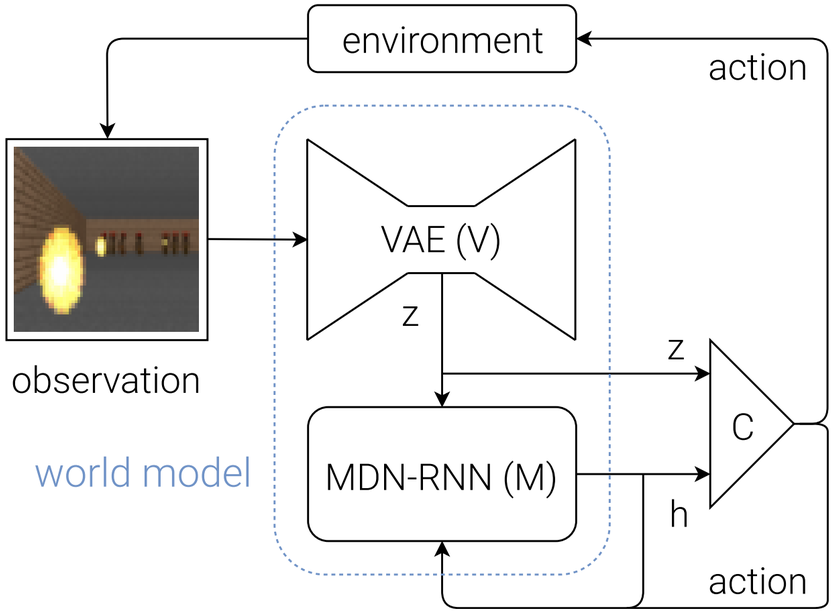

**VAE (V) model**



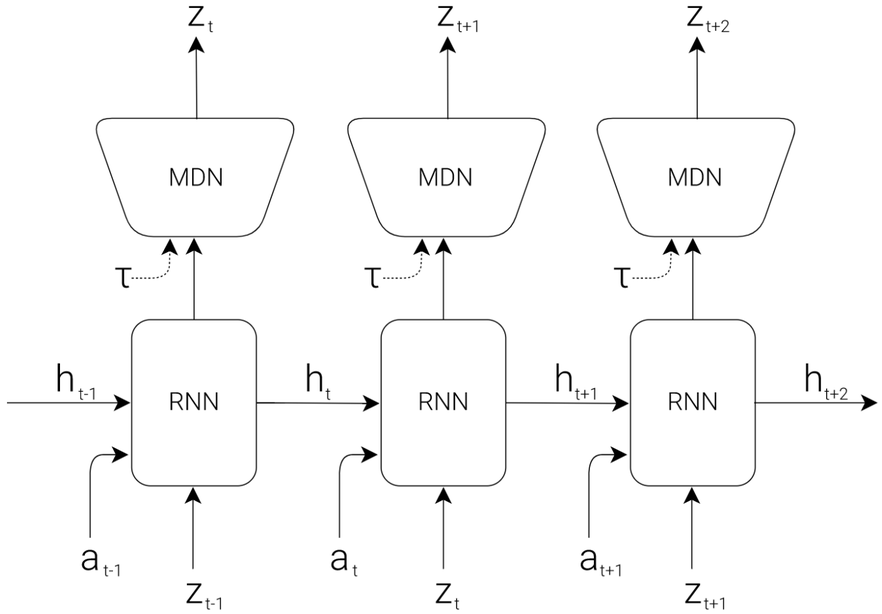



David Ha and Jürgen Schmidhuber (2018) "World Models". arXiv preprint arXiv:1803.10122.

# Agent *Model*




**MDN-RNN (M) model**

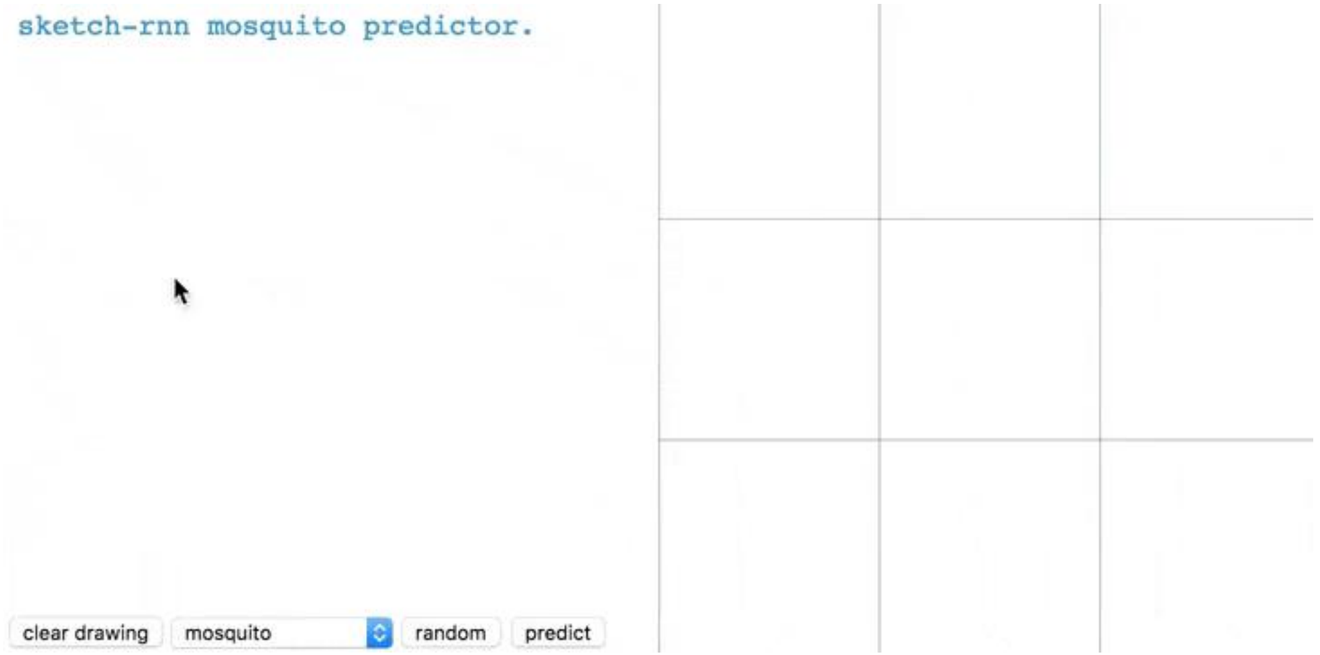


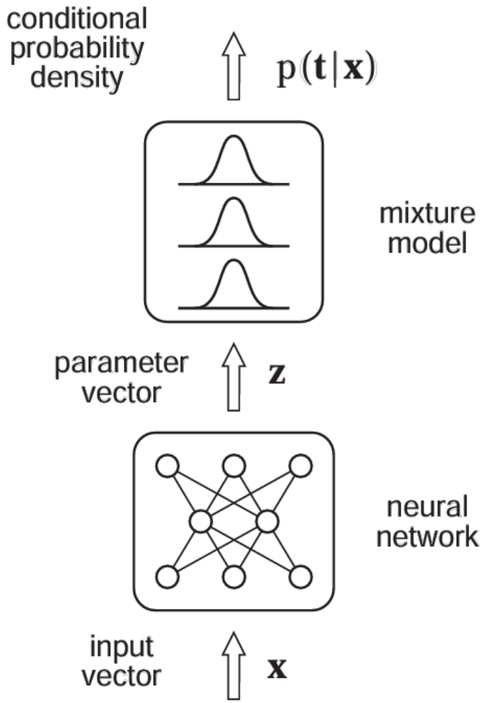



David Ha and Jürgen Schmidhuber (2018) "World Models". arXiv preprint arXiv:1803.10122.

# Agent *Model*

## Mixture Density Network

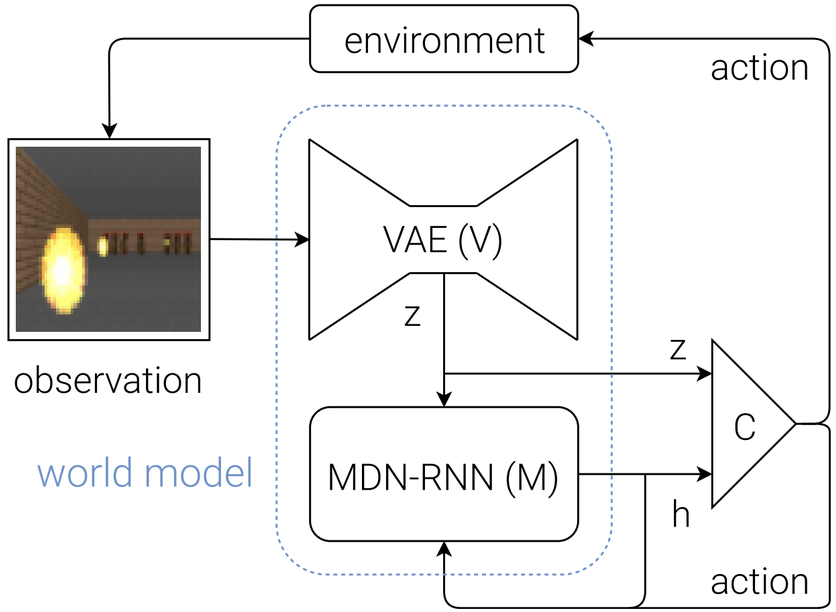





David Ha and Jürgen Schmidhuber (2018) "World Models". arXiv preprint arXiv:1803.10122.

# Agent *Model*




**Controller (C) Model**

$$a_t = W_c \, [z_t \; h_t] + bc$$

where:

| | |
|---|---|
| $W_c$ | weight matrix |
| $b_c$ | bias vector |
| $z_t$ | latent vector |
| $h_t$ | hidden state |
| $a_t$ | action |



David Ha and Jürgen Schmidhuber (2018) "World Models". arXiv preprint arXiv:1803.10122.

# Agent *Model*

At each time step, our agent receives an **observation** from the environment.

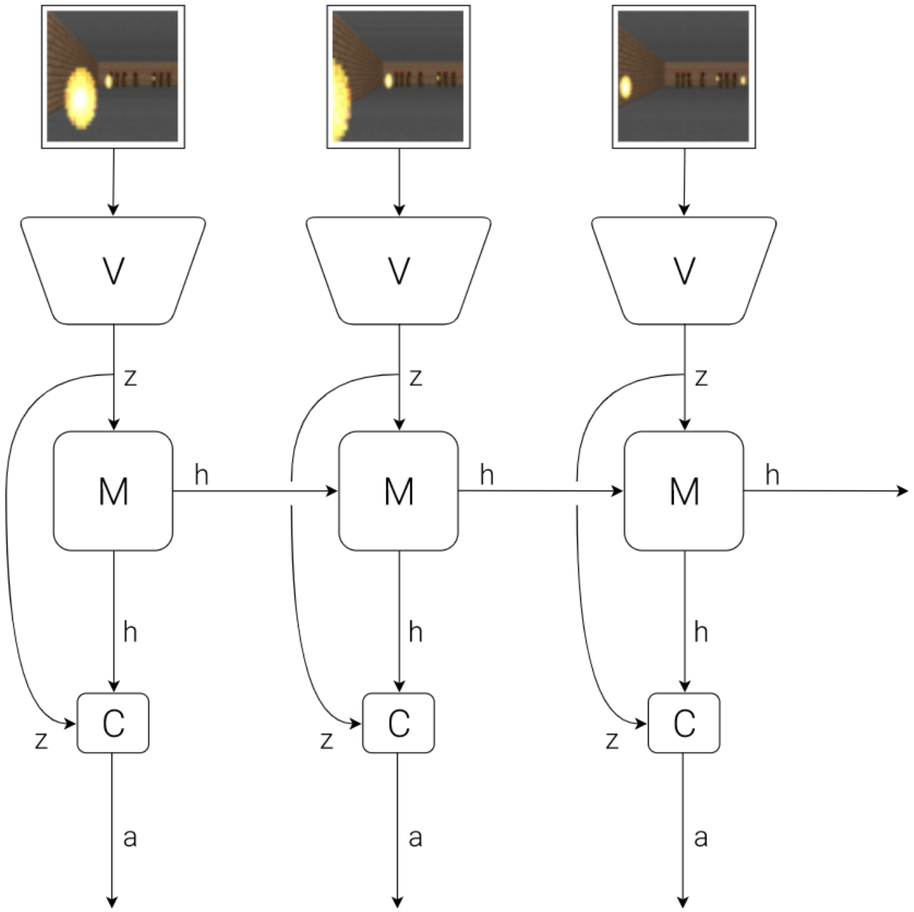
World Model

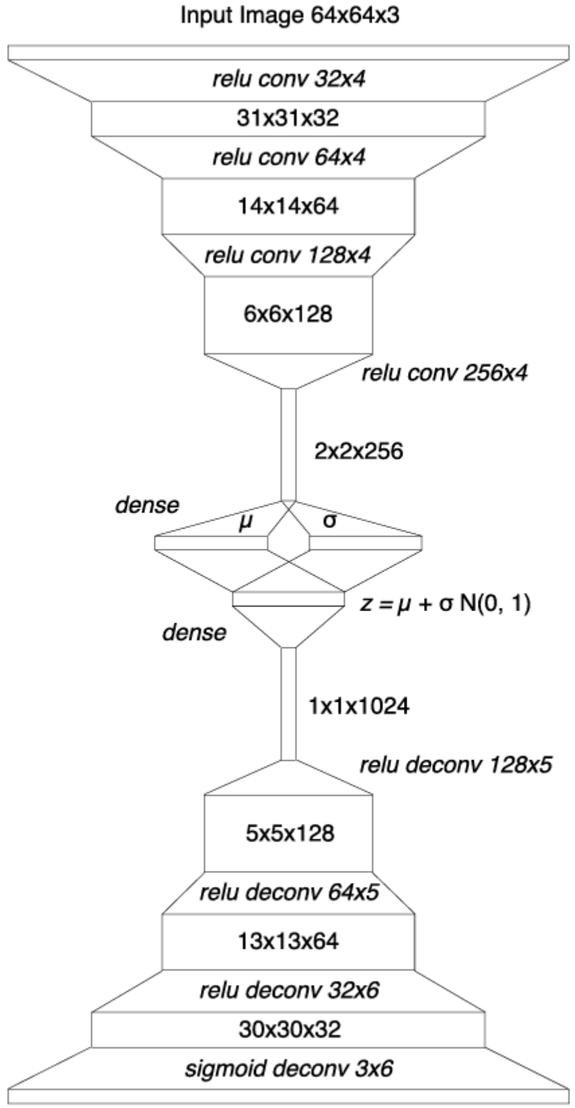
The **Vision Model (V)** encodes the high-dimensional observation into a low-dimensional latent vector.

The **Memory RNN (M)** integrates the historical codes to create a representation that can predict future states.

A small **Controller (C)** uses the representations from both **V** and **M** to select good actions.

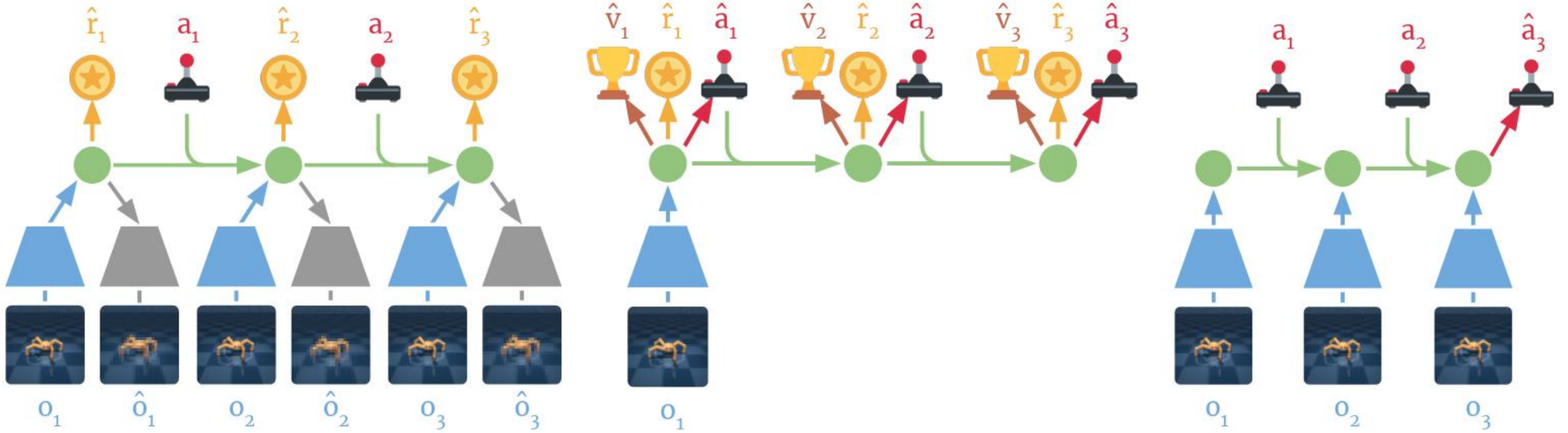
The agent performs **actions** that go back and affect the environment.




David Ha and Jürgen Schmidhuber (2018) "World Models". arXiv preprint arXiv:1803.10122.

# Dream to *Control*

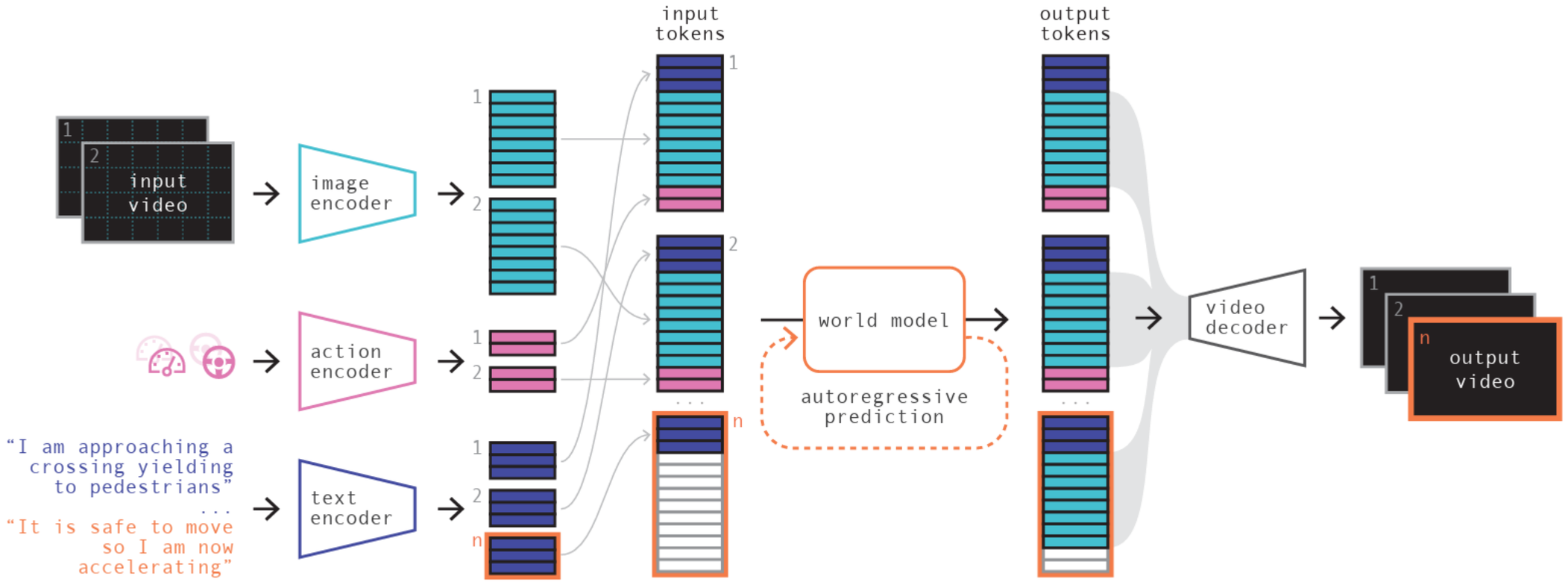



(a) Learn dynamics from experience (b) Learn behavior in imagination (c) Act in the environment



Danijar Hafner et al. (2020) "Dream to control: Learning behaviors by latent imagination". International Conference on Learning Representations (ICRL).

# GAIA-1






Anthony Hu et al. (2023) "GAIA-1: A Generative World Model for Autonomous Driving". arXiv preprint arXiv:1803.10122.

# GAIA-1




Anthony Hu et al. (2023) "GAIA-1: A Generative World Model for Autonomous Driving". arXiv preprint arXiv:1803.10122.

# GAIA-1




Anthony Hu et al. (2023) "GAIA-1: A Generative World Model for Autonomous Driving". arXiv preprint arXiv:1803.10122.

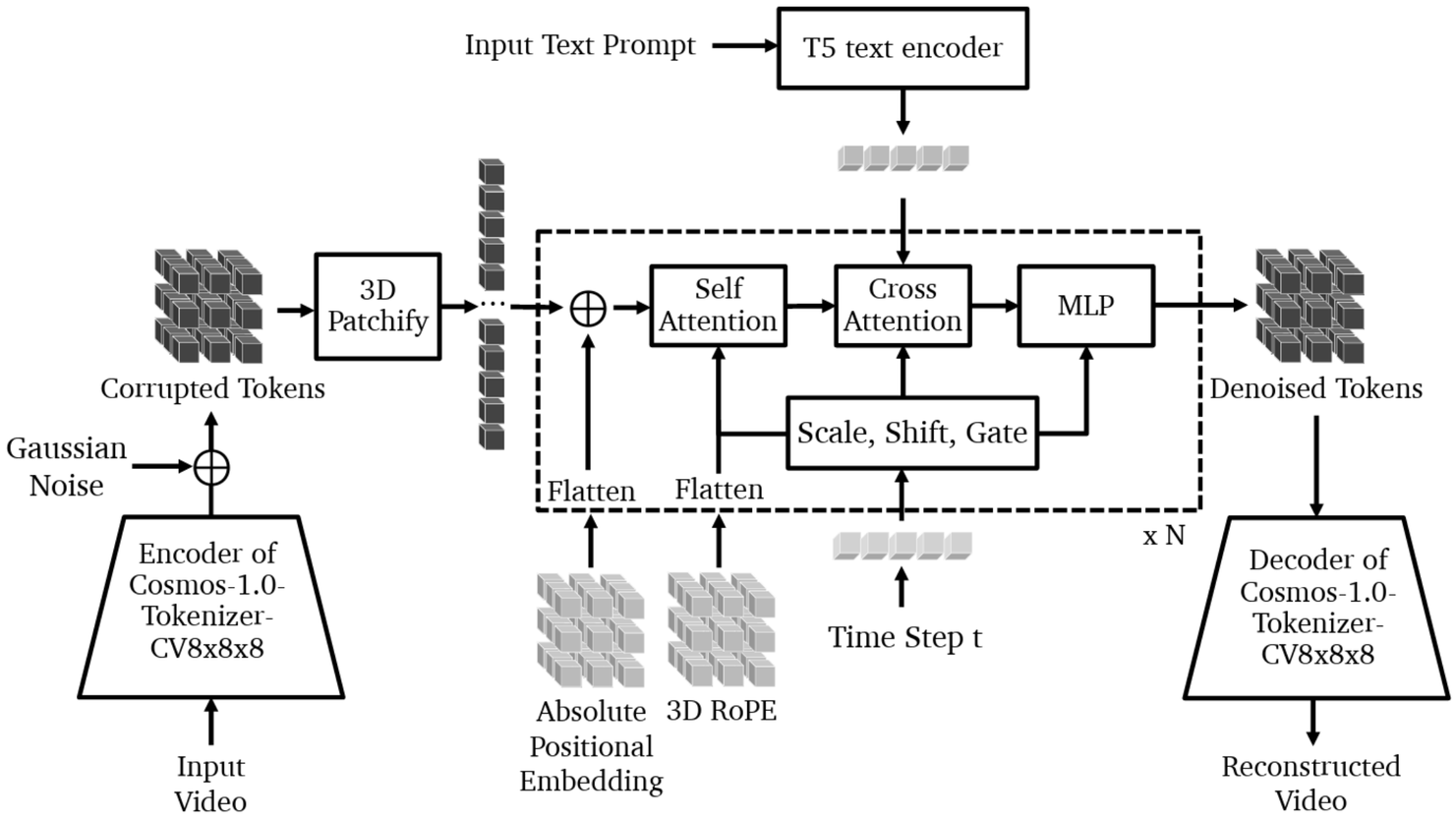
# GAIA-1





Anthony Hu et al. (2023) "GAIA-1: A Generative World Model for Autonomous Driving". arXiv preprint arXiv:1803.10122.

# Cosmos-1.0-Diffusion *World Foundation Model*






Niket Agarwal et al. (2025) "Cosmos World Foundation Model Platform for Physical AI". arXiv preprint arXiv:2501.03575

# Cosmos-1.0-Diffusion *World Foundation Model*




Niket Agarwal et al. (2025) "Cosmos World Foundation Model Platform for Physical AI".
arXiv preprint arXiv:2501.03575

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

> Reinventa el mundo <

# GRACIAS

*Victor Flores Benites*

TRANSFORMATEC