Aprendizaje profundo

Autocodificadores variacionales

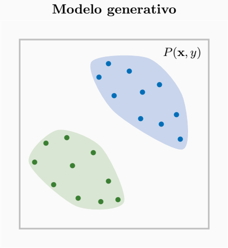
Gibran Fuentes-Pineda

Noviembre 2023

Introducción

- Entrenamiento discriminativo busca modelar directamente $P(y|\mathbf{x})$, por lo que generar muestras de la misma distribución puede ser bastante difícil.

- Redes generativas aprenden $P(\mathbf{x}, y)$ y pueden generar muestras de forma directa.



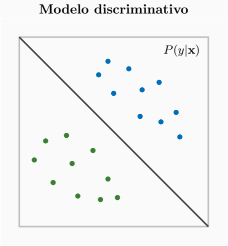


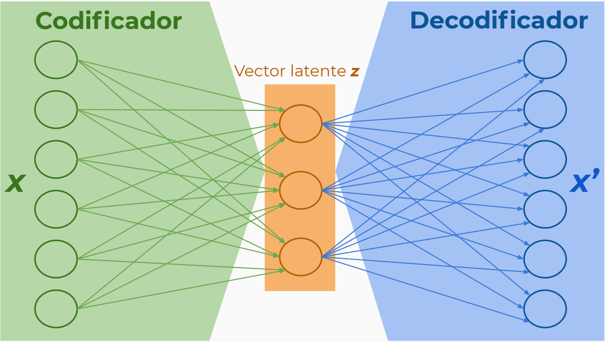
Imagen cortesía de Ricardo Montalvo Lezama

Modelos generativos profundos

- Modelos autoregresivos
- Flujos normalizadores
- Modelos basados energía (por ej. Máquinas de Boltzmann)
- Autocodificadores variacionales
- Redes generativas antagónicas
- Modelos de difusión

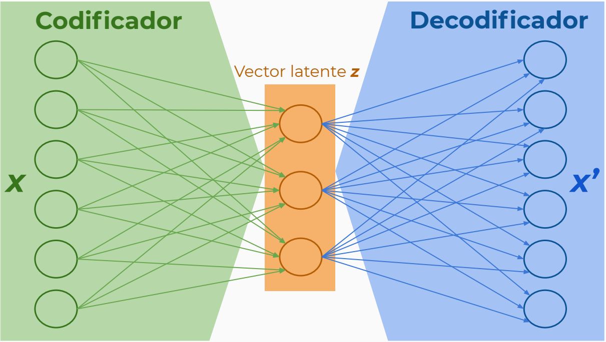
Esquema codificador–decodificador

- Red entrenada generando sus mismas entradas
  - Codificador: $\mathbf{z} = F_{\mathbf{W}_c,\mathbf{b}_c}(\mathbf{x})$
  - Decodificador: $\mathbf{x}' = G_{\mathbf{W}_d,\mathbf{b}_d}(\mathbf{z})$
  - $\mathbf{W}_c, \mathbf{b}_c, \mathbf{W}_d, \mathbf{b}_d = \arg\min_{\mathbf{W}_c,\mathbf{b}_c,\mathbf{W}_d,\mathbf{b}_d} \|\mathbf{x} - \mathbf{x}'\|_2^2$



Autocodificadores contractivos

- Hace **z** de menor dimensionalidad que **x**



Autocodificadores quita ruido

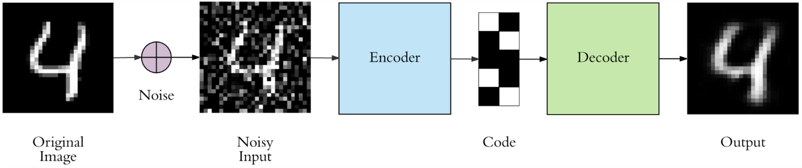


Imagen tomada de `https://towardsdatascience.com/applied-deep-learning-part-3-autoencoders-1c083af4d798`

Autocodificadores dispersos

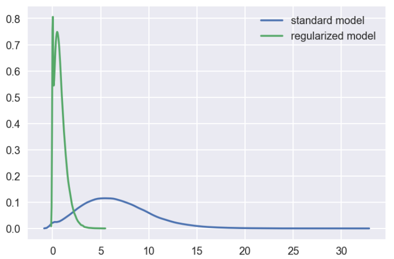


Figura tomada de `https://medium.com/towards-data-science/applied-deep-learning-part-3-autoencoders-1c083af4d798`

Autocodificadores como estrategia de inicialización

- Usado como estrategia de inicialización de pesos por capa (Restricted Boltzman Machines)

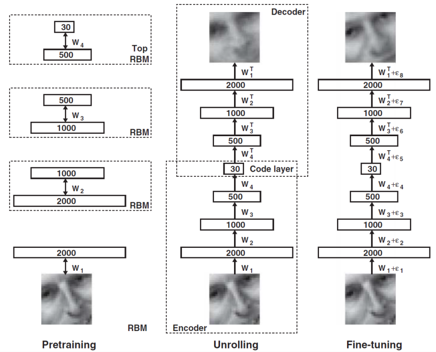


Imagen tomada de Hinton and Salakhutdinov. Reducing the Dimensionality of Data with Neural Networks, 2006.

Modelos de variables latentes profundas

- Sea $\mathbf{x}$ una observación muestreada aleatoriamente de una distribución no conocida, se presupone que

$$\mathbf{x} \sim P_{\boldsymbol{\theta}}(\mathbf{x}) \approx P_{real}(\mathbf{x}).$$

- Los modelos de variables latentes representan la distribución $P_{\boldsymbol{\theta}}(\mathbf{x})$ usando variables latentes $\mathbf{z}$

$$P_{\boldsymbol{\theta}}(\mathbf{x}) = \int P_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z}) d\mathbf{z}.$$

- La distribución conjunta comúnmente se factoriza como

$$P_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z}) = P_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}) \cdot P_{\boldsymbol{\theta}}(\mathbf{z}).$$

- En los modelos profundos de variables latentes (DLVM), estas distribuciones están parametrizadas por redes neuronales.

Proceso generativo de modelos de variables latentes

- Proceso generativo:
    1. Se obtiene una muestra $\mathbf{z}$ de la distribución a priori $\mathbf{z} \sim P_{\boldsymbol{\theta}}(\mathbf{z})$.
    2. Se obtiene una muestrea $\mathbf{x}$ de la distribución condicional $\mathbf{x} \sim P_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$.

- La probabilidad a posteriori está dada por

$$P_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}) = \frac{P_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})}{P_{\boldsymbol{\theta}}(\mathbf{x})}.$$

- Tanto $P_{\boldsymbol{\theta}}(\mathbf{x})$ como $P_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})$ son intratables en DLVMs.

Autocodificador variacional

- Un autocodificador variacional es un DLVM en el que se presupone que

$$P_{\boldsymbol{\theta}}(\mathsf{z}) = \mathcal{N}(0, \mathsf{I})$$
$$P_{\boldsymbol{\theta}}(\mathsf{x}|\mathsf{z}) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma}^2)$$

- Para realizar aprendizaje de forma eficiente, se aproxima $P_{\boldsymbol{\theta}}(\mathsf{z}|\mathsf{x})$ con $Q_{\boldsymbol{\phi}}(\mathsf{z}|\mathsf{x})$:

$$\boldsymbol{\mu}, \log \boldsymbol{\sigma} \leftarrow \text{RedNeuronalCodificadora}_{\boldsymbol{\phi}}(\mathsf{x})$$
$$Q_{\boldsymbol{\phi}}(\mathsf{z}|\mathsf{x}) \leftarrow \mathcal{N}(\mathsf{z}|\boldsymbol{\mu}, \operatorname{diag}(\boldsymbol{\sigma}))$$

- A esto se le conoce como inferencia amortizada.

Evidence Lower BOund (ELBO)

- En el entrenamiento se maximiza la cota inferior de evidencia:

$$
\begin{aligned}
\log P_{\boldsymbol{\theta}}(\mathrm{x}) &= \mathbb{E}_{Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x})}\left[\log P_{\boldsymbol{\theta}}(\mathrm{x})\right] \\
&= \mathbb{E}_{Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x})}\left[\log\left(\frac{P_{\boldsymbol{\theta}}(\mathrm{x},\mathrm{z})}{P_{\boldsymbol{\theta}}(\mathrm{z}|\mathrm{x})}\right)\right] = \mathbb{E}_{Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x})}\left[\log\left(\frac{P_{\boldsymbol{\theta}}(\mathrm{x},\mathrm{z})}{Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x})}\frac{Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x})}{P_{\boldsymbol{\theta}}(\mathrm{z}|\mathrm{x})}\right)\right] \\
&= \underbrace{\mathbb{E}_{Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x})}\left[\log\left(\frac{P_{\boldsymbol{\theta}}(\mathrm{x},\mathrm{z})}{Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x})}\right)\right]}_{\mathcal{L}_{\boldsymbol{\theta},\phi}(\mathrm{x})\ \text{(ELBO)}} + \underbrace{\mathbb{E}_{Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x})}\left[\log\left(\frac{Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x})}{P_{\boldsymbol{\theta}}(\mathrm{z}|\mathrm{x})}\right)\right]}_{D_{KL}\left[Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x})\|P_{\boldsymbol{\theta}}(\mathrm{z}|\mathrm{x})\right]\geq 0}
\end{aligned}
$$

- Debido a que la divergencia KL es no negativa

$$
\mathcal{L}_{\boldsymbol{\theta},\boldsymbol{\phi}}(\mathrm{x}) \leq \log P_{\boldsymbol{\theta}}(\mathrm{x})
$$

Entrenamiento de codificadores variacionales (1)

- Cota inferior de $\log P_{\boldsymbol{\theta}}(\mathrm{x})$ (aproxima máxima verosimilitud): $\boldsymbol{\phi}$ y $\boldsymbol{\theta}$ optimizadas conjuntamente
    - Equivalente a minimizar la divergencia de Kullback Leibler (KL) dada por

$$
\begin{aligned}
D_{KL}\left[Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x}) \| P_{\boldsymbol{\theta}}(\mathrm{z}|\mathrm{x})\right] &= \mathbb{E}_{Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x})}\left[\log \frac{Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x})}{P_{\boldsymbol{\theta}}(\mathrm{z}|\mathrm{x})}\right] \\
&= \sum_{\mathrm{x}} Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x}) \log \frac{Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x})}{P_{\boldsymbol{\theta}}(\mathrm{z}|\mathrm{x})} \\
&= \sum_{\mathrm{x}} Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x}) \left[\log Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x}) - \log P_{\boldsymbol{\theta}}(\mathrm{z}|\mathrm{x})\right] \\
&= \mathbb{E}_{Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x})}\left[\log Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x}) - \log P_{\boldsymbol{\theta}}(\mathrm{z}|\mathrm{x})\right]
\end{aligned}
$$

- Por regla de bayes

$$
\begin{aligned}
D_{KL}\left[Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x}) \| P_{\boldsymbol{\theta}}(\mathrm{z}|\mathrm{x})\right] = \mathbb{E}_{Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x})}\left[\log Q_{\boldsymbol{\phi}}(\mathrm{z}|\mathrm{x}) - \log P_{\boldsymbol{\theta}}(\mathrm{x}|\mathrm{z}) - \log P_{\boldsymbol{\phi}}(\mathrm{z})\right] \\
+ \log P_{\boldsymbol{\theta}}(\mathrm{x})
\end{aligned}
$$

Entrenamiento de codificadores variacionales (2)

- Despejando tenemos

$$\log P_{\boldsymbol{\theta}}(\mathsf{x}) - D_{KL}\left[Q_{\boldsymbol{\phi}}(\mathsf{z}|\mathsf{x}) \| P_{\boldsymbol{\theta}}(\mathsf{z}|\mathsf{x})\right] = \mathbb{E}_{Q_{\boldsymbol{\phi}}(\mathsf{z}|\mathsf{x})}\left[\log P_{\boldsymbol{\theta}}(\mathsf{x}|\mathsf{z})\right] - D_{KL}\left[Q_{\boldsymbol{\phi}}(\mathsf{z}|\mathsf{x}) \| P_{\boldsymbol{\theta}}(\mathsf{z})\right]$$

- Se busca maximizar $\mathbb{E}_{Q_{\boldsymbol{\phi}}(\mathsf{z}|\mathsf{x})}\left[\log P_{\boldsymbol{\theta}}(\mathsf{x}|\mathsf{z})\right]$ y minimizar $D_{KL}\left[Q_{\boldsymbol{\phi}}(\mathsf{z}|\mathsf{x}) \| P_{\boldsymbol{\theta}}(\mathsf{z})\right]$ mediante descenso por gradiente estocástico o variantes

Codificador-decodificador variacional

- **Red codificadora** genera media μ_{z} y covarianza diagonal Σ_{z} de $Q_\phi(\mathsf{z}|\mathsf{x})$
- **Red dedificadora** genera media μ_{x} y covarianza diagonal Σ_{x} de $P_\theta(\mathsf{x}|\mathsf{z})$
- $\mathsf{x}'^{(i)}$ se obtiene
    1. Muestreando $\mathsf{z}^{(i)}$ de $Q_\phi(\mathsf{z}|\mathsf{x})$
    2. Muestreando $\mathsf{x}'^{(i)}$ de $P_\theta(\mathsf{x}|\mathsf{z})$

Esquema general del modelo de autocodificador variacional

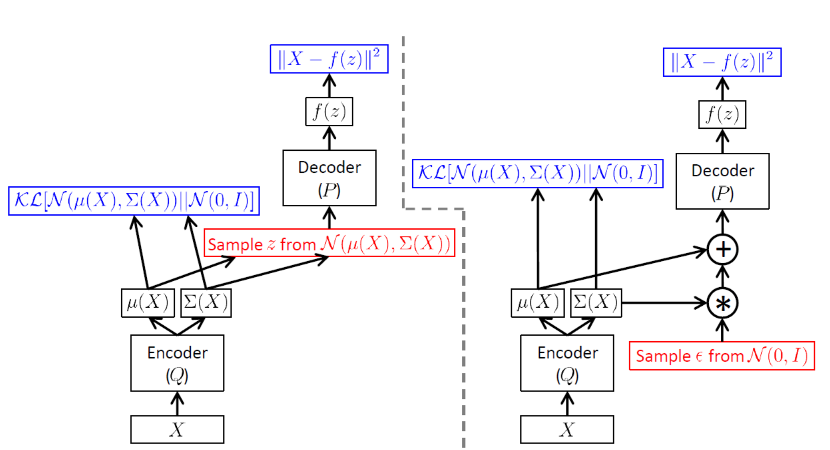


Imagen tomada de Carl Doersch. *Tutorial on Variational Autoencoders*, arXiv:1606.05908, 2016

Generación de datos

- Para generar nuevos datos se muestrea **z** de $\mathcal{N}(0, \mathbb{I})$

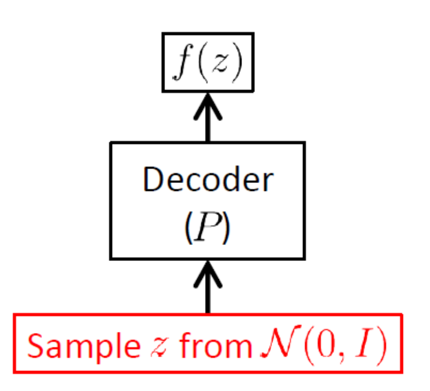


Imagen tomada de Carl Doersch. *Tutorial on Variational Autoencoders*, arXiv:1606.05908, 2016

Aplicaciones: generación de dígitos

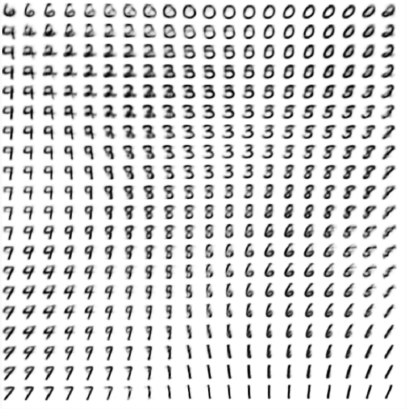


Imagen tomada de Kingma and Welling. Auto-Encoding Variational Bayes, 2013

Aplicaciones: generación de imágenes

Imagen tomada de Kingma and Welling. Auto-Encoding Variational Bayes, 2013

Aplicaciones: generación de imágenes a partir de texto

A stop sign is flying in blue skies. | A herd of elephants flying in the blue skies. | A toilet seat sits open in the grass field. | A person skiing on sand clad vast desert.

Imagen tomada de Mansimov et al. Generating Images from Captions with Attention, 2015

Aplicaciones: generación de dibujos (1)

- sketch-rnn

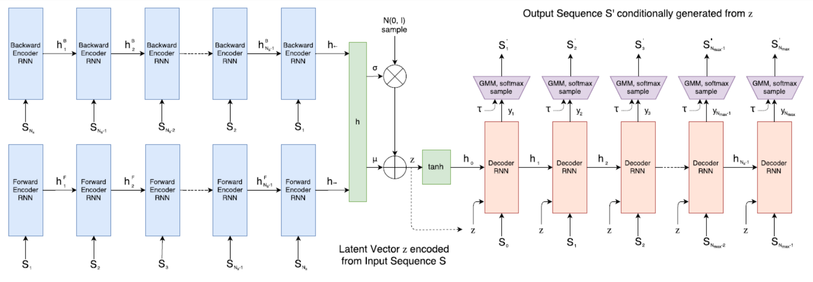


Imagen tomada de Ha and Eck. A Neural Representation of Sketch Drawings, 2017.

Aplicaciones: generación de dibujos (2)

- sketch-rnn

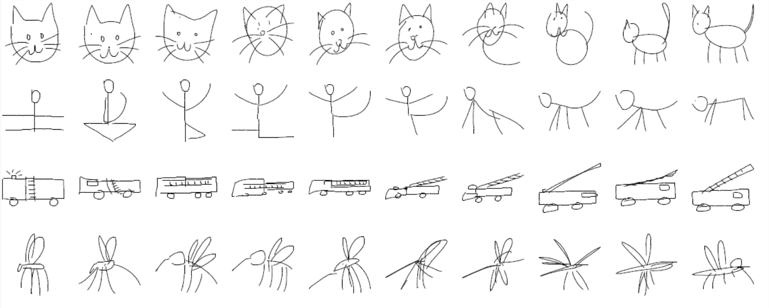


Imagen tomada de Ha and Eck. A Neural Representation of Sketch Drawings, 2017.

Colapso del aposteriori

- Ocurre cuando la distribución variacional se aproxima a la distribución a priori no informativa para un conjunto de variables latentes
    - La generación es independiente del vector latente

- Estrategias de mitigación
    - Escalar pérdida KL por una tasa $\beta \in (0, 1)$ y programarla respecto a épocas
    - Usar apriori y aposteriori que no sean gaussianas (por ej. von Mises-Fisher)
    - Agregar conexiones de salto
    - Mejorar el entrenamiento de la red codificadora
    - Reemplazar ELBO (por ej. con autocodificadores antagónicos)