

# Sistemas de Recomendación y Autoencoders Variacionales

---

Erika Rivadeneira Pérez

*erika.rivadeneira@cimat.mx*

**Asesor:** Johan Lode Van Horebeek

*horebeek@cimat.mx*

Febrero 17, 2022

CIMAT A.C.

# Contenido

1. Motivación
2. Objetivos
3. Avances
4. Cronograma de actividades
5. Referencias Principales







# Motivación

---

# Motivación

**What to watch** [Browse our What to Watch page >](#)














**Top picks >**  
TV shows and movies just for you

 ★ 8.5 <b>The Pianist</b> <a href="#">Watch options</a>	 ★ 8.5 <b>City Lights</b> <a href="#">Watch options</a>	 ★ 8.1 <b>The Circus</b> <a href="#">Watch options</a>	 ★ 7.9 <b>Toy Story 2</b> <a href="#">Watch options</a>	 ★ 8.0 <b>Monsters, Inc.</b> <a href="#">Watch options</a>	 ★ 7.9 <b>Before Midnight</b> <a href="#">Watch options</a>
---	---	--	---	---	---

[explore](#)  **movielens**

**top picks** [see more](#)

MovieLens recommends these movies

<b>Interstellar</b> 2014 <b>[PG-13]</b> 169 min   ★★★★★	<b>Toy Story 3</b> 2010 <b>[G]</b> 103 min   ★★★★★	<b>The Lives of Others</b> 2006 <b>[R]</b> 137 min   ★★★★★	<b>Despicable Me</b> 2010 <b>[PG]</b> 95 min   ★★★★★	<b>Equilibrium</b> 2002 <b>[R]</b> 107 min   ★★★★★	<b>Tangled</b> 2010 <b>[PG]</b> 100 min   ★★★★★	<b>Fear</b> 1998  ★★★★★
--	---	---	---	--	--	---

# Motivación



En la actualidad tenemos que hacer elecciones sobre un conjunto de opciones muy grande.

- Youtube:
  - 5B videos (vistos todos los días)
  - 2B usuarios
- Amazon:
  - Productos 3B (en 11 mercados)
  - 200 millones de usuarios (por mes)

# Motivación


✦ En la actualidad tenemos que hacer elecciones sobre un conjunto de opciones muy grande.

- Youtube:
  - 5B videos (vistos todos los días)
  - 2B usuarios
- Amazon:
  - Productos 3B (en 11 mercados)
  - 200 millones de usuarios (por mes)

✦ ¡Sistemas de recomendación al rescate!

- Predice la calificación o preferencia que un usuario le daría a un artículo.
- Proporciona a los usuarios sugerencias de elementos que les sean de utilidad.

- ✦ Se cuenta una matriz de calificaciones  $\mathbb{X} = [\mathbb{X}_{u,i}]$  donde  $u \in \{1, \dots, U\}$  indexa a los usuarios y  $i \in \{1, \dots, I\}$  indexa a los ítems de la base de datos:




Usuario\ítem	ítem 1	ítem 2	ítem 3	ítem 4
Usuario 1	3		1	
Usuario 2		3	4	
Usuario 3	2			5
Usuario 4	2	5		1

Usuario\ítem	ítem 1	ítem 2	ítem 3	ítem 4
Usuario 1	1	0	1	0
Usuario 2	0	1	1	0
Usuario 3	1	0	0	1
Usuario 4	1	1	0	1

miro

Tenemos vectores de calificaciones (por usuario) y el problema se acota a aquellas entradas con valores en cada vector.

- ✦ Se cuenta una matriz de calificaciones  $\mathbb{X} = [\mathbb{X}_{u,i}]$  donde  $u \in \{1, \dots, U\}$  indexa a los usuarios y  $i \in \{1, \dots, I\}$  indexa a los ítems de la base de datos:



Usuario\ítem	ítem 1	ítem 2	ítem 3	ítem 4
Usuario 1	3		1	
Usuario 2		3	4	
Usuario 3	2			5
Usuario 4	2	5		1

Usuario\ítem	ítem 1	ítem 2	ítem 3	ítem 4
Usuario 1	1	0	1	0
Usuario 2	0	1	1	0
Usuario 3	1	0	0	1
Usuario 4	1	1	0	1

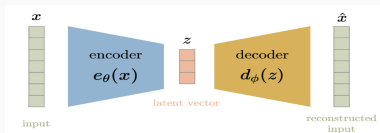
miro

Tenemos vectores de calificaciones (por usuario) y el problema se acota a aquellas entradas con valores en cada vector.

- ✦ Están involucrados muchos desafíos:
- Precisión de las recomendaciones.
  - Escalabilidad (cantidad de usuarios).
  - Explicabilidad.
  - Los intereses evolucionan con el tiempo.
  - Sesgo en datos históricos.
  - Datos incompletos.

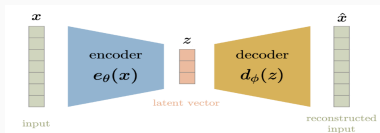


# Autoencoders



$$\text{loss} = \sum_{u,i} (x_{u,i} - \hat{x}_{u,i})^2 = \sum_u \|x_u - d_{\phi}(e_{\theta}(x_u))\|_2^2 = \sum_u \|x_u - d_{\phi}(z)\|_2^2$$

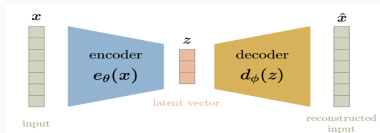
# Autoencoders



$$\text{loss} = \sum_{u,i} (x_{u,i} - \hat{x}_{u,i})^2 = \sum_u \|x_u - d_{\phi}(e_{\theta}(x_u))\|_2^2 = \sum_u \|x_u - d_{\phi}(z)\|_2^2$$

✧ Modelo de reducción de dimensión.

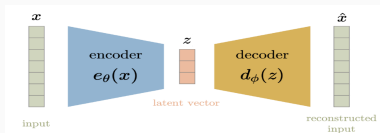
# Autoencoders



$$\text{loss} = \sum_{u,i} (x_{u,i} - \hat{x}_{u,i})^2 = \sum_u \|x_u - d_{\phi}(e_{\theta}(x_u))\|_2^2 = \sum_u \|x_u - d_{\phi}(z)\|_2^2$$

- ✧ Modelo de reducción de dimensión.
- ✧ Introducción de variables latentes  $z$ .

# Autoencoders

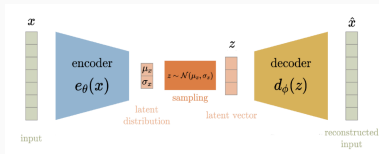


$$\text{loss} = \sum_{u,i} (x_{u,i} - \hat{x}_{u,i})^2 = \sum_u \|x_u - d_{\phi}(e_{\theta}(x_u))\|_2^2 = \sum_u \|x_u - d_{\phi}(z)\|_2^2$$

- ✦ Modelo de reducción de dimensión.
- ✦ Introducción de variables latentes  $Z$ .
- ✦ Aproxima  $X$  a  $\hat{X}$ .

# Autoencoders Variacionales (VAE)

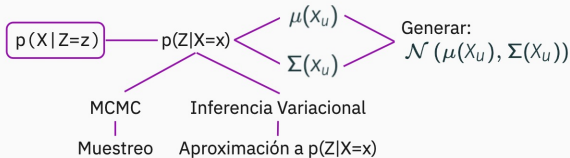
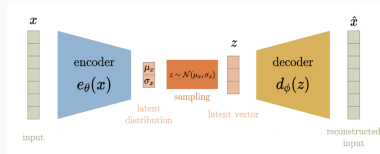
El proceso generativo de un VAE propuesto por [Liang et al. \(2018\)](#) se desglosa de la siguiente manera:



$$\boxed{p(X|Z=z)} \rightarrow p(Z|X=x) \begin{cases} \mu(x_u) \\ \Sigma(x_u) \end{cases} \rightarrow \text{Generar: } \mathcal{N}(\mu(x_u), \Sigma(x_u))$$

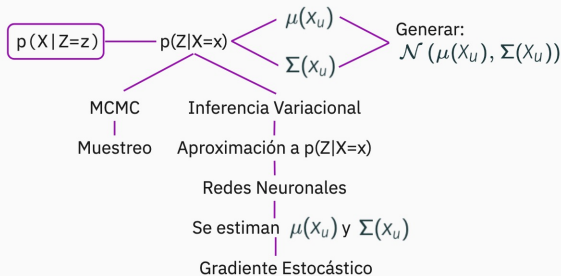
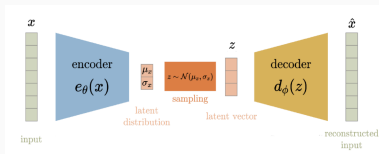
# Autoencoders Variacionales (VAE)

El proceso generativo de un VAE propuesto por Liang et al. (2018) se desglosa de la siguiente manera:



# Autoencoders Variacionales (VAE)

El proceso generativo de un VAE propuesto por [Liang et al. \(2018\)](#) se desglosa de la siguiente manera:



$$\log p(X; \theta) \geq \mathbb{E}_{q_{\theta}(Z|X)}[\log p(X | Z)] - \text{KL}(\mathcal{N}(\mu_{\theta}(X_u), \Sigma_{\theta}(X_u)) \parallel \mathcal{N}(0, I))$$

# Objetivos

---

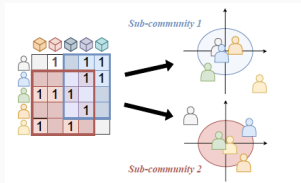




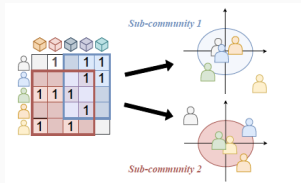
Realizar revisión y discusión sobre VAEs para sistemas de recomendación.

- ✦ Realizar revisión y discusión sobre VAEs para sistemas de recomendación.
- ✦ Aplicar y experimentar los diferentes métodos considerados.

- ✦ Realizar revisión y discusión sobre VAEs para sistemas de recomendación.
- ✦ Aplicar y experimentar los diferentes métodos considerados.
- ✦ Incorporar un componente de localidad, inspirado por [Choi et al. \(2021\)](#).



- ✦ Realizar revisión y discusión sobre VAEs para sistemas de recomendación.
- ✦ Aplicar y experimentar los diferentes métodos considerados.
- ✦ Incorporar un componente de localidad, inspirado por [Choi et al. \(2021\)](#).



# Avances

---

- ✦ Revisión y escritura de el estado del arte, [Choi et al. \(2021\)](#), [Liang et al \(2018\)](#) y [Dilokthanakul et al. \(2016\)](#), sobre VAEs en sistemas de recomendación, autoencoders colaborativos locales, VAEs con mezclas de Gaussianas, respectivamente.

- ✦ Revisión y escritura de el estado del arte, [Choi et al. \(2021\)](#), [Liang et al \(2018\)](#) y [Dilokthanakul et al. \(2016\)](#), sobre VAEs en sistemas de recomendación, autoencoders colaborativos locales, VAEs con mezclas de Gaussianas, respectivamente.
- ✦ Adaptación de códigos de MultVAE, DAE, CDAE, MF, EASE, LOCA-VAE, LOCA-EASE implementados por [Choi et al. \(2021\)](#) y [Liang et al. \(2018\)](#) a librerías de pycharm. Se han replicado los resultados de los autores y mediante experimentación se ha encontrado que:
  - ✦ La precisión de los métodos locales con datasets de mayor dimensión aumenta, a diferencia de los modelos globales (su precisión casi no difiere).
  - ✦ Los métodos son estables.

- ✦ Revisión y escritura de el estado del arte, Choi et al. (2021), Liang et al (2018) y Dilokthanakul et al. (2016), sobre VAEs en sistemas de recomendación, autoencoders colaborativos locales, VAEs con mezclas de Gaussianas, respectivamente.
- ✦ Adaptación de códigos de MultVAE, DAE, CDAE, MF, EASE, LOCA-VAE, LOCA-EASE implementados por Choi et al. (2021) y Liang et al. (2018) a librerías de pycharm. Se han replicado los resultados de los autores y mediante experimentación se ha encontrado que:
  - ✦ La precisión de los métodos locales con datasets de mayor dimensión aumenta, a diferencia de los modelos globales (su precisión casi no difiere).
  - ✦ Los métodos son estables.
- ✦ Exploración de la idea de introducir mezclas de Gaussianas como distribuciones a priori de los datos, para incorporar el componente local. Modificando así, la divergencia  $KL(f(x)||g(x))$ , con

$$f(x) = \sum_{i=1}^N \pi_i \mathcal{N}(\mu_i, \theta_i), \quad y \quad g(x) = \sum_{k=1}^M \pi_k \mathcal{N}(\mu_k, \theta_k),$$

$N$  y  $M$  son el número de componentes de cada mezcla,  $\pi$  son las a priori de cada mezcla.



- ✦ Revisión y escritura de el estado del arte, [Choi et al. \(2021\)](#), [Liang et al \(2018\)](#) y [Dilokthanakul et al. \(2016\)](#), sobre VAEs en sistemas de recomendación, autoencoders colaborativos locales, VAEs con mezclas de Gaussianas, respectivamente.
- ✦ Adaptación de códigos de MultVAE, DAE, CDAE, MF, EASE, LOCA-VAE, LOCA-EASE implementados por [Choi et al. \(2021\)](#) y [Liang et al. \(2018\)](#) a librerías de pycharm. Se han replicado los resultados de los autores y mediante experimentación se ha encontrado que:
  - ✦ La precisión de los métodos locales con datasets de mayor dimensión aumenta, a diferencia de los modelos globales (su precisión casi no difiere).
  - ✦ Los métodos son estables.
- ✦ Exploración de la idea de introducir mezclas de Gaussianas como distribuciones a priori de los datos, para incorporar el componente local. Modificando así, la divergencia  $KL(f(x)||g(x))$ , con

$$f(x) = \sum_{i=1}^N \pi_i \mathcal{N}(\mu_i, \theta_i), \quad y \quad g(x) = \sum_{k=1}^M \pi_k \mathcal{N}(\mu_k, \theta_k),$$

$N$  y  $M$  son el número de componentes de cada mezcla,  $\pi$  son las a priori de cada mezcla.

- ✦ Revisión de propuesta para entrenamiento de la función objetivo con mezclas de Gaussianas en los VAE introducido por [Nat Dilokthanakul et al \(2016\)](#).

# Cronograma de actividades

---

- ✦ Continuación de la revisión del estado del arte (Febrero-Julio).
- ✦ Exploración y adaptación de la idea de localidad (Febrero-Marzo).
- ✦ Implementación del método local (Abril-Mayo).
- ✦ Revisión y corrección del proyecto (julio-agosto).

# Referencias Principales

---

- ✧ Dawen Liang et al. Variational Autoencoders for Collaborative Filtering. 2018. arXiv: 1802 . 05814 [stat.ML].
- ✧ Minjin Choi et al. “Local Collaborative Autoencoders”. In: Proceedings of the 14th ACM International Conference on Web Search and Data Mining. 2021, pp. 734–742.
- ✧ Nat Dilokthanakul et al. “Deep unsupervised clustering with gaussian mixture variational autoencoders”. In: arXiv preprint arXiv:1611.02648 (2016).
- ✧ Hareesh Bahuleyan et al. “Probabilistic Natural Language Generation with Wasserstein Autoencoders”. In: CoRR abs/1806.08462 (2018). arXiv: 1806.08462. URL: <http://arxiv.org/abs/1806.08462>.