



UC | Chile



UC | Chile

Otros métodos latentes de factorización matricial de ratings



Denis Parra Santander

— Profesor Asociado DCC UC



Una de las grandes lecciones del premio Netflix fue la noción de dimensiones latentes para representar usuarios e ítems, así como la introducción de FunkSVD y otros métodos de factores latentes



Contenidos



Tema 1

**Métodos de
factores latentes**

Tema 2

**Relación de
factores latentes
y factorización
matricial**

Tema 3

**FunkSVD como
método de
factores latentes**

Tema 4

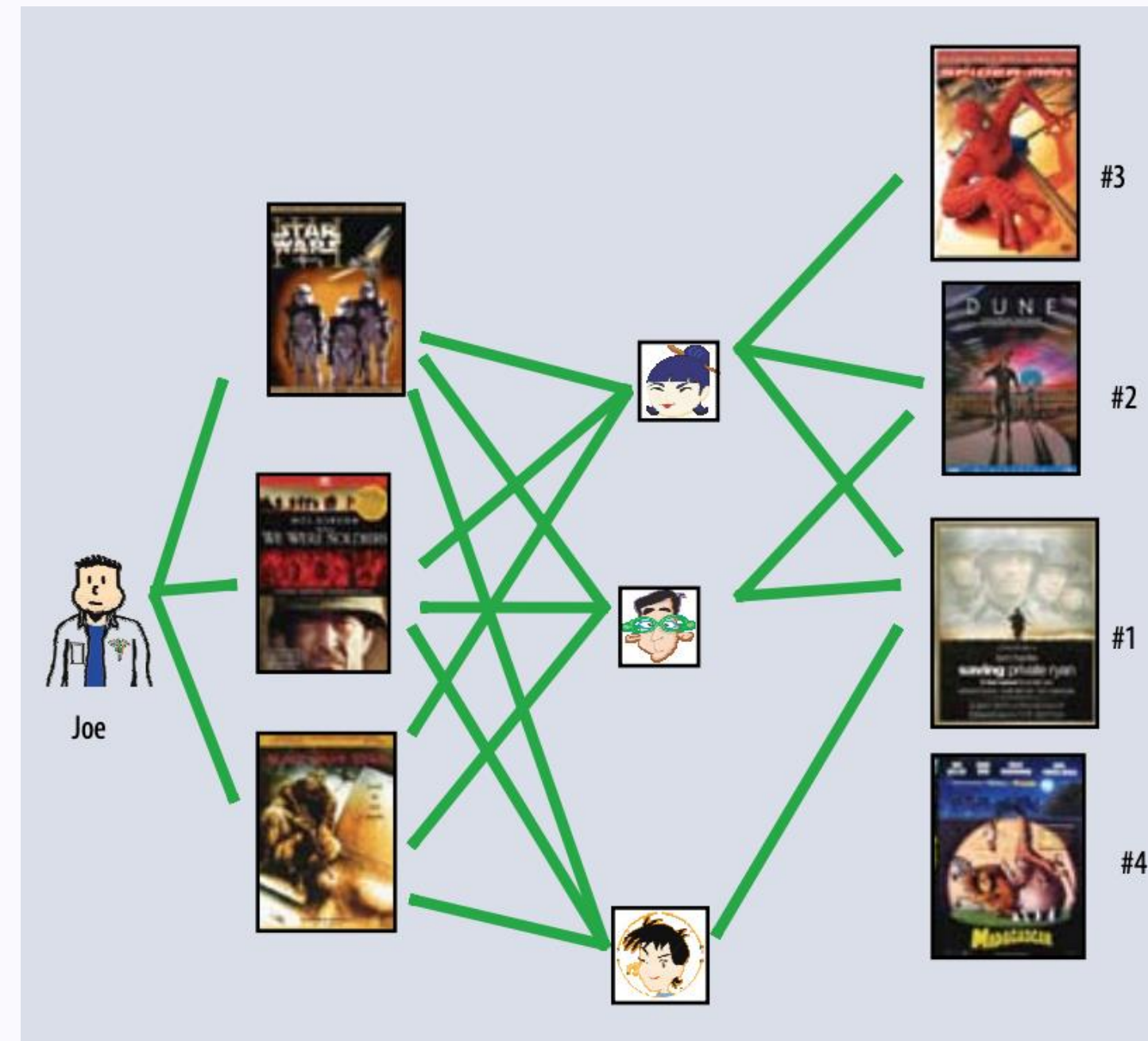
**Otros métodos
de factores
latentes**

Tema 1

Métodos de factores latentes

Hasta ahora: Filtrado colaborativo

- ¿qué entienden hasta ahora por filtrado colaborativo?
- ¿cuál es su principal fortaleza?
- ¿qué versiones hay y qué diferencias tienen?
- ¿por qué se les denomina también métodos *KNN?
- ¿cuál es su principal debilidad desde un punto de vista computacional?



Relación filtrado colaborativo – métodos latentes

Filtrado Colaborativo

Recomendar lo que usuarios similares han consumido

$$\hat{r}_{u,i} = \bar{r}_u + \frac{\sum_{j \in V(u)} \text{sim}(u, j)(r_{j,i} - \bar{r}_j)}{\sum_{j \in V(u)} \text{sim}(u, j)}$$

Métodos latentes

Genera vectores latentes por cada usuario e ítem del set de datos

$$\hat{r}_{ui} = \vec{q}_i \cdot \vec{p}_u$$

donde

\vec{q}_i : vector latente para el ítem i

\vec{p}_u : vector latente para el usuario u

Representación de usuarios e items









- ¿cómo se podría definir el espacio latente?

Representación de usuarios e ítems

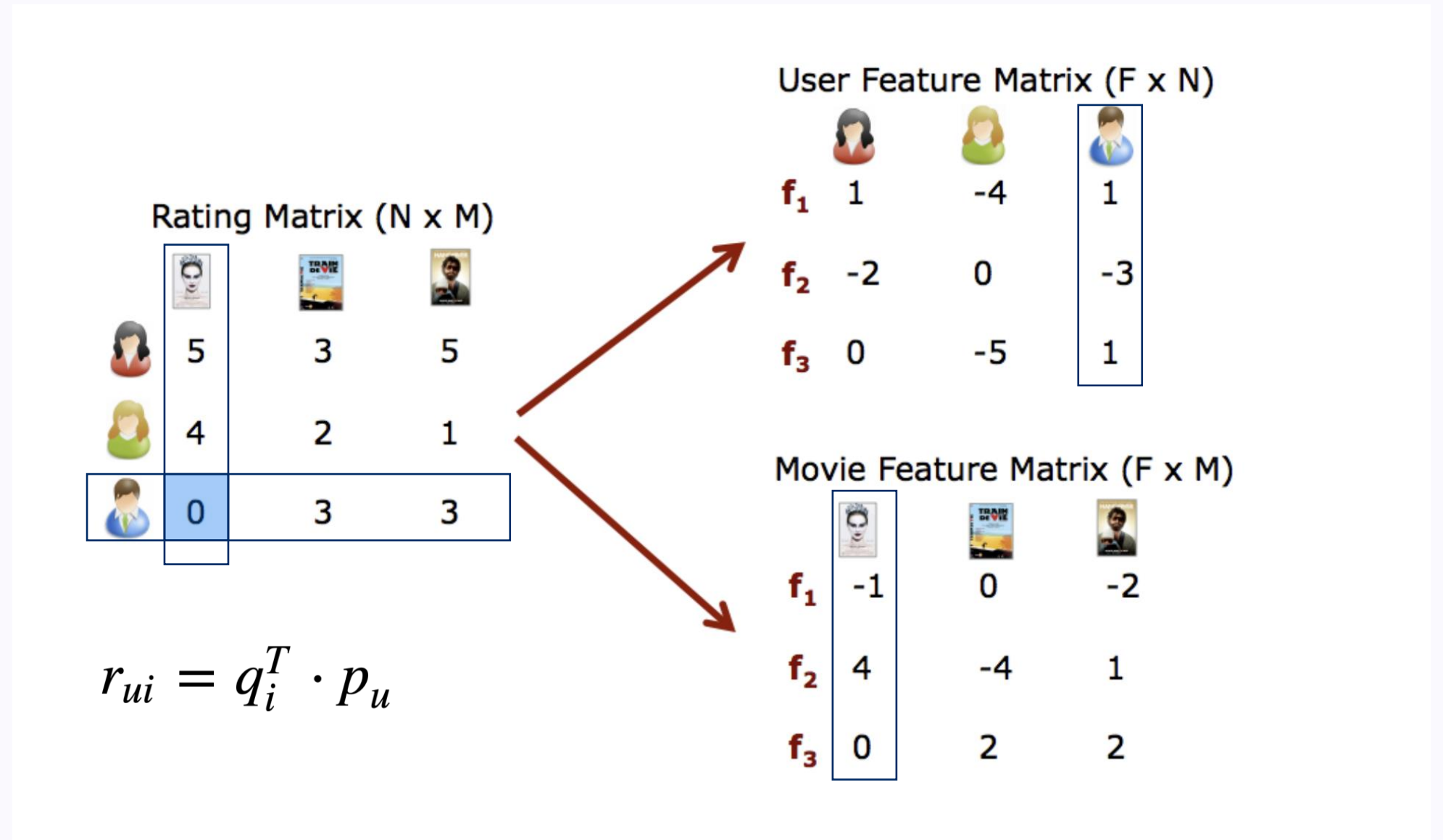
- ¿Cómo se podría definir el espacio latente?
- La representación de usuarios en la matriz de usuario-ítem ¿es latente?

Rating Matrix (N x M)

			
	5	3	5
	4	2	1
	0	3	3

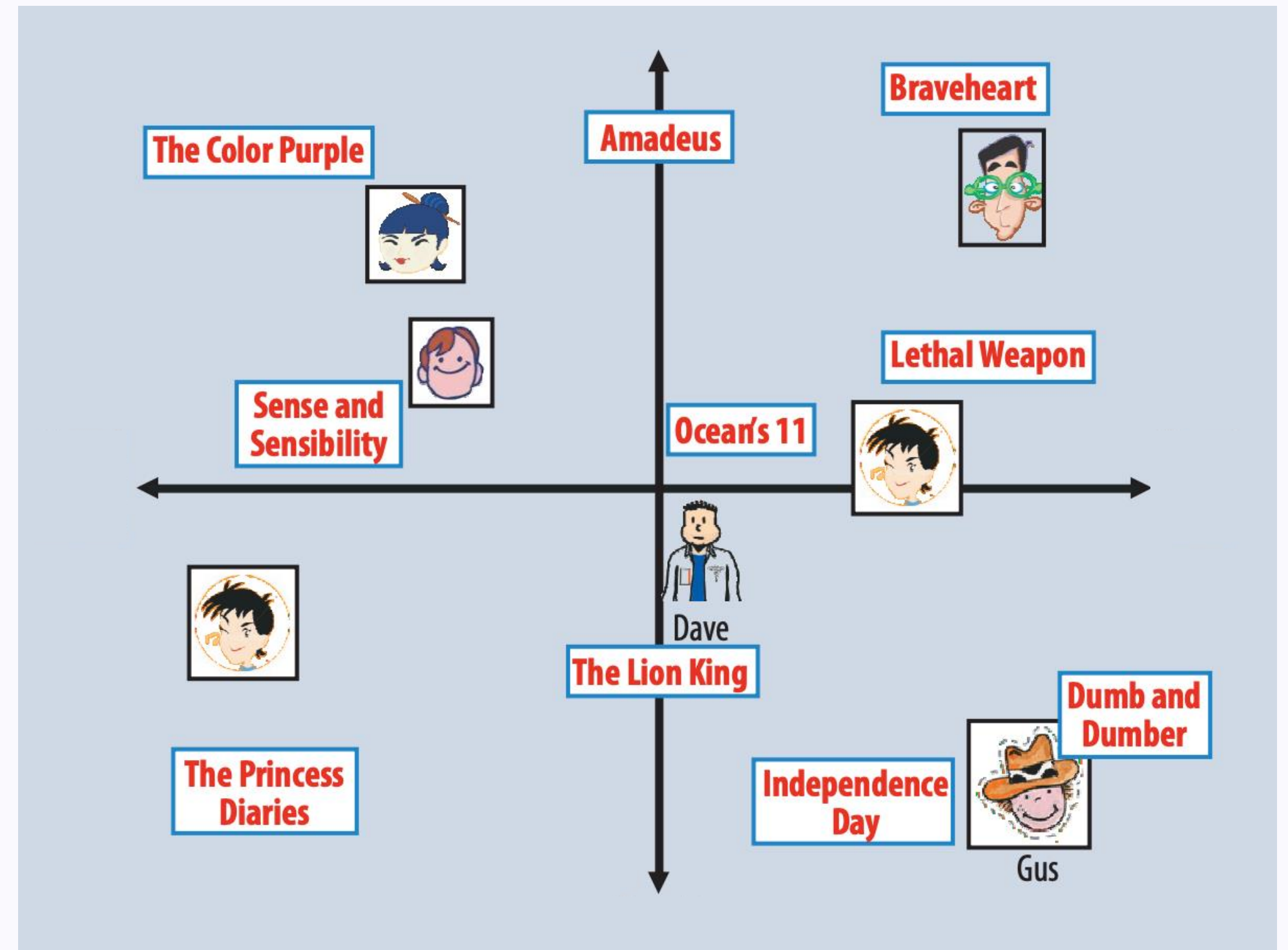
Representación de usuarios e ítems

- ¿cómo se podría definir el espacio latente?
- ¿qué diferencia tiene con la representación tradicional de usuarios e ítems con la representación en el espacio latente?



La matriz usuario-item y el espacio latente

- En la figura a continuación, ¿por qué se denomina como un ejemplo de representación en el espacio latente?
- ¿cómo podríamos obtener automáticamente la representación de espacio latente?



Tema 2

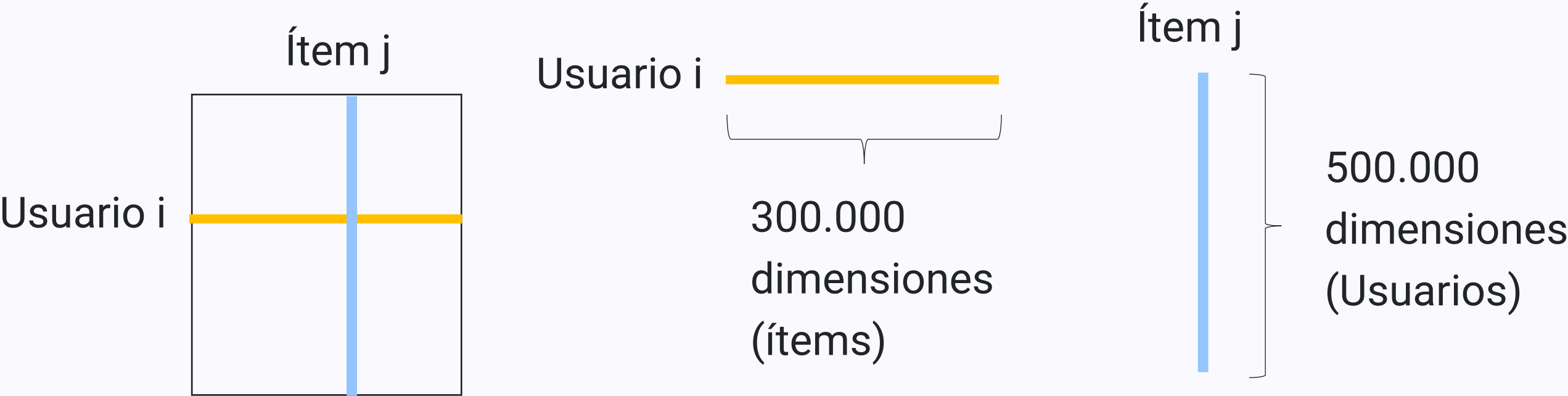
Relación de factores latentes y factorización matricial

Costo computacional de recomendación KNN

- Es muy costoso recomendar con métodos KNN
- La representación “no latente” tiene altísima dimensionalidad

Ejemplo
M usuarios, M = 500.000
N ítems, N = 300.000

15 * 10¹⁰ celdas! (150 miles de millones de celdas)



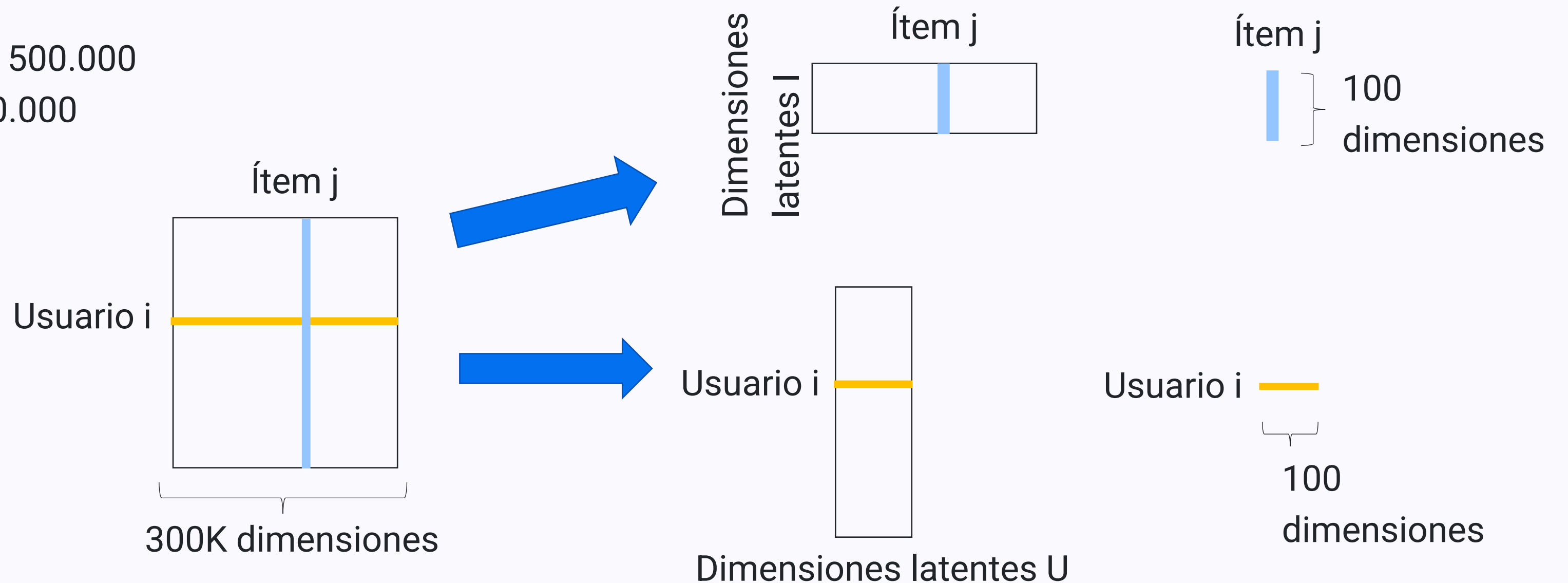
Reducción de dimensionalidad

- Se reduce el costo computacional de calcular similitud y la maldición de la dimensionalidad.

Ejemplo:

M usuarios, $M = 500.000$

N ítems, $N = 300.000$

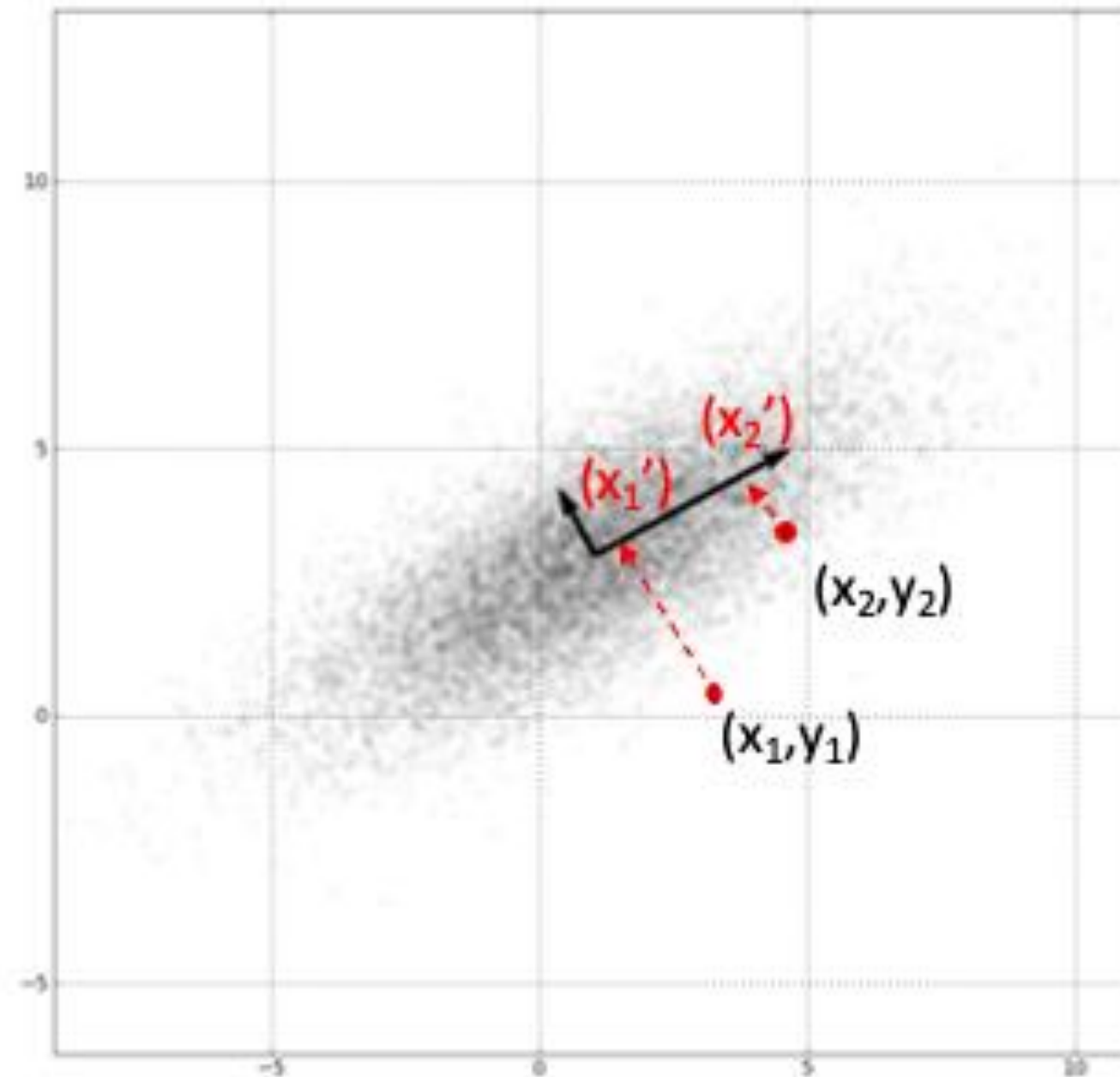


¿Qué técnicas de reducción de dimensionalidad conocen?



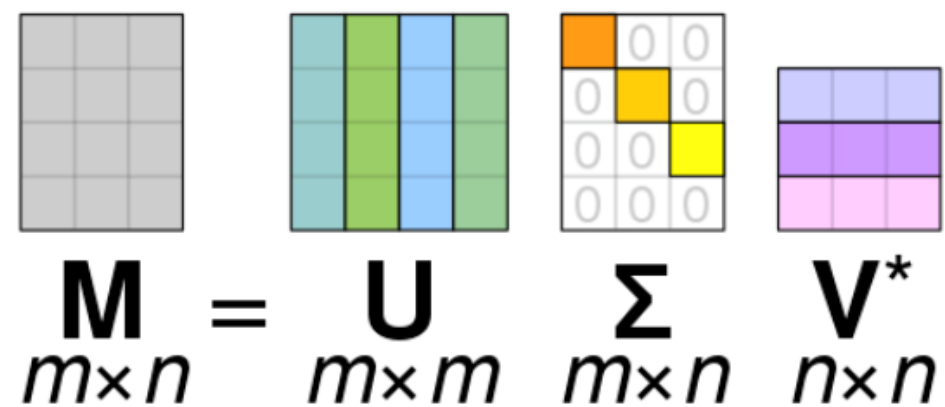
¿qué técnicas de reducción de dimensionalidad conocen?

- La técnica PCA (Principal Component Analysis) permite reducir dimensionalidad



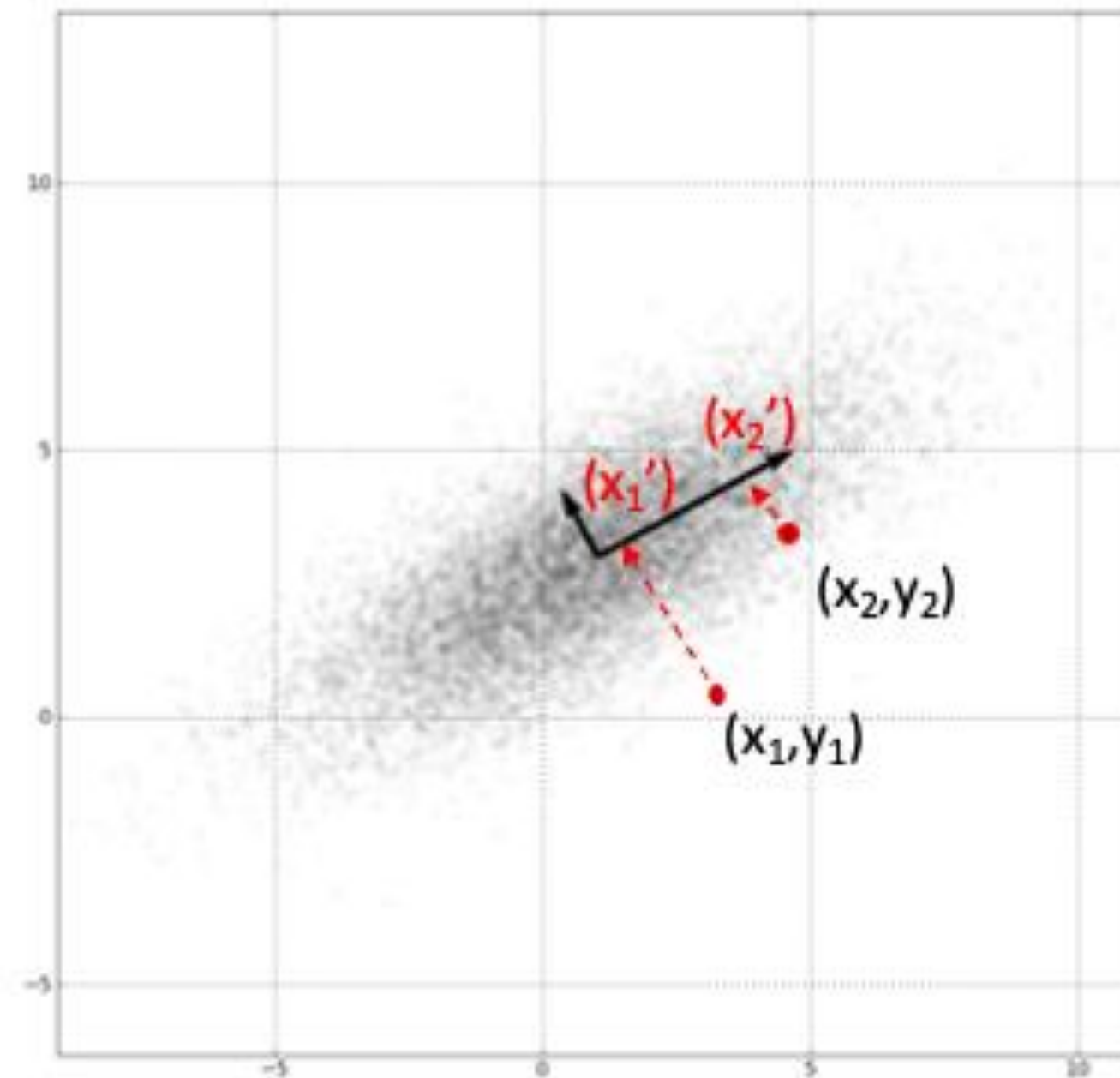
¿Cómo puedo obtener los vectores solución en PCA?

- La técnica PCA (Principal Component Analysis) permite reducir dimensionalidad
- Los valores de “principal componentes” se pueden obtener vía SVD, una técnica de factorización matricial

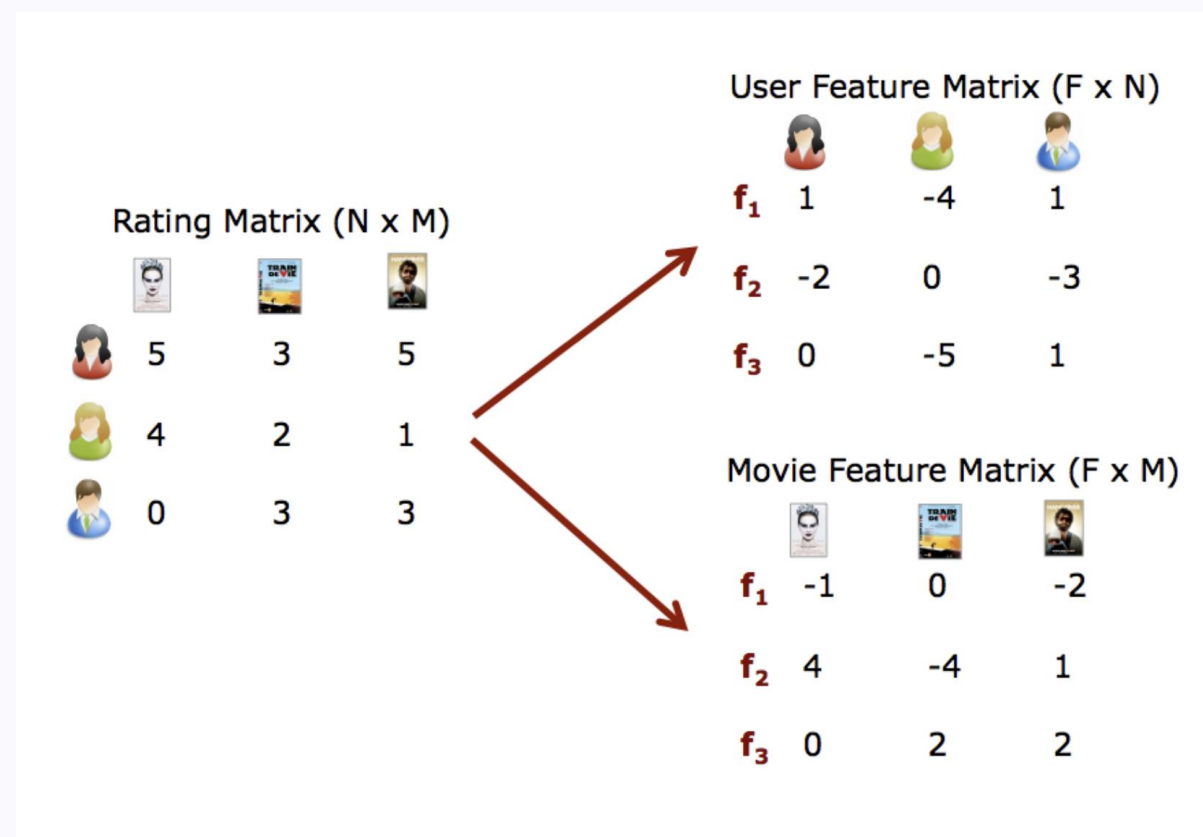


The diagram shows the SVD factorization of a matrix M into three matrices: U , Σ , and V^* . Matrix M is a 4x4 grid. Matrix U is a 4x4 grid with columns colored teal, green, blue, and green. Matrix Σ is a 4x4 grid with a diagonal of colored squares (orange, yellow, yellow, yellow) and zeros elsewhere. Matrix V^* is a 4x4 grid with rows colored blue, purple, purple, and pink.

$$\begin{matrix} \text{4x4 grid} \\ \mathbf{M} \\ m \times n \end{matrix} = \begin{matrix} \text{4x4 grid with colored columns} \\ \mathbf{U} \\ m \times m \end{matrix} \begin{matrix} \text{4x4 grid with colored diagonal and zeros} \\ \mathbf{\Sigma} \\ m \times n \end{matrix} \begin{matrix} \text{4x4 grid with colored rows} \\ \mathbf{V}^* \\ n \times n \end{matrix}$$



Representación latente vía factorización matricial



Item

	W	X	Y	Z
A		4.5	2.0	
B	4.0		3.5	
C		5.0		2.0
D		3.5	4.0	1.0

Rating Matrix

=

	W	X
A	1.2	0.8
B	1.4	0.9
C	1.5	1.0
D	1.2	0.8

User Matrix

X

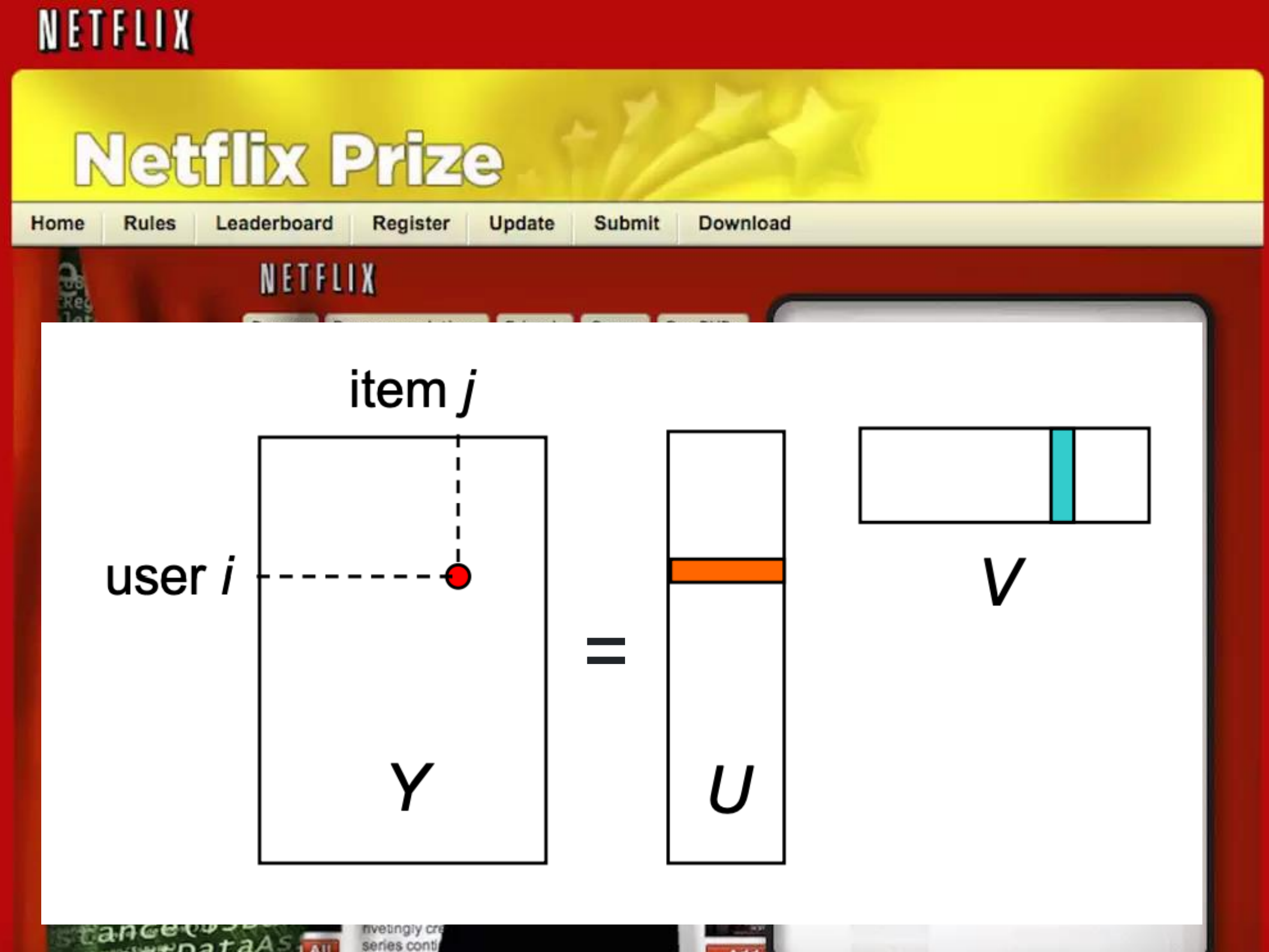
	W	X	Y	Z
	1.5	1.2	1.0	0.8
	1.7	0.6	1.1	0.4

Item Matrix

Tema 3

FunkSVD como método de factores latentes

Obtener
representación
latente de usuarios
e ítems vía
factorización
matricial
= Netflix Prize



**FunkSVD fue
introducido por
Simon Funk durante
el Netflix Prize, en
un blog**

NETFLIX

Monday, December 11, 2006

Netflix Update: Try This at Home

Home



<https://sifter.org/simon/journal/20061211.html>

¿Por qué necesitamos de un SVD “especial”?



- PCA está basado en SVD (factorización matricial)
- ¿Por qué simplemente no se usa SVD?

Matriz dispersa

Simon Funk indica que Netflix distribuyó un dataset con 500 mil usuarios (filas) sobre 17 mil películas (columnas), lo que indica una matriz de 8,5 mil millones de celdas, pero solo 100 millones de esas celdas tienen calificaciones o ratings. Es decir, ¡solo un 1,17% de las celdas tiene valores!

	Item 1	Item 2	...	Item n
Usuario 1	1	5	?	4
Usuario 2	5	1	?	?
...	?	?	?	?
Usuario m	2	5	?	?

m: 500K usuarios
n: 17K ítems
1,17% de celdas con ratings

Función de pérdida de FunkSVD

La ecuación que resume el modelo de factorización matricial FunkSVD, que nos permite encontrar vectores en el espacio latente tanto para usuarios como para ítems, es la siguiente:

$$\min_{p^*, q^*} \sum_{(u,i) \in K} (r_{ui} - q_i^T \cdot p_u)^2 + \lambda(||q_i||^2 + ||p_u||^2)$$

donde

r_{ui} : rating que asignó el usuario u al ítem i

q_i : vector latente del ítem i

p_u : vector latente del usuario u

λ : constante de regularización

Aprendizaje: Descenso de Gradiente

- Descenso de gradiente es un procedimiento para optimizar funciones
- La intuición es mover los parámetros del modelo en dirección contraria del gradiente
- En el caso de *FunkSVD* la función de aprendizaje se aplica para cada par (u, i)

Función objetivo

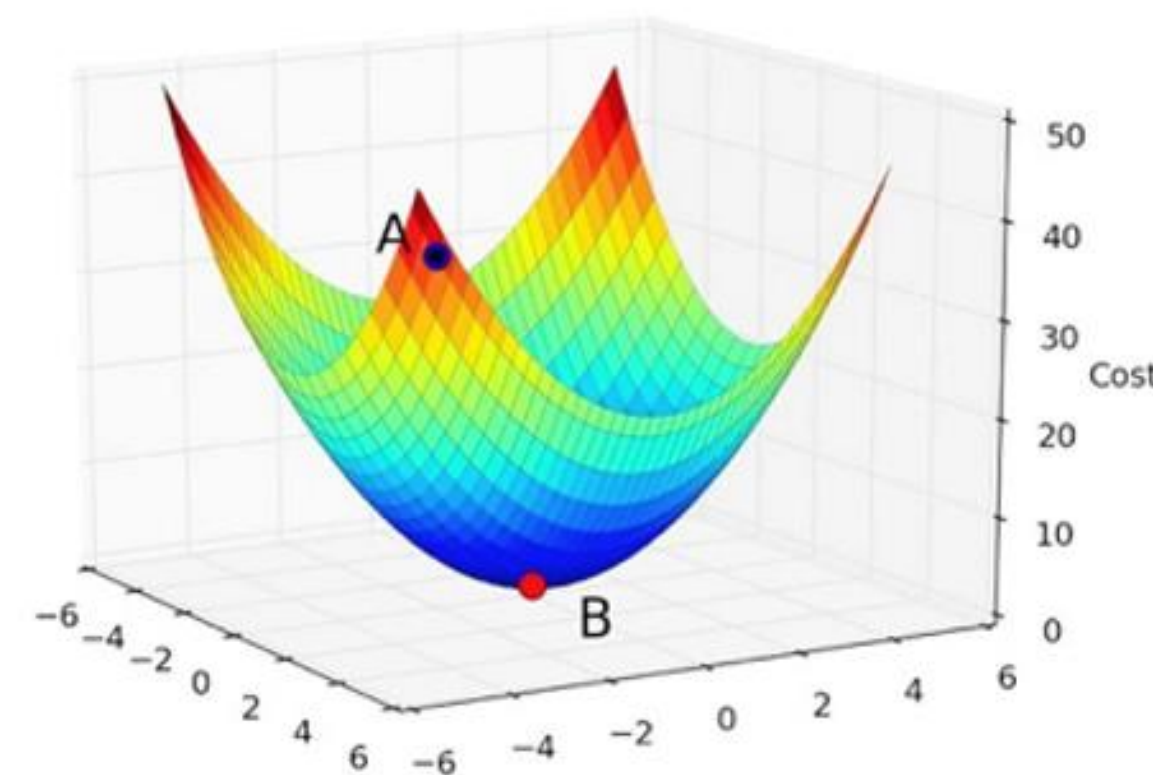
$$\min_{\theta} \frac{1}{2} \sum_i (y_i - f(x_i, \theta))^2$$

Función de error

$$E_i = \frac{1}{2} (y_i - f(x_i, \theta))^2$$

Actualización de parámetros

$$\Delta w_i \leftarrow w_i - \eta \frac{\partial E}{\partial w_i}$$



Tema 4

Otros métodos de factores latentes

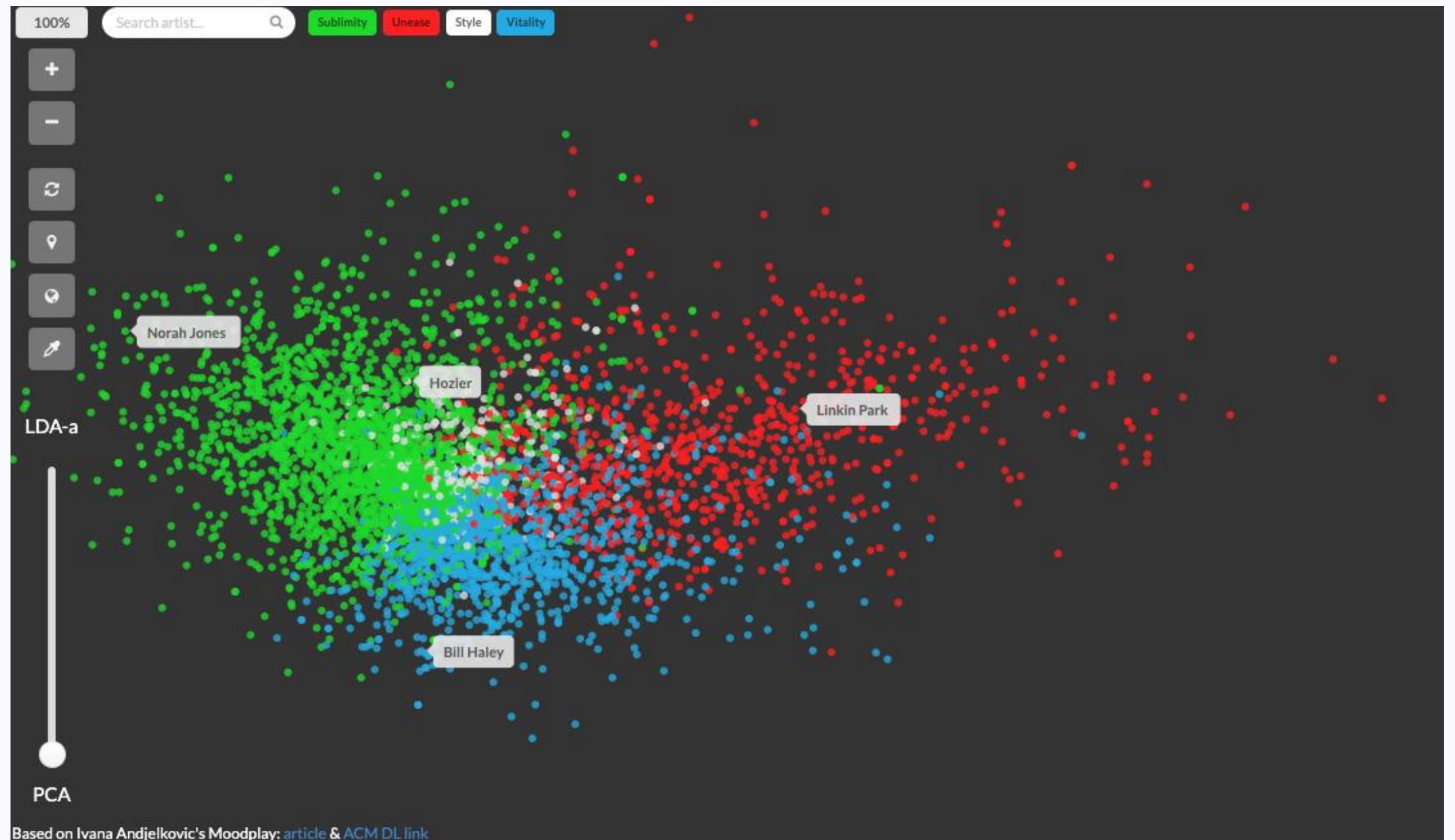
Otras técnicas para encontrar factores latentes



- ¿Por qué se necesitarían otras técnicas para encontrar factores latentes?

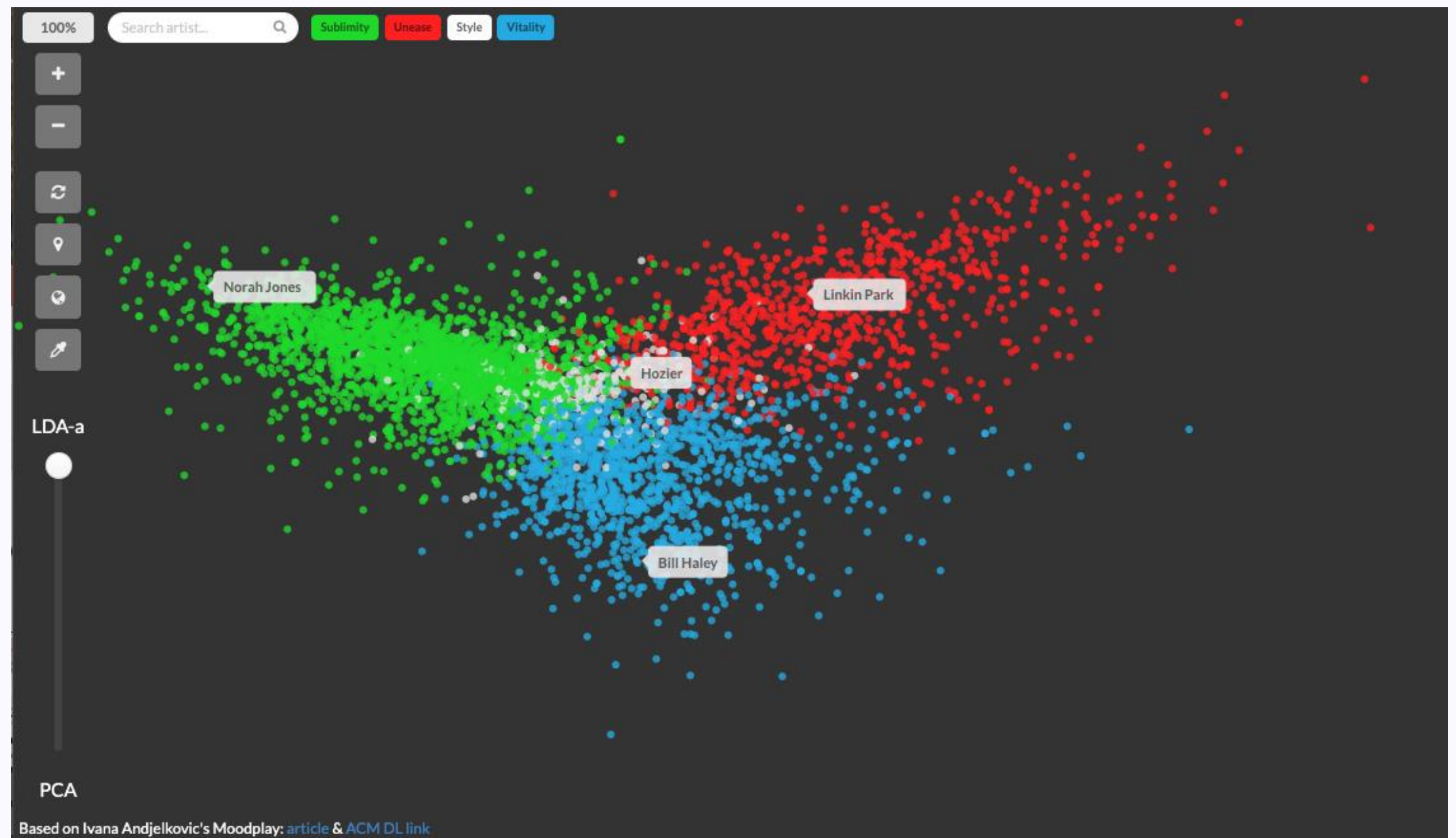
Otras técnicas para encontrar factores latentes

- ¿Por qué se necesitarían otras técnicas para encontrar factores latentes?
- <http://moodplay.pythonanywhere.com/visualize/lda-a/pca/>



Otras técnicas para encontrar factores latentes

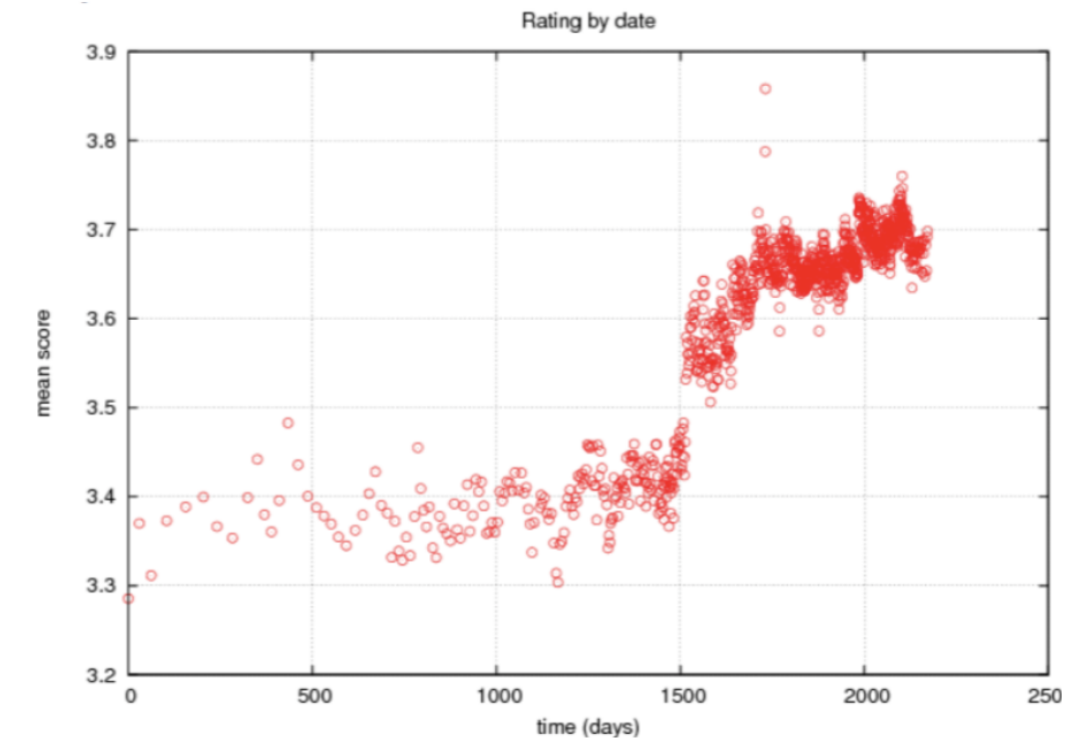
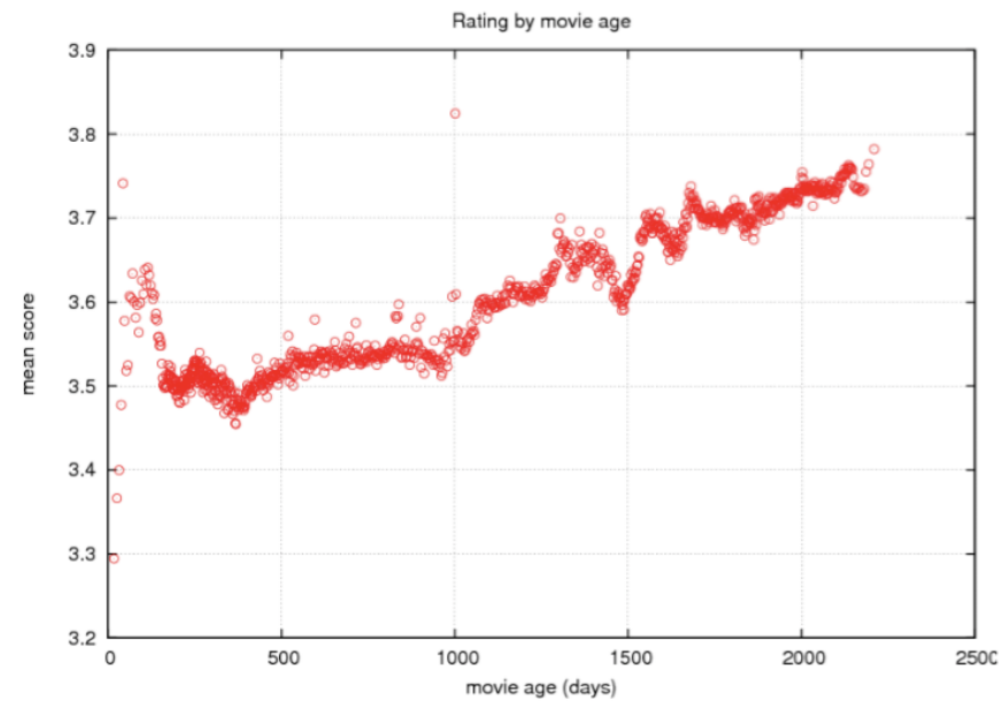
- ¿Por qué se necesitarían otras técnicas para encontrar factores latentes?
- <http://moodplay.pythonanywhere.com/visualize/lda-a/pca/>



Otras técnicas para encontrar factores latentes

- **timeSVD** : permite agregar variables adicionales, en este modelo información temporal

$$\hat{r}_{ui}(t) = \mu + b_i(t) + b_u(t) + q_i^T p_u(t)$$



Otras técnicas para encontrar factores latentes

- **SVD++** : permite agregar información de los “vecinos” al espacio latente de un usuario en particular

$$\hat{r}_{ui} = \mu + b_i + b_u + q_i^T \left[p_u + |N(u)|^{-0.5} \sum_{i \in N(u)} x_i + \sum_{a \in A(u)} y_a \right]$$

Otras técnicas para encontrar factores latentes

- **ALS:** Permite incorporar actividad implícita de los usuarios en el sistema (no solo ratings o calificaciones explícitas)

Binarización de los datos

$$p_{ui} = \begin{cases} 1 & \text{si } r_{ui} > 0 \\ 0 & \text{si } r_{ui} = 0 \end{cases}$$

Función de confianza

$$c_{ui} = 1 + \alpha r_{ui}$$

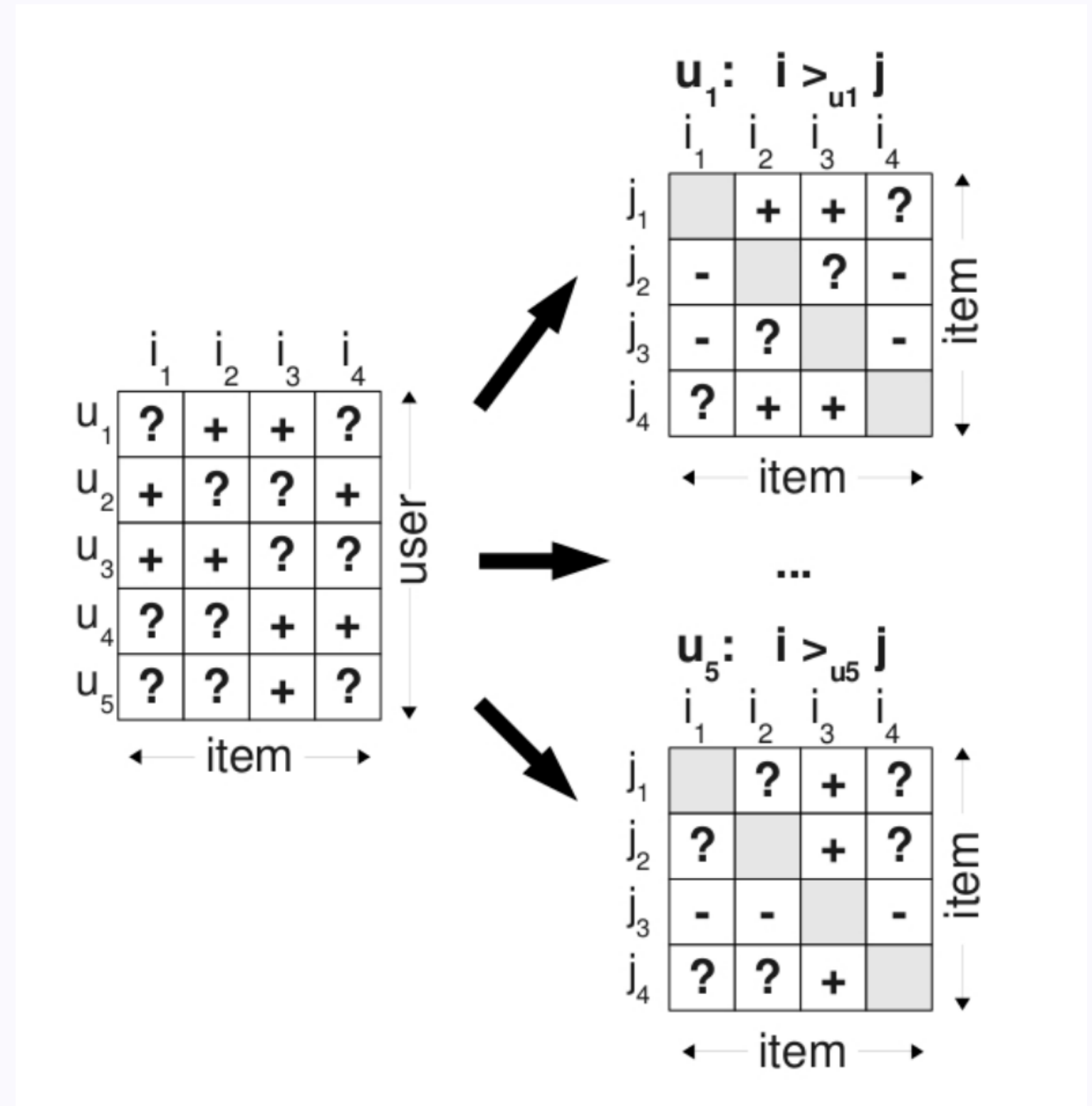
Función de pérdida

$$\min_{x^*, y^*} \sum_{u, i} c_{ui} (p_{ui} - x_u^T y_i)^2 + \lambda (||x_u||^2 + ||y_i||^2)$$

Otras técnicas para encontrar factores latentes

- **BPR:** Permite entrenar el modelo de forma contrastiva, considerando por cada usuario pares de ítems en lugar de un sólo ítem a la vez

$$\mathcal{D}_p = \{(u, i, j) \in \mathcal{D} | i \in \mathcal{I}_u \wedge j \in \mathcal{I} \setminus \mathcal{I}_u\}$$



Otras técnicas para encontrar factores latentes

- Máquina de factorización:
Modelo que permite
generalizar muchos modelos
de factorización matricial y
además agregar atributos
discretos y continuos
adicionales

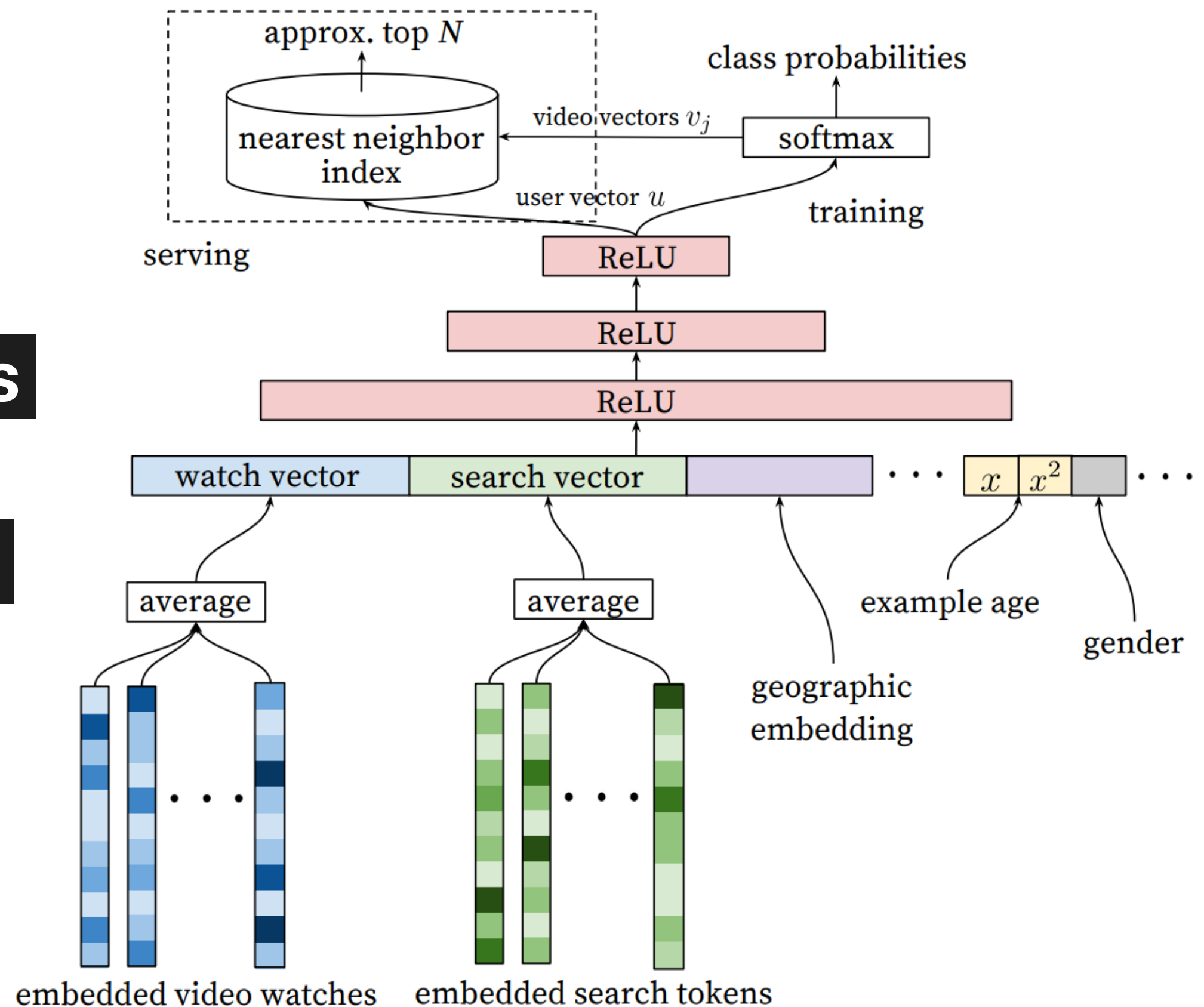
C. Dos variables categóricas y tiempo como predictor continuo

Feature vector \mathbf{x}											
$\mathbf{x}^{(1)}$	1	0	0	...	1	0	0	0	...	0.2	
$\mathbf{x}^{(2)}$	1	0	0	...	0	1	0	0	...	0.6	
$\mathbf{x}^{(3)}$	1	0	0	...	0	0	1	0	...	0.61	
$\mathbf{x}^{(4)}$	0	1	0	...	0	0	1	0	...	0.3	
$\mathbf{x}^{(5)}$	0	1	0	...	0	0	0	1	...	0.5	
$\mathbf{x}^{(6)}$	0	0	1	...	1	0	0	0	...	0.1	
$\mathbf{x}^{(7)}$	0	0	1	...	0	0	1	0	...	0.8	
	A	B	C	...	TI	NH	SW	ST	...		
	User				Movie						Time

- Modelo corresponde a:

$$\hat{y}(\mathbf{x}) := w_0 + w_i + w_u + t w_{\text{time}} + \langle \mathbf{v}_u, \mathbf{v}_i \rangle + t \langle \mathbf{v}_u, \mathbf{v}_{\text{time}} \rangle + t \langle \mathbf{v}_i, \mathbf{v}_{\text{time}} \rangle$$

Otra técnica usada más recientemente para obtener representación latente de usuarios e ítems son las redes neuronales profundas



Covington, P., Adams, J., & Sargin, E. (2016). Deep neural networks for YouTube recommendations. In *Proceedings of the 10th ACM conference on recommender systems* (pp. 191-198).



UC | Chile