ACÁMICA

# ¡Bienvenidas/os a Data Science!

# Agenda

Repaso Reducción de Dimensionalidad

Explicación: Sistemas de recomendación

Break

Explicación: Sistemas de recomendación

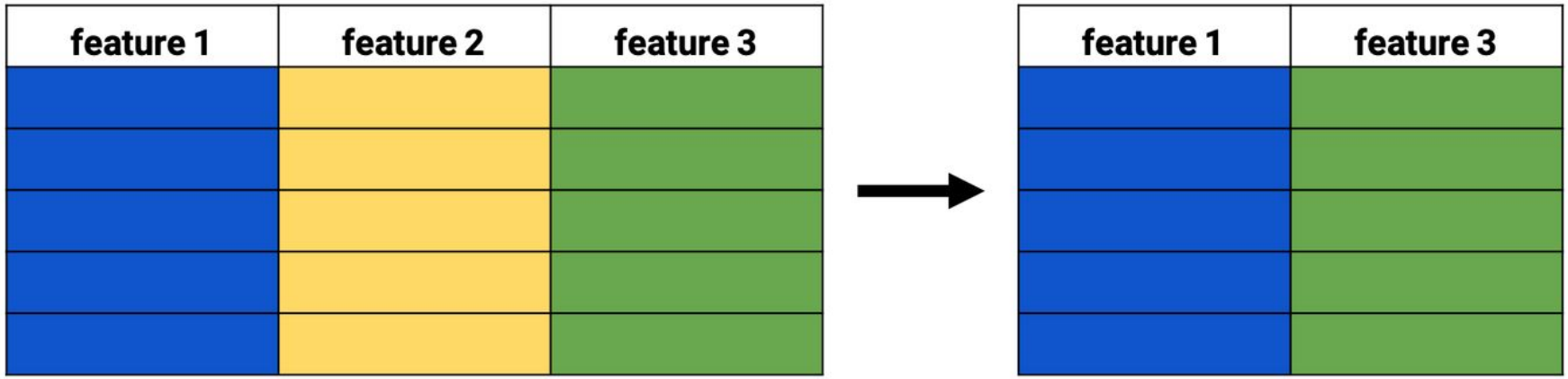
Hands-on training

Entrega 06

Cierre

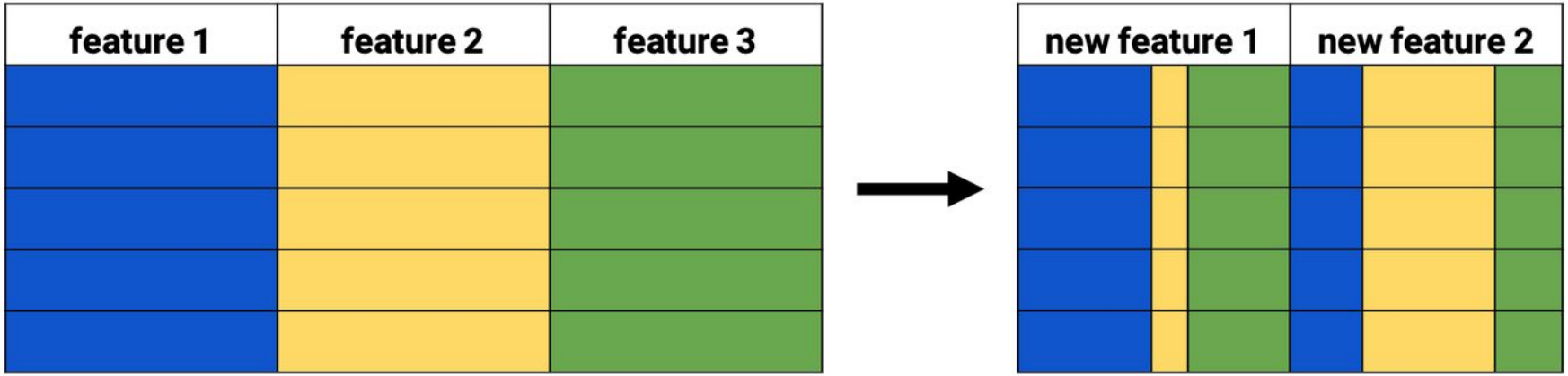
# Reducción de la dimensionalidad

Esto que hicimos, también es conocido como **FEATURE SELECTION**




Elegimos, bajo cierto criterio, las features que van a formar parte del dataset “final”.

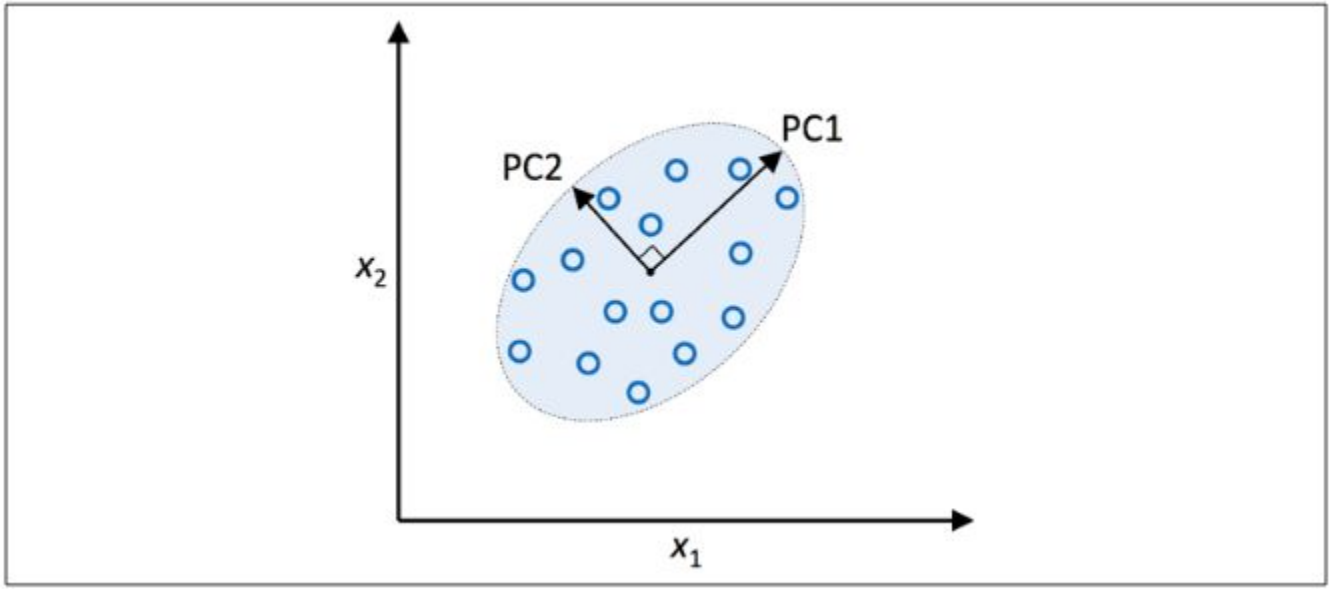
Otra forma de reducir la dimensionalidad es haciendo **FEATURE EXTRACTION**. Es un enfoque muy distinto al anterior, pero que busca cumplir el mismo objetivo.




"Extraemos" nuevas features a partir de las originales. Estas nuevas features tienen la menor redundancia de información posible, por lo tanto, son menos cantidad.

**¿Se les ocurre una desventaja al trabajar las features de esta manera?**

# PCA

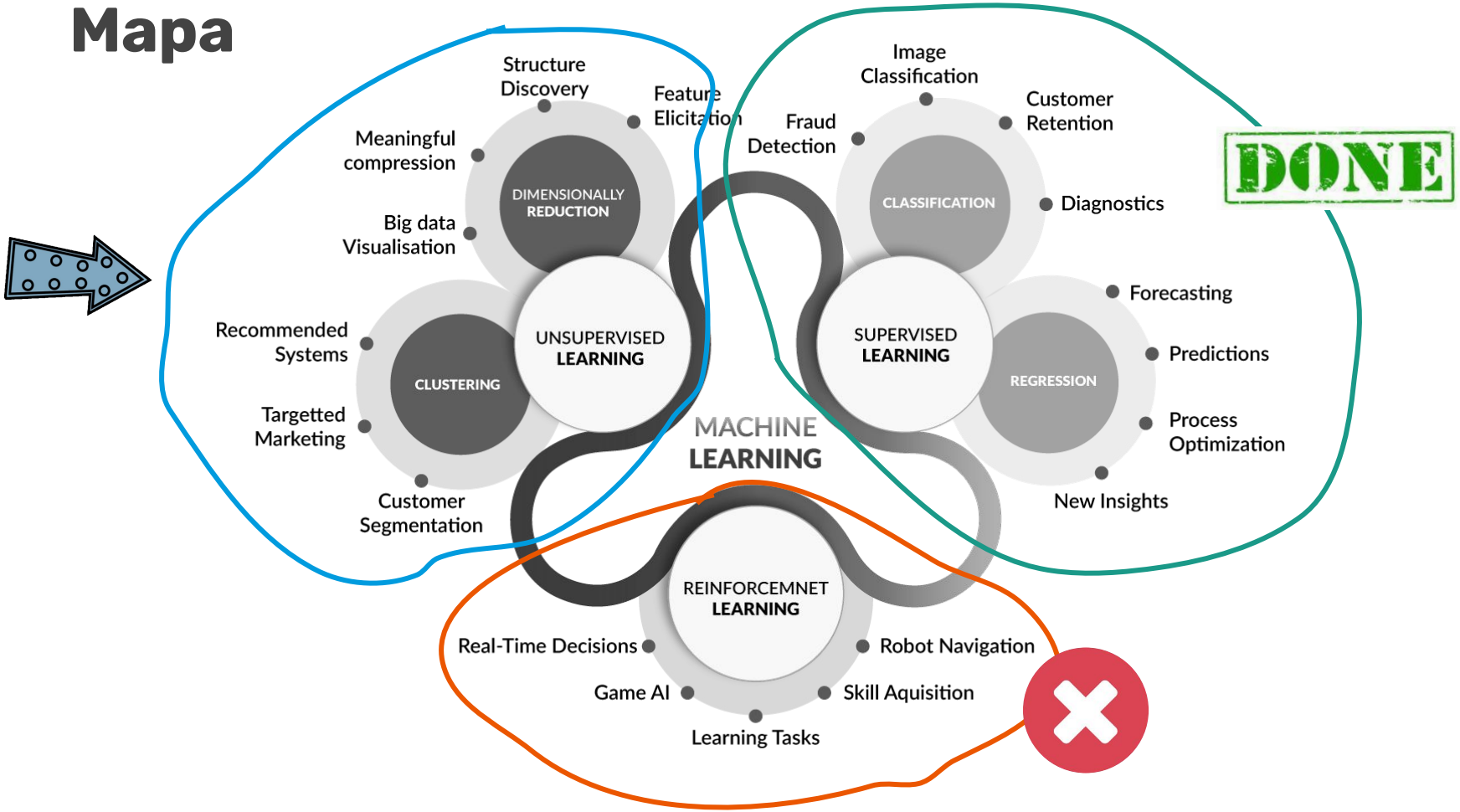


**IMPORTANTE**

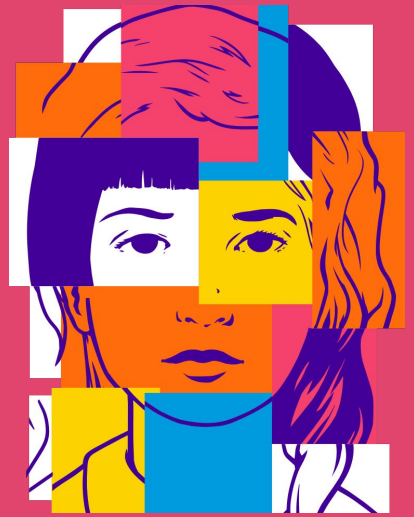
Bajar el siguiente dataset:

**https://www.kaggle.com/netflix-inc/netflix-prize-data**

# Mapa


Structure Discovery
Feature Elicitation
Meaningful compression
Big data Visualisation
DIMENSIONALLY REDUCTION
UNSUPERVISED LEARNING
Recommended Systems
Targetted Marketing
Customer Segmentation
CLUSTERING
Image Classification
Customer Retention
Fraud Detection
Diagnostics
CLASSIFICATION
SUPERVISED LEARNING
Forecasting
Predictions
Process Optimization
New Insights
REGRESSION
MACHINE LEARNING
REINFORCEMNET LEARNING
Real-Time Decisions
Robot Navigation
Game AI
Skill Aquisition
Learning Tasks
DONE

# Sistemas de recomendación



# Hoy nos inspiramos con

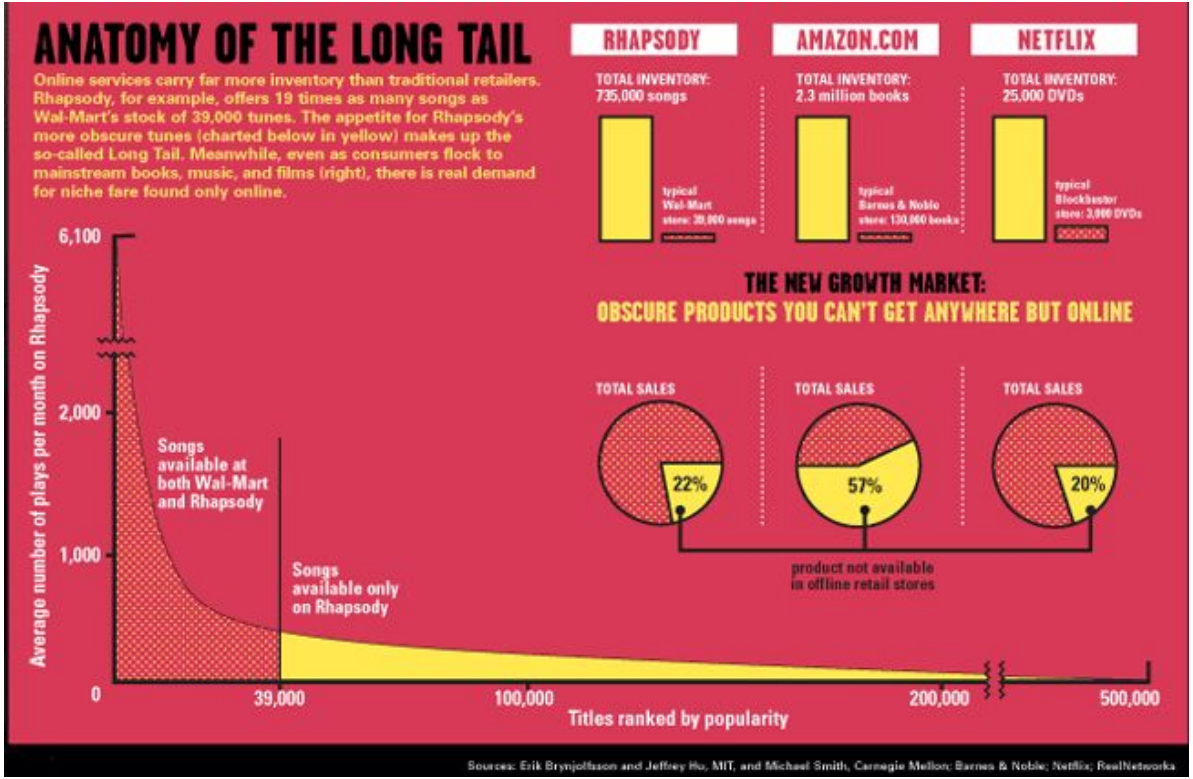

**Mining of Massive Datasets**

Jure Leskovec, Anand Rajaraman, Jeff Ullman

http://www.mmds.org/

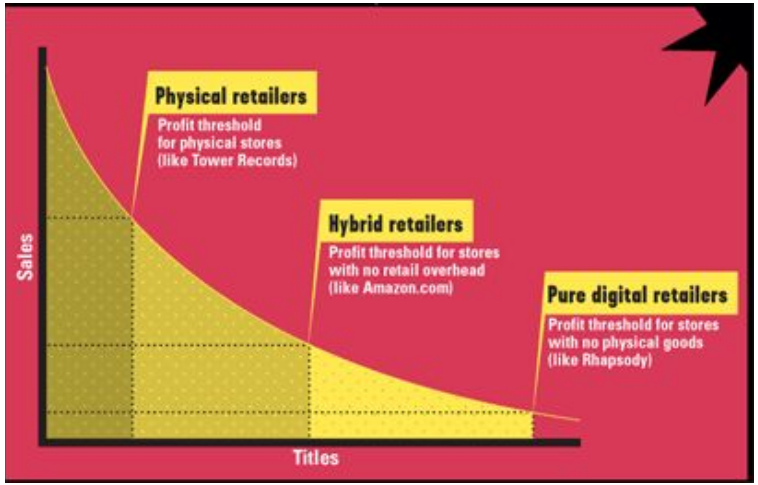
# The long tail

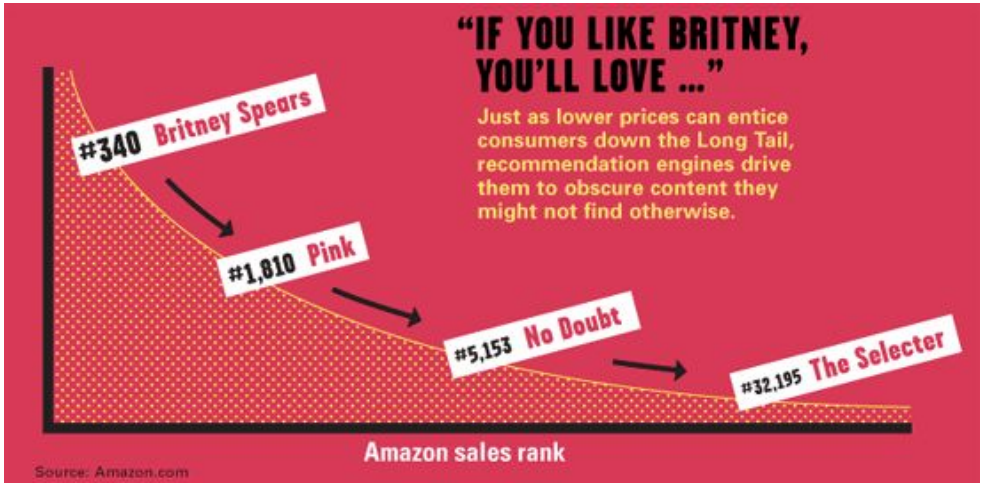
## Ejemplo: Into Thin Air y Touching Void



# The long tail

## Ejemplo: Into Thin Air y Touching Void



# Matriz de utilidad

- Existen *usuarios* e *ítems*. Los *usuarios* prefieren algunos *ítems* por sobre otros
- Ejemplo: Usuarios de Netflix y Películas. De 1 a 5 estrellas.

| | **P1** | **P2** | **P3** | **P4** | **P5** | **P6** | **...** | **$P_m$** |
|---|---|---|---|---|---|---|---|---|
| **Usuario 1** | 5 | 4 | | | 2 | | ... | 1 |
| **Usuario 2** | 2 | 1 | | 5 | | | ... | 5 |
| **Usuario 3** | | 1 | 5 | | 4 | 3 | ... | 2 |
| **Usuario 4** | 4 | | | 2 | 1 | | | |
| **...** | ... | ... | ... | | | | ... | |
| **Usuario *n*** | 1 | 2 | 5 | | 5 | | ... | 3 |

# Matriz de utilidad

- Existen *usuarios* e *ítems*. Los *usuarios* prefieren algunos *ítems* por sobre otros
- Ejemplo: Usuarios de Netflix y Películas. De 1 a 5 estrellas.

| | P1 | P2 | P3 | P4 | P5 | P6 | ... | $P_m$ |
|---|---|---|---|---|---|---|---|---|
| **Usuario 1** | 5 | 4 | ? | ? | 2 | ? | ... | 1 |
| **Usuario 2** | 2 | 1 | ? | 5 | ? | ? | ... | 5 |
| **Usuario 3** | ? | 1 | 5 | ? | 4 | 3 | ... | 2 |
| **Usuario 4** | 4 | ? | ? | 2 | 1 | ? | ... | ? |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... |
| **Usuario *n*** | 1 | 2 | 5 | ? | 5 | ? | ... | 3 |

# Matriz de utilidad

- Existen *usuarios* e *ítems*. Los *usuarios* prefieren algunos *ítems* por sobre otros
- Ejemplo: Usuarios de Netflix y Películas. De 1 a 5 estrellas.

El objetivo del sistema de recomendación es *poblar* la matriz de utilidad

| | P1 | P2 | P3 | P4 | P5 | P6 | ... | $P_m$ |
|---|---|---|---|---|---|---|---|---|
| **Usuario 1** | 5 | 4 | ? | ? | 2 | ? | ... | 1 |
| **Usuario 2** | 2 | 1 | ? | 5 | ? | ? | ... | 5 |
| **Usuario 3** | ? | 1 | 5 | ? | 4 | 3 | ... | 2 |
| **Usuario 4** | 4 | ? | ? | 2 | 1 | ? | ... | ? |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... |
| **Usuario *n*** | 1 | 2 | 5 | ? | 5 | ? | ... | 3 |

# Matriz de utilidad

- Existen *usuarios* e *ítems*. Los *usuarios* prefieren algunos *ítems* por sobre otros
- Ejemplo: Usuarios de Netflix y Películas. De 1 a 5 estrellas.

Ejemplo: Netflix tiene 150 millones suscriptores, 5 mil películas. La matriz tiene 750000000000 espacios, de los cuales la mayoría están vacíos.

- Cuando buscamos recomendar, interesa más recomendar ítems que van a gustar que aquellos que no van a gustar.

# Matriz de utilidad

- Existen *usuarios* e *ítems*. Los *usuarios* prefieren algunos *ítems* por sobre otros
- Ejemplo: Usuarios de Netflix y Películas. De 1 a 5 estrellas.

Ejemplo: Netflix tiene 150 millones suscriptores, 5 mil películas. La matriz tiene 750000000000 espacios, de los cuales la mayoría están vacíos.

- Cuando buscamos recomendar, interesa más recomendar ítems que van a gustar que aquellos que no van a gustar.
- En algunos casos, interesa mover a los usuarios del mainstream a la cola

# Matriz de utilidad

- Existen *usuarios* e *ítems*. Los *usuarios* prefieren algunos *ítems* por sobre otros
- Ejemplo: Usuarios de Netflix y Películas. De 1 a 5 estrellas.

Ejemplo: Netflix tiene 150 millones suscriptores, 5 mil películas. La matriz tiene 750000000000 espacios, de los cuales la mayoría están vacíos.

> - Cuando buscamos recomendar, interesa más recomendar ítems que van a gustar que aquellos que no van a gustar.
> - En algunos casos, interesa mover a los usuarios del mainstream a la cola
> - Algunas veces, ni siquiera hay calificaciones, solamente si vio o no (o escuchó, leyó, compró, etc.).

# Matriz de utilidad

- Existen *usuarios* e *ítems*. Los *usuarios* prefieren algunos *ítems* por sobre otros
- Ejemplo: Usuarios de Netflix y Películas. De 1 a 5 estrellas.

Ejemplo: Netflix tiene 150 millones suscriptores, 5 mil películas. La matriz tiene 750000000000 espacios, de los cuales la mayoría están vacíos.

El objetivo del sistema de recomendación es poblar la matriz de utilidad de una manera inteligente y bajo los requisitos que imponga cada entorno

# Matriz de utilidad

- Existen *usuarios* e *ítems*. Los *usuarios* prefieren algunos *íte*
- Ejemplo: Usuarios de Netflix y Películas. De 1 a 5 estrellas.

El objetivo del sistema de recomendación es *poblar* la matriz de utilidad

| | P1 | P2 | P3 | P4 | P5 | P6 | ... | $P_m$ |
|---|---|---|---|---|---|---|---|---|
| **Usuario 1** | 5 | 4 | | | 2 | ? | ... | 1 |
| **Usuario 2** | 2 | 1 | ? | 5 | ? | ? | ... | 5 |
| **Usuario 3** | | 1 | 5 | ? | 4 | 3 | ... | 2 |
| **Usuario 4** | 4 | ? | | 2 | 1 | ? | ... | |
| **Usuario 5** | | Vista | Vista | ? | Vista | ? | ... | |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... |

# ¿Cómo conseguir ratings?

# Matriz de utilidad

## Explícitamente

Pedir a los usuarios que puntúen los ítems.

- Los usuarios no suelen hacerlo
- Si lo hacen, puede estar sesgado (gente que prefiere puntuar cosas que no le gustan a puntuar cosas que sí, etc.).

## Implícitamente

Inferir a partir de acciones

- Ejemplo: compra muchas cosas de camping → le gusta el camping, aire libre, etc.
- ¿Qué pasa con las cosas que no le gustan?

# ¿Cómo funcionan los sistemas de recomendación?

# Algo cambió en los sistemas de recomendación...

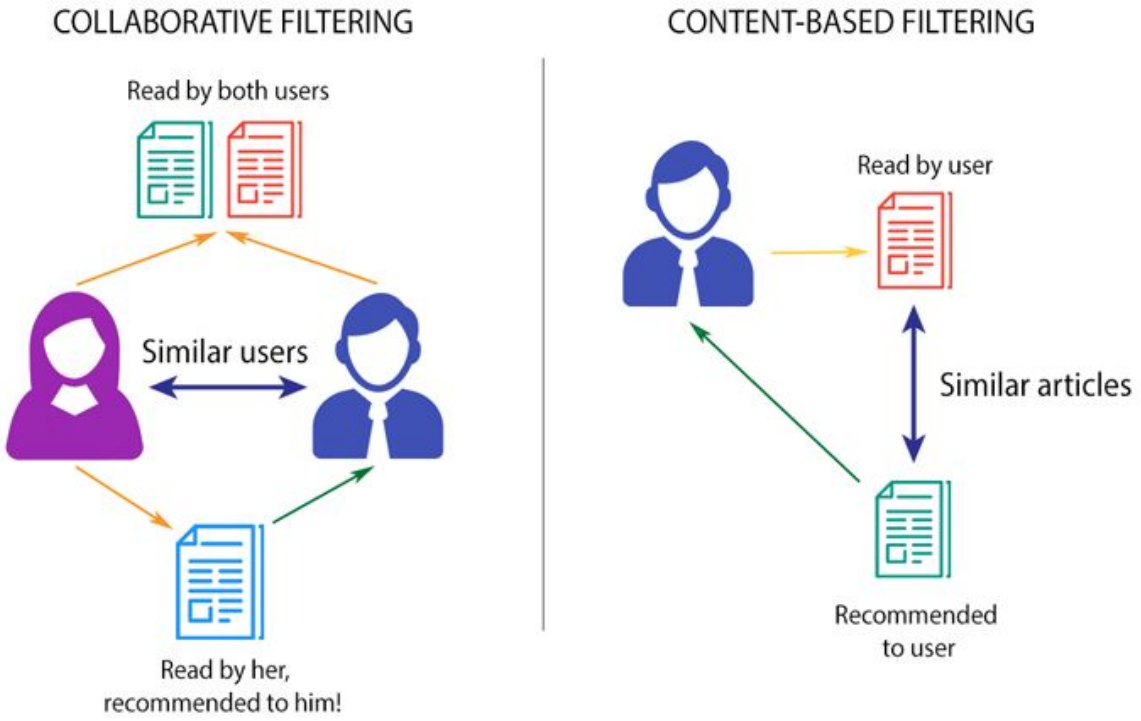
## HISTÓRICAMENTE

- Crítica de expertos, aclamadas/os por la crítica
- Listas de favoritos
- Listas de clásicos
- Más populares
- Recientes

## HOY

Recomendaciones específicas para el usuario

# Algo cambió en los sistemas de recomendación...



# Sistemas de recomendación • Tipos

| | ¿Cómo? | Ventaja | Desventaja / Problema |
|---|---|---|---|
| **Basado en contenidos** | Recomienda ítems con características similares a los que el usuario consumió (y, preferiblemente, indicó que le gustaban). | Basta con *conocer* los ítems para comenzar a recomendar | 1) calcular la similitud entre dos ítems puede ser una tarea difícil y muy costosa. En la mayoría de los casos hay que obtener atributos.<br><br>2) Suele recomendar ítems que no son novedosos para el usuario |

# Sistemas de recomendación • Tipos

| | ¿Cómo? | Ventaja | Desventaja / Problema |
|---|---|---|---|
| **Basado en contenidos** | Recomienda ítems con características similares a los que el usuario consumió (y, preferiblemente, indicó que le gustaban). | Basta con *conocer* los ítems para comenzar a recomendar | 1) calcular la similitud entre dos ítems puede ser una tarea difícil y muy costosa. En la mayoría de los casos hay que obtener atributos.<br><br>2) Suele recomendar ítems que no son novedosos para el usuario |
| **Filtro colaborativo** | Recomienda ítems basadas en medidas de similaridad entre ítems y/o usuarios. | No necesita *conocer* los ítems, en principio alcanza con la información de la matriz de utilidad | Necesito la matriz de utilidad |

# Sistemas de recomendación • Tipos

| | ¿Cómo? | Ventaja | Desventaja / Problema |
|---|---|---|---|
| **Basado en contenidos** | Recomienda ítems con características similares a los que el usuario consumió (y, preferiblemente, indicó que le gustaban). | Basta con *conocer* los ítems para comenzar a recomendar | 1) calcular la similitud entre dos ítems puede ser una tarea difícil y muy costosa. En la mayoría de los casos hay que obtener atributos.<br>2) Suele recomendar ítems que no son novedosos para el usuario |
| **Filtro colaborativo** | Recomienda ítems basadas en medidas de similaridad entre ítems y/o usuarios. | No necesita *conocer* los ítems, en principio alcanza con la información de la matriz de utilidad | Necesito la matriz de utilidad |
| **Pensarlo como problema de clasificación** | Podemos entrenar un clasificador para cada usuario | | Pocas calificaciones por usuario |

# Sistemas de recomendación • Tipos

| | ¿Cómo? | Ventaja | Desventaja / Problema |
|---|---|---|---|
| **Basado en contenidos** | Recomienda ítems con características similares a los que el usuario consumió (y, preferiblemente, indicó que le gustaban). | Basta con *conocer* los ítems para comenzar a recomendar | 1) calcular la similitud entre dos ítems puede ser una tarea difícil y muy costosa. En la mayoría de los casos hay que obtener atributos.<br>2) Suele recomendar ítems que no son novedosos para el usuario |
| **Filtro colaborativo** | Recomienda ítems basadas en medidas de similaridad entre ítems y/o usuarios. | No necesita *conocer* los ítems, en principio alcanza con la información de la matriz de utilidad | Necesito la matriz de utilidad |
| **Pensarlo como problema de clasificación** | Podemos entrenar un clasificador para cada usuario | | Pocas calificaciones por usuario |
| **Híbridos** | Combinar lo mejor de varios mundos | | |

# Sistemas de recomendación • Tipos

| | ¿Cómo? | Ventaja | Desventaja / Problema |
|---|---|---|---|
| **Basado en contenidos** | Recomienda ítems con características similares a los qu… consum… preferi… que le … | Basta con *conocer* los ítems para comenzar a recomendar | 1) calcular la similitud entre dos ítems puede ser una tarea difícil y muy costosa. En la mayoría de los casos hay …utos.<br><br>…dar ítems que no son …l usuario |
| **Filtro colaborativo** | Recom… basada… similari… y/o usu… | matriz de utilidad | …z de utilidad |
| **Pensarlo como problema de clasificación** | Podemos entrenar un clasificador para cada usuario | | Pocas calificaciones por usuario |
| **Híbridos** | Combinar lo mejor de varios mundos | | |

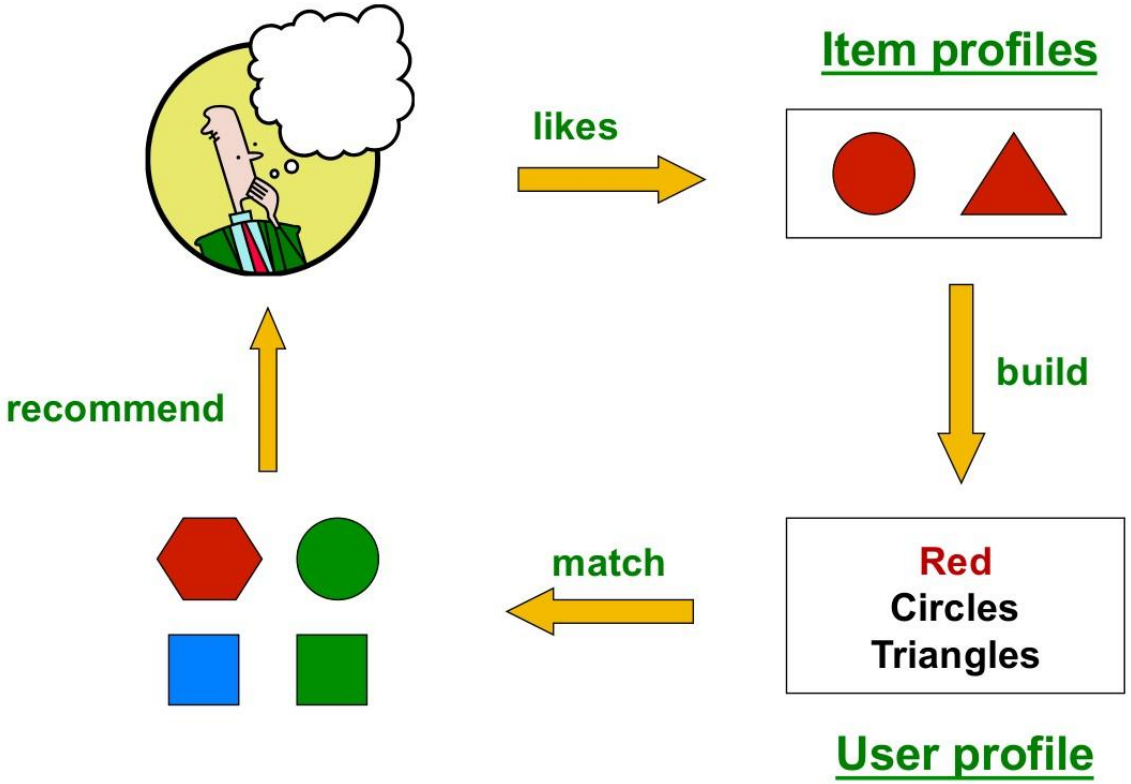
**Cold Start**

New items have no ratings, new users have no history.

# Sistemas de recomendación

1. **Basado en contenidos**
2. Filtro colaborativo

# SR • Basado en contenidos



# SR • Basado en contenidos

**Idea:** recomendar ítems al usuario que sean similares a aquellos que puntuó positivamente antes (o, en su defecto, que consumió). Para ello:

1. Para cada ítem, debemos construir un perfil.
   a. Casos sencillos: información fácilmente disponible. Películas: director, género, actores, año, etc.
   b. Casos no-sencillos. Debemos extraer features de los ítems. Noticias: hay que usar la batería de herramientas de NLP (tf-idf, etc.)
2. Idealmente, también hay que construir un perfil de qué cosas le gustan al usuario.
3. Usamos una métrica de distancia para encontrar ítems similares.
   a. Índice Jaccard
   b. Distancia coseno
4. Recomendamos

# SR • Basado en contenidos

No necesitamos información de otros usuarios. (Sin *Cold-Start*)

Puede recomendar a usuarios con “paladar exquisito” o único

Puede recomendar ítems nuevos o poco populares (basta ver su contenido)

Explicable

Hay que armar el perfil de los ítems. Puede ser difícil encontrar buenos features.

Difícil recomendar a nuevos usuarios.

Puede ser muy específico: no recomienda ítems por fuera perfil del usuario. El usuario puede tener muchos intereses.

# Sistemas de recomendación

1. Basado en contenidos
2. **Filtro colaborativo**

# SR • Filtro colaborativo

| | P1 | P2 | P3 | P4 | P5 | P6 | ... | $P_m$ |
|---|---|---|---|---|---|---|---|---|
| **Usuario 1** | 5 | 4 | | | 2 | | ... | 1 |
| **Usuario 2** | 2 | 1 | | 5 | | | ... | 5 |
| **Usuario 3** | | 1 | 5 | | 4 | 3 | ... | 2 |
| **Usuario 4** | 4 | | | 2 | 1 | | | |
| **...** | ... | ... | ... | | | | ... | |
| **Usuario *n*** | 1 | 2 | 5 | | 5 | | ... | 3 |

# SR • Filtro colaborativo

| | P1 | P2 | P3 | P4 | P5 | P6 | ... | $P_m$ |
|---|---|---|---|---|---|---|---|---|
| **Usuario 1** | 5 | 4 | ? | ? | 2 | ? | ... | 1 |
| **Usuario 2** | 2 | 1 | ? | 5 | ? | ? | ... | 5 |
| **Usuario 3** | ? | 1 | 5 | ? | 4 | 3 | ... | 2 |
| **Usuario 4** | 4 | ? | ? | 2 | 1 | ? | ... | ? |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... |
| **Usuario *n*** | 1 | 2 | 5 | ? | 5 | ? | ... | 3 |

# SR • Filtro colaborativo

Funciona para cualquier tipo de ítem (películas, libros, música, etc.).

Puede recomendar ítems por fuera del *perfil* del usuario

Necesitamos la matriz de utilidad

La matriz de utilidad está, en general, vacía y es muy grande. Esto trae dificultades computacionales.

No puede recomendar ítems que no hayan sido calificados previamente

Tiende a recomendar ítems populares

# ¡BREAK!

Ph. Credit: Drew Coffmann

# ¿Cómo funciona un filtro colaborativo?

# SR • Filtro colaborativo

Hay muchas formas de llenar la matriz.

**Ejemplo**:
Podemos probar con técnicas de clusterización para encontrar grupos de usuarios similares. De esos usuarios similares, los que tengan algún faltante en un ítem, se lo completa con, por ejemplo, el promedio del cluster.

> **Preprocesamiento**
> **Normalización**: hay usuarios que tienden a dar calificaciones altas y otros que tienden a dar calificaciones bajas. Entonces, restamos a cada calificación la calificación promedio del usuario.

# Vamos a contarles
# Descomposición UV

# SR • Filtro colaborativo • Descomposición UV

Reducción de dimensionalidad - Descomposición UV

$$\begin{bmatrix} 5 & 2 & 4 & 4 & 3 \\ 3 & 1 & 2 & 4 & 1 \\ 2 & & 3 & 1 & 4 \\ 2 & 5 & 4 & 3 & 5 \\ 4 & 4 & 5 & 4 & \end{bmatrix} = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ u_{31} & u_{32} \\ u_{41} & u_{42} \\ u_{51} & u_{52} \end{bmatrix} \times \begin{bmatrix} v_{11} & v_{12} & v_{13} & v_{14} & v_{15} \\ v_{21} & v_{22} & v_{23} & v_{24} & v_{25} \end{bmatrix}$$

# SR • Filtro colaborativo • Descomposición UV

Reducción de dimensionalidad - Descomposición UV

$$\begin{bmatrix} 5 & 2 & 4 & 4 & 3 \\ 3 & 1 & 2 & 4 & 1 \\ 2 & & 3 & 1 & 4 \\ 2 & 5 & 4 & 3 & 5 \\ 4 & 4 & 5 & 4 & \end{bmatrix} = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ u_{31} & u_{32} \\ u_{41} & u_{42} \\ u_{51} & u_{52} \end{bmatrix} \times \begin{bmatrix} v_{11} & v_{12} & v_{13} & v_{14} & v_{15} \\ v_{21} & v_{22} & v_{23} & v_{24} & v_{25} \end{bmatrix}$$

5x5 U: 5x2 V: 2x5

# SR • Filtro colaborativo • Descomposición UV

Reducción de dimensionalidad - Descomposición UV

$$\begin{bmatrix} 5 & 2 & 4 & 4 & 3 \\ 3 & 1 & 2 & 4 & 1 \\ 2 & & 3 & 1 & 4 \\ 2 & 5 & 4 & 3 & 5 \\ 4 & 4 & 5 & 4 & \end{bmatrix} = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ u_{31} & u_{32} \\ u_{41} & u_{42} \\ u_{51} & u_{52} \end{bmatrix} \times \begin{bmatrix} v_{11} & v_{12} & v_{13} & v_{14} & v_{15} \\ v_{21} & v_{22} & v_{23} & v_{24} & v_{25} \end{bmatrix}$$

5x5 U: 5x2 V: 2x5

**d**: lo elegimos, es un hiperparámetro

# SR • Filtro colaborativo • Descomposición UV

Reducción de dimensionalidad - Descomposición UV

$$\begin{bmatrix} 5 & 2 & 4 & 4 & 3 \\ 3 & 1 & 2 & 4 & 1 \\ 2 & & 3 & 1 & 4 \\ 2 & 5 & 4 & 3 & 5 \\ 4 & 4 & 5 & 4 & \end{bmatrix} = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ u_{31} & u_{32} \\ u_{41} & u_{42} \\ u_{51} & u_{52} \end{bmatrix} \times \begin{bmatrix} v_{11} & v_{12} & v_{13} & v_{14} & v_{15} \\ v_{21} & v_{22} & v_{23} & v_{24} & v_{25} \end{bmatrix}$$

Buscamos $u_{ij}$ y $v_{ij}$ de forma que cuando multipliquemos las matrices se aproximen bastante a los valores originales. Ej: $5 = u_{11} * v_{11} + u_{12} * v_{21}$

# SR • Filtro colaborativo • Descomposición UV

Reducción de dimensionalidad - Descomposición UV

$$\begin{bmatrix} 5 & 2 & 4 & 4 & 3 \\ 3 & 1 & 2 & 4 & 1 \\ 2 & \bigcirc & 3 & 1 & 4 \\ 2 & 5 & 4 & 3 & 5 \\ 4 & 4 & 5 & 4 & \end{bmatrix} = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ u_{31} & u_{32} \\ u_{41} & u_{42} \\ u_{51} & u_{52} \end{bmatrix} \times \begin{bmatrix} v_{11} & v_{12} & v_{13} & v_{14} & v_{15} \\ v_{21} & v_{22} & v_{23} & v_{24} & v_{25} \end{bmatrix}$$

Buscamos $u_{ij}$ y $v_{ij}$ de forma que cuando multipliquemos las matrices se aproximen bastante a los valores originales. Ej: $5 = u_{11}*v_{11} + u_{12}*v_{21}$

Para completar los lugares vacíos, simplemente ponemos lo que de la multiplicación de la derecha. Ej: ◯ $= u_{31}*v_{12} + u_{32}*v_{22}$

# SR • Filtro colaborativo • Descomposición UV

**Reducción de dimensionalidad - Descomposición UV**

¿Cómo encontramos los valores para U y V?

# SR • Filtro colaborativo • Descomposición UV

**Reducción de dimensionalidad - Descomposición UV**

## ¿Cómo encontramos los valores para U y V?

1. Necesitamos una métrica para minimizar. En general, RMSE para los **valores no nulos de la matriz**.

# SR • Filtro colaborativo • Descomposición UV

**Reducción de dimensionalidad - Descomposición UV**

## ¿Cómo encontramos los valores para U y V?

1. Necesitamos una métrica para minimizar. En general, RMSE para los **valores no nulos de la matriz**.
2. Empezamos en algún lugar al azar.

# SR • Filtro colaborativo • Descomposición UV

**Reducción de dimensionalidad - Descomposición UV**

## ¿Cómo encontramos los valores para U y V?

1. Necesitamos una métrica para minimizar. En general, RMSE para los **valores no nulos de la matriz**.
2. Empezamos en algún lugar al azar.
3. Buscamos el mínimo de la función de costo

# ¿Les suena?

# SR • Filtro colaborativo • Descomposición UV

**Reducción de dimensionalidad - Descomposición UV**

## ¿Cómo encontramos los valores para U y V?

1. Necesitamos una métrica para minimizar. En general, RMSE para los **valores no nulos de la matriz**.
2. Empezamos en algún lugar al azar.
3. Buscamos el mínimo de la función de costo

¡ Es el problema que resuelve el descenso por gradiente !

# ¿Sabías que...?

Netflix Challenge

# Netflix Challenge

- Lanzada en 2006, finalizada en 2009.
- 1.000.000 de dólares en premio a quienes mejoren su sistema de recomendación, CineMatch, en un diez por ciento. Varios premios más por año.
- **Entrenamiento**: ratings de ~500 mil usuarios a ~17 mil series y películas. En total, 100 millones de puntajes (no todos los usuarios puntúan todos los ítems).
- **Testeo**: 3 millones de ratings (que se guardó Netflix).

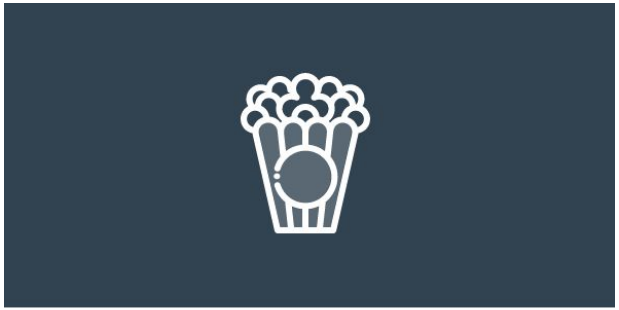
# Hands-on training

## Hands-on training

DS_Encuentro_40_Sistemas_Recomendacion.ipynb

# **Proyecto 2:**
# Lanzamiento
## Entrega 06

# Proyecto 2: Sistemas de recomendación (Entrega 06)



Entrega 6: Sistema de recomen...

Creá tu propio sistema de recomendación mediante el aprendizaje no supervisado

Principiante por Francisco Dorr

1. Bajar los materiales.
2. Leer la Checklist
3. ¡Empezar a trabajar en la entrega!

# Para la próxima:
# Data Science en mi vida

# Data Science en mi vida

¡Preparen sus charlas relámpago!
En 7 minutos con 7 slides comparte con tus compañeros:

En qué problemas estás aplicando lo aprendido en Data Science y cómo lo estás haciendo.
O bien, en qué problemas te gustaría aplicar Data Science y cómo lo harías.

¡Elige algún tema o proyecto que te interese y relaciónalo con lo aprendido!

# Para la próxima

1. Completar el notebook de hoy (Secciones 1 a 3).
2. Terminar de ver los videos de sistemas de recomendación.
3. Comenzar con la Entrega 06.
4. Preparar el relato “Data Science en mi vida”.

ACÁMICA