PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

IIC3633 Sistemas Recomendadores (2025-2)

Tarea 1

Indicaciones

- Fecha de entrega: **Viernes 5 de septiembre del 2025 , 20:00 horas.**
- La tarea debe realizarse **en grupos de dos personas**. Para inscribir sus grupos, deberán inscribirse en CANVAS a más tardar el día **viernes 22 de agosto del 2025 a las 20:00** . Quiénes no tengan grupo para esa fecha, serán asignados de manera aleatoria.
- La copia será sancionada con una nota 1,1 en la tarea, además de las sanciones disciplinarias correspondientes.
- Entrega a través de CANVAS, en el buzón de la tarea correspondiente. Solamente deberá entregar uno de los estudiantes del grupo, con los nombres de ambos estudiantes en todos los archivos entregados.
- Cada hora o fracción de atraso descuenta 0,5 puntos de la nota obtenida, llegando a 1,0 en 12 horas. Se considera como entrega el último archivo subido por alguno de los miembros del grupo. No se revisarán tareas que hayan sido subidas con anterioridad a la última.
- Se debe hacer la tarea en Google Colab o en Jupyter Notebooks para facilitar la revisión. Deberán entregar estos notebooks ejecutados como parte de su código.
- Los datos entregados contienen más información de la estrictamente necesaria para el desarrollo de las actividades. Está permitido utilizar esta información extra si se desean mejorar los métodos descritos en las actividades. Sin embargo, debe existir una justificación para el uso de los mismos y además, los métodos de las actividades no pueden perder su estructura.
- **Uso de IA**: Se permite el uso de modelos de lenguaje como ayuda para el desarrollo de la tarea siempre y cuando se cumplan las siguientes condiciones:
  - Se debe compartir la conversación completa de cada una de las sesiones usadas (link de ChatGPT, Claude, Gemini, etc)
  - El código y el informe debe ser redactado por ustedes. Se puede usar la IA para realizar consultas de contenido, ortografía y dudas de código, pero no se puede pedir a la IA que realice la tarea (o una parte de esta) por ustedes.

OBJETIVO

En esta tarea pondrán en práctica sus conocimientos sobre sistemas recomendadores utilizando un conjunto de datos de reseñas de la categoría de videojuegos obtenidas desde Amazon. Esto incluye los juegos, accesorios y consolas. Experimentarán con recomendación no personalizada, basada en feedback implícito y explícito, y recomendación basada en contenido.

EVALUACIÓN

La calificación de esta tarea se dividirá en dos instancias. La primera corresponderá al 30 % de la nota y consistirá en un práctico de métricas, el cual se evaluará en clases el día 26 de agosto. El 70 % restante se obtendrá a partir del desarrollo del práctico descrito a continuación. Además, esta tarea incluye la posibilidad de aumentar la calificación mediante bonos, los cuales se detallan en la sección *Bonus*.

DESCRIPCIÓN DEL CONJUNTO DE DATOS

En esta tarea utilizarán un dataset de preferencias de videojuegos. Pueden descargarlo en este enlace. Este set de datos se compone de los siguientes archivos:

- Dataset de entrenamiento (*train.csv*): **68,584** registros que contienen la evaluación de un videojuego realizada por un usuario. Cada fila incluye:
  - rating: Calificación entregada por el usuario al videojuego (valor entre 1 y 5).
  - review: Texto de la reseña escrita por el usuario.
  - title: Título de la reseña.
  - user_id: Identificador único del usuario que realizó la reseña.
  - timestamp: Momento en que se publicó la reseña, en formato fecha-hora.
  - item_title: Nombre del videojuego evaluado.
  - item_id: Identificador único del videojuego en el sistema.
- Dataset de validación (*validation.csv*): **22,861** registros con el mismo formato del set de entrenamiento. Tipicamente el dataset de validación se usa para optimizar hiperparámetros.
- Dataset de metadata de los productos (*videogames_metadata.csv*): **81,257** Cada fila contiene:
  - title: Título del producto.

- average_rating: Calificación promedio obtenida a partir de todas las reseñas de usuarios.
- rating_numbers: Número total de calificaciones registradas para el producto.
- features: Características listadas del producto (por ejemplo: modos de juego, plataformas soportadas, etc.).
- description: Descripción textual del producto.
- price: Precio del producto (en dólares).
- store: Nombre de la tienda o vendedor registrado.
- categories: Lista de categorías o géneros a los que pertenece el producto relacionado a videojuegos.
- author: Desarrollador o editor responsable del videojuego.
- item_id: Identificador único del producto en el sistema.
- image: URL de la imagen de portada del producto.

- Dataset de testeo de ratings (*videogames_bonus_rating.csv*): **6,859** registros de usuarios que hicieron reviews. Estos registros tienen la columna de rating entregada vacía y es la que hay que completar para participar por el bonus de la tarea.
- Dataset de testeo de rankings (*videogames_bonus_ranking.json*): **5,813** registros que tienen la id del usuario. Hay que rellenar cada usuario con una lista de 10 recomendaciones para poder optar al bonus.

ALGORITMOS A UTILIZAR

Para la predicción de los ratings entregados por los usuarios se deben utilizar los siguientes algoritmos:

- User-based collaborative filtering
- Item-based collaborative filtering
- FunkSVD
- SVD++ (opcional)

Para la generación de las listas de recomendación se deben utilizar los siguientes algoritmos:

- Random
- Item-based collaborative filtering
- FunkSVD
- ALS
- BPR
- Factorization Machines

LIBRERÍAS

Pueden utilizar cualquier librería en python implementadas para recomendación. Las más utilizadas son PyReclab, Surprise, Implicit y LigthFM, pero esto queda a su criterio.

MÉTRICAS DE EVALUACIÓN DE MODELOS

Para la primera parte se deberá utilizar RMSE para medir performance, mientras que en la segunda parte se deberá utilizar Recall@K, nDCG y MAP para medir la calidad de las recomendaciones, además de diversidad y novedad de las listas de recomendación. Recordar que se está trabajando con listas de 10 recomendaciones para cada usuario (Top-N recommendation).

ENTREGABLES

La tarea deberá ser entregada a través de la plataforma CANVAS, se les solicita enviar los siguientes archivos:

1. Informe de análisis de los datos entregados y de los métodos utilizados
2. Código de los algoritmos implementados
3. Resultados de la evaluación de las predicciones

A continuación encontrarán una descripción detallada de cada uno de estos.

1. Informe (4.2 pts.)

El informe debe estar en formato pdf y deberá ser parte de la entrega en canvas. El informe tiene dos partes.

1.1. Análisis de datos (0.8 pts.)

La primera consiste en un análisis de los datos (tablas y gráficos) que incluya, al menos, estadísticas de los usuarios, de los ítems, densidad del dataset (ítems por usuario y usuarios por ítem), como por ejemplo la tabla Cuadro 1. Así mismo debes reportar gráficos de distribución, como las figuras Figura 1 y Figura 2.

1.2. Métodos RecSys (3.4 pts.)

En la segunda parte deben comparar y analizar los métodos utilizados respecto a, al menos, su implementación (dificultades y otros), tiempo de ejecución y memoria requeridos, para estos dos se solicitará que estén de forma tabulada. Además, de las métricas:

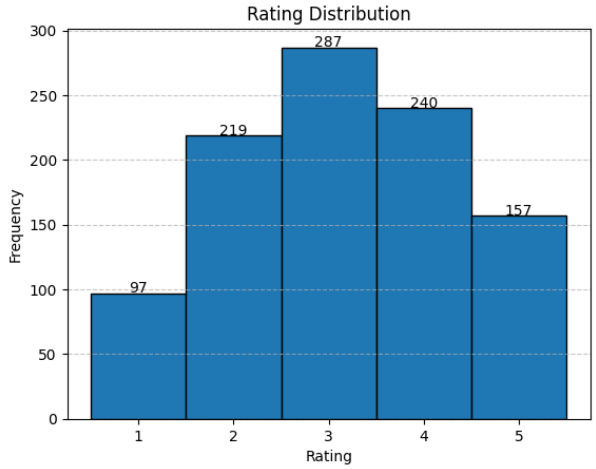

Figura 1: Ejemplo de gráfico mostrando la distribución de ratings en un dataset ficticio.

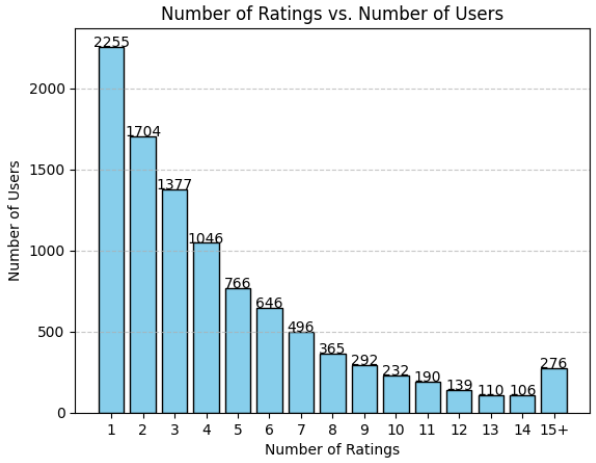

Figura 2: Ejemplo de gráfico mostrando una distribución long-tail del número de ratings por usuario.

Statistic	Training	Validation
Number of Users	80	20
Number of Items	400	100
Total Ratings	960	240
Average Number of Ratings per User	12.0	12.0
Average Number of Ratings per Item	2.4	2.4
Average Rating	3.7	3.8
Rating Standard Deviation	0.5	0.6
Highest Number of Ratings by a User	45	20
Highest Number of Ratings for an Item	14	5
Density (%)	3.0 %	12.0 %

Cuadro 1: Ejemplo de tabla con estadísticas ficticias de un dataset.

Method	RMSE	Recall@10	NDCG@10	MAP@10	Diversity	Novelty
Method A	0.845	0.175	0.890	0.340	0.620	0.740
Method B	0.812	0.185	0.905	0.350	0.640	0.755
Method C	0.830	0.180	0.895	0.345	0.635	0.750
Method D	0.870	0.170	0.885	0.335	0.610	0.730
Method E	0.910	0.160	0.870	0.320	0.590	0.710

Cuadro 2: Ejemplo de tabla comparando métodos de recomendación en varias métricas.

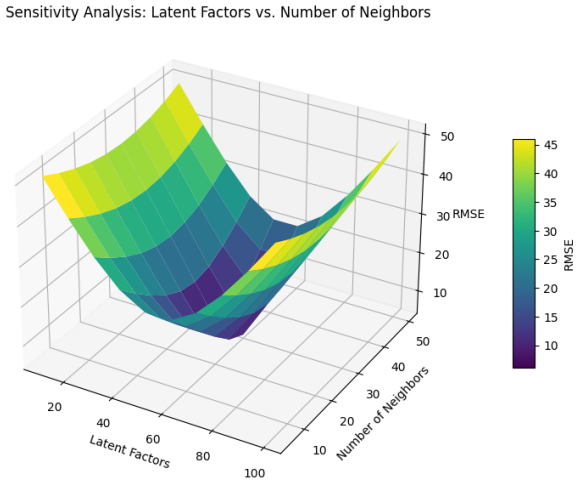

Figura 3: Ejemplo de gráfico 3D mostrando el efecto de dos variables sobre RMSE.

- RMSE (Predicción de ratings)
- Recall@10, MAP@10 y nDCG@10 (Generación de rankings)
- Diversidad (cantidad de estilos distintos recomendados en promedio)
- Novedad (self-information): $\frac{1}{|U|}\sum_{u\in U}\sum_{i\in L} log(1/pop_i)$

Para reportar estas métricas debes entrenar con el dataset de entrenamiento y **reportar los resultados sobre el dataset de validación**. Debes reportar con una tabla similar a la tabla Cuadro 2.

ANÁLISIS DE SENSIBILIDAD. Se espera que el informe contenga gráficos sobre el impacto de usar diferente número de vecinos (mínimo dos K en métodos KNN) y de factores latentes (mínimo tres números de factores en métodos como funkSVD, ALS y BPR). Considerar también informar sobre impacto de hiperparámetros como learning rate y regularización (λ). Para esto es ideal gráficos como la Figura 3.

2. Código (1.8 pts.)

Por cada uno de los métodos solicitados debe entregar el código que permita replicar los resultados obtenidos. Se solicita entregar uno o varios jupyter notebooks que permitan replicar experimentos.

Es obligatorio agregar un archivo README.md que permita entender la estructura de archivos y detalles necesarios para replicar los experimentos realizados.

3. Bono por uso de metadata de imágenes en sistemas content-based (hasta 0,5 pts)

Se otorgará un bono adicional de hasta 0.5 puntos a los grupos que integren de forma efectiva la metadata de imágenes contenida en el archivo videogames_metadata.csv para mejorar la calidad de las recomendaciones. La integración podrá realizarse de las siguientes formas (no excluyentes):

- Extracción de características visuales a partir de las imágenes de portada de los videojuegos (ej. mediante un modelo pre-entrenado en visión por computadora como ResNet, EfficientNet o CLIP).
- Generación de embeddings visuales y su posterior uso en un sistema de recomendación content-based, combinándolos con otros atributos como categorías, descripciones y características del videojuego.
- Hibridación del modelo content-based visual con un modelo colaborativo (ej. mezcla ponderada de puntuaciones o concatenación de embeddings).

Requisitos para optar al bono:

- Explicar en el informe el proceso de extracción de características de las imágenes, incluyendo el modelo utilizado, dimensionalidad de las representaciones y posibles técnicas de reducción de dimensionalidad aplicadas (PCA, UMAP, etc.).
- Describir cómo se integraron estas representaciones visuales en el sistema recomendador.
- Comparar el rendimiento del sistema con y sin el uso de información visual, reportando métricas como Recall@10, nDCG@10 y MAP@10.
- Justificar el impacto observado en las métricas y discutir posibles mejoras.

El puntaje del bono se asignará considerando tanto la correcta implementación técnica como la claridad del análisis y discusión de resultados en el informe.

4. Competición (Bono para top5)

Para optar a una bonificación, los grupos deberán enviar sus predicciones y recomendaciones completando los archivos videogames_bonus_rating.csv y videogames_bonus_ranking.json

con los resultados obtenidos por su mejor modelo. Deben enviarse ambos archivos para participar en la competición.

Las bonificaciones se otorgarán a los cinco mejores trabajos según la siguiente distribución:

- 1.5 punto para el primer lugar
- 1 puntos para el segundo lugar
- 0.75 punto para el tercer lugar
- 0.4 puntos para el cuarto lugar
- 0.15 puntos para el quinto lugar

La evaluación de las predicciones de rating se realizará utilizando el error RMSE, mientras que la evaluación de ranking se basará en el promedio de 3 métricas (NDCG, MAP y recall).

Para esta parte pueden ir más allá de los algoritmos analizados en las partes previas y experimentar con ensambles (ver ensemble learning) y utilizar otras técnicas no vistas en clases, como SVD++. A continuación se detallan los archivos:

- videogames_bonus_rating.csv: Este archivo tiene el fomarto de user_id, item_id, rating, pero con los valores de ratings vacios. La columna **rating** deberá ser llenada con sus predicciones de ratings para los correspondientes usuarios y productos de videojuegos.

	A	B	C
1	user_id	item_id	rating
2	526619	8539	
3	223040	8882	
4	475152	57834	
5	1392836	926	
6	84327	7954	
7	526482	4947	
8	234042	1977	
9	642813	14122	
10	349968	7372	
11	543405	3768	
12	456708	2323	
13	1036085	794	

Figura 4: Inicio del archivo videogames_bonus_rating.csv

- videogames_bonus_ranking.json: Este archivo contiene una lista de IDs de usuarios y columnas vacías con enumeración del 1 al 10. Se deben completar las columnas vacías con el top 10 de recomendaciones para el usuario del ID correspondiente, siendo la recomendación de la columna 1 la más deseada a recomendar y la de la columna 10 la menos deseada.

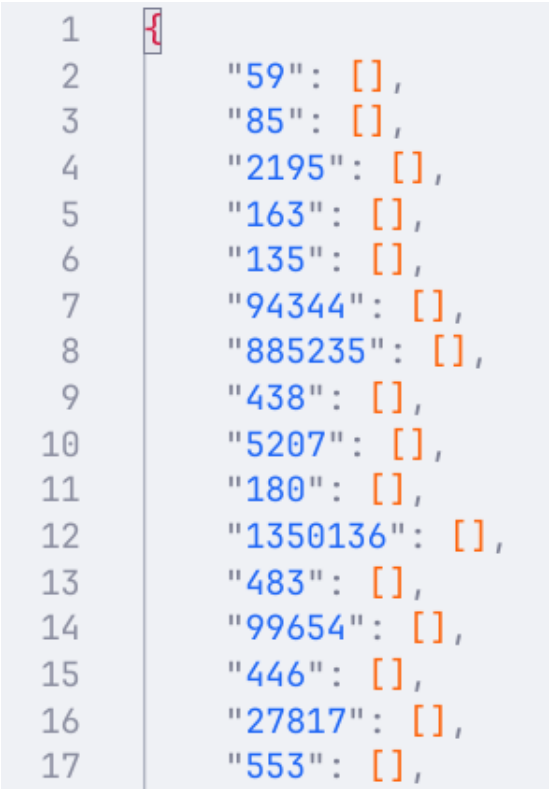

Figura 5: Inicio del archivo videogames_bonus_ranking.json