

# Sistemas inteligentes para la Gestión en la Empresa



## Universidad de Granada

### Sistema de recomendación de películas

Raúl Rodríguez Fernández  
José Antonio Laserna Beltrán

# Índice

1. Introducción.....	3
1.1 Presentación del Problema.....	3
1.2 Objetivos del trabajo.....	3
1.3 Descripción del Dataset.....	4
2. Preprocesamiento.....	6
2.1 Selección de los datos.....	6
2.2 Limpieza de filas malformadas.....	7
2.3 Imputación de valores perdidos.....	7
3. Estudio estadístico de los datos.....	10
3.1 Distribución por género.....	10
3.2 Películas mejor valoradas.....	11
3.3 Popularidad en función del año de estreno.....	11
3.4 Nube de palabras de palabras clave.....	12
3.5 Top de directores con más películas.....	13
3.6 Top directores con mejor valorados.....	13
3.7 Número de películas por año.....	14
4. Análisis predictivo.....	16
4.1 Objetivos del análisis.....	16
4.2 Preparación de los datos.....	17
4.4 Modelos utilizados.....	17
4.5 Evaluación de modelos.....	18
4.5.1 Regresión lineal.....	18
4.5.2 Random Forest.....	20
4.5.3 XGBoost.....	21
4.6 Predicción de ingresos para una nueva película.....	23
4.7 Predicción de ingresos y valoración a partir de texto (BERT + XGBoost).....	24
4.8 Resultados de las predicciones en películas nuevas.....	26
4.9 Conclusiones del análisis predictivo.....	29
5. Sistemas de recomendación.....	31
5.1 Introducción a los sistemas de recomendación.....	31
5.2 Redes neuronales para sistemas de recomendación.....	31
5.3 Sistemas de recomendación implementados.....	32
5.3.1 Sistema con el módulo surprise y SVD.....	32
5.3.2 Sistema basado en redes neuronales con PyTorch.....	33
5.3.3 Sistema de recomendación con LightFM.....	35
5.4 Resultados y comparación de los sistemas de recomendación.....	35
6. Conclusiones del proyecto.....	37
7. Bibliografía.....	38

# 1. Introducción

## 1.1 Presentación del Problema

En la industria cinematográfica, la predicción del éxito de una película es una tarea de gran interés tanto para productores como para distribuidores, plataformas de streaming y departamentos de marketing. Lanzar una película conlleva una gran inversión económica, y las decisiones sobre el guión, el reparto, el director o la campaña promocional deben tomarse con el mayor grado de certidumbre posible.

Ante esta realidad, surgen preguntas clave:

- ¿Qué factores determinan el éxito comercial de una película?
- ¿Es posible anticipar su impacto en taquilla o su valoración por parte del público antes incluso de su estreno?

Además, desde una perspectiva de recomendación:

- ¿Podemos encontrar películas similares en función de su sinopsis, género o estilo narrativo?

Gracias al crecimiento del análisis de datos y al uso de modelos de aprendizaje automático, hoy es posible analizar grandes volúmenes de información de películas pasadas y entrenar algoritmos que detecten patrones y relaciones significativas.

## 1.2 Objetivos del trabajo

El objetivo principal de este trabajo es desarrollar un modelo inteligente que permita predecir el éxito de una película a partir de diversas características disponibles antes de su estreno. Para ello, se plantea un enfoque basado en ciencia de datos que combina:

1. Predicción de ingresos (revenue) y valoración media (vote\_average) usando variables como:
  - Presupuesto, duración, año de estreno
  - Géneros, director, productora
  - Sinopsis y descripción textual (NLP)
2. Análisis de la importancia de los factores predictivos mediante técnicas como Random Forest y XGBoost.
3. Construcción de un sistema basado en texto que, dado un resumen de una película, sea capaz de:
  - Estimar su éxito potencial
  - Recomendar películas similares en estilo y temática

Este trabajo busca integrar técnicas avanzadas de preprocesamiento, visualización, modelado predictivo y procesamiento de lenguaje natural (NLP), aplicadas a un dominio real y con impacto comercial directo como es el cine.

## 1.3 Descripción del Dataset<sup>1</sup>

El dataset con el que se va a trabajar consiste en una colección de datos sobre 50000 películas aproximadamente. Esta colección de películas contiene distintos datasets con información diversa. El dataset principal, `movies_metadata`, contiene los siguientes datos:

- **Adult:** indica si la película debe ser restringida solo a adultos.
- **Belongs to collection:** indica si la película es parte de una saga/grupo/colección.
- **Budget:** presupuesto de la película.
- **Géneros:** géneros a los que pertenece la película.
- **Homepage:** link a la web oficial de la película.
- **Id:** identificador de la película en la base de datos
- **Imdb Id:** identificador en la base de datos de Imdb<sup>2</sup>.
- **Original language:** lenguaje en el que se rodó la película.
- **Título original:** título original de la película (escrito en el idioma original).
- **Overview:** sinopsis de la película.
- **Popularidad:** métrica que indica cómo de popular es la película (a mayor valor más popular). Por desgracia no hay información sobre cómo se llega a un valor concreto de popularidad
- **Poster path:** enlace al póster de la película en la base de datos.
- **Production companies:** empresas que produjeron la película.
- **Production countries:** países que produjeron la película.
- **Release date:** fecha en la que se estrenó la película.
- **Revenue:** recaudación de la película.
- **Runtime:** duración de la película.
- **Spoken languages:** idiomas hablados en la película/idiomas en los que se ofrece la película (de nuevo no hay información sobre el significado de este campo en el dataset).
- **Status:** estado de la película (estrenada, en proceso, cancelada).
- **Title:** Título en inglés de la película.
- **Tagline:** frase pegadiza que suele aparecer en el póster y que sirve como gag publicitario.
- **Video:** no sabemos qué significa este dato.
- **Vote average:** puntuación media del 0 al 10 que le dieron los usuarios a esta película.
- **Vote count:** cuántas personas han valorado la película.

---

<sup>1</sup> <https://www.kaggle.com/datasets/rounakbanik/the-movies-dataset>

<sup>2</sup> <https://www.imdb.com/es-es/>

Además, se tienen otros dataset adicionales:

- credits.csv: información de cada uno de los participantes en la película.
- keywords.csv: contiene palabras clave asociadas a cada película.
- links.csv, links\_small.csv: contienen links a otras bases de datos como imdb
- ratings.csv, ratings\_small.csv: datos de la valoración de un usuario (identificado por un id) a una película concreta.

Con estos datos, nuestro trabajo será el siguiente:

- ☐ Preprocesamiento de los datos
- ☐ Análisis y descripción estadística de los datos
- ☐ Modelos predictivos
- ☐ Sistema de recomendación

## 2. Preprocesamiento

### 2.1 Selección de los datos

Vistos los datos disponibles, lo primero que se hizo fue añadir al conjunto principal de los datos parte de los conjuntos de datos adicionales. En concreto, se añadieron las keywords correspondientes a cada película y el director de la película.

A continuación se ha hecho un filtrado de las columnas que consideramos que no son relevantes para el problema. Inicialmente se hizo la siguiente selección:

Columnas a eliminar:

- Budget
- Homepage
- Original language
- Spoken language
- Poster path
- Overview
- Production countries
- Revenue
- Status
- Tagline
- Video

Dudas:

- Production companies
- Release date
- Runtime

Sin embargo, tras evaluar las posibilidades a la hora de realizar predicciones se decidió volver a considerar qué variables eliminar a las siguientes:

Columnas a eliminar:

- Homepage
- Original language
- Poster path
- Overview
- Production countries
- Revenue
- Status
- Video
- Tagline

Una vez hecho esto los datos quedan de la siguiente manera:

adult	belongs_to_collection	budget	genres	id	imdb_id	overview	popularity	production_companies	release_date	revenue	runtime	title	vote_average	vote_count	keywords	dir
False	{'id': 10194, 'name': 'Toy Story Collection', ...}	30000000	[{'id': 16, 'name': 'Animation'}, {'id': 35, 'name': 'Comedy'}]	862	tt0114709	Led by Woody, Andy's toys live happily in his room.	21.946943	[{'name': 'Pixar Animation Studios', 'id': 3}]	1995-10-30	373554033.0	81.0	Toy Story	7.7	5415.0	[jealousy, toy, boy, friendship, friends, rivalry, ...]	Lars Kld
False	NaN	65000000	[{'id': 12, 'name': 'Adventure'}, {'id': 14, 'name': 'Fantasy'}]	8844	tt0113497	When siblings Judy and Peter discover an enchanted forest, they must find out why their father went missing before it's too late.	17.015539	[{'name': 'TriStar Pictures', 'id': 559}, {'name': 'Walt Disney Pictures', 'id': 104}]	1995-12-15	262797249.0	104.0	Jumanji	6.9	2413.0	[board game, disappearance, based on children's book, ...]	John Dahl
False	{'id': 119050, 'name': 'Grumpy Old Men Collect...	0	[{'id': 10749, 'name': 'Romance'}, {'id': 35, 'name': 'Comedy'}]	15602	tt0113228	A family wedding reignites the ancient feud between two bickering old men.	11.7129	[{'name': 'Warner Bros.', 'id': 6194}, {'name': 'Columbia Pictures', 'id': 104}]	1995-12-22	0.0	101.0	Grumpier Old Men	6.5	92.0	[fishing, best friend, duringcreditsstinger, ...]	Hector Daz
False	NaN	16000000	[{'id': 35, 'name': 'Comedy'}, {'id': 18, 'name': 'Drama'}]	31357	tt0114885	Cheated on, mistreated and stepped on, the woman returns to her abusive ex-husband and finds out how he's really living.	3.859495	[{'name': 'Twentieth Century Fox Film Corporation', 'id': 104}]	1995-12-22	81452156.0	127.0	Waiting to Exhale	6.1	34.0	[based on novel, interracial relationship, sin...	F. Whitney
False	{'id': 96871, 'name': 'Father of the Bride Col...	0	[{'id': 35, 'name': 'Comedy'}]	11862	tt0113041	Just when George Banks has recovered from his divorce, his ex-wife announces she's pregnant.	8.387519	[{'name': 'Sandollar Productions', 'id': 5842}, {'name': 'Columbia Pictures', 'id': 104}]	1995-02-10	76578911.0	106.0	Father of the Bride Part II	5.7	173.0	[baby, midlife crisis, confidence, aging, ...]	Clay A. Johnson

Como se puede observar, los datos varían desde tipos numéricos, texto a listas y diccionarios.

## 2.2 Limpieza de filas malformadas

Un análisis detallado de los datos ha revelado la existencia de 3 filas que no has sido bien formateadas:

```
0 s movies['adult'].value_counts()
```



	count
adult	
False	46620
True	9
- Written by Ørnås	1
Rune Balot goes to a casino connected to the October corporation to try to wrap up her case once and for all.	1
Avalanche Sharks tells the story of a bikini contest that turns into a horrifying affair when it is hit by a shark avalanche.	1

dtype: int64

Asimismo, se han eliminado 4 películas que no tienen título.

## 2.3 Imputación de valores perdidos

Un análisis exploratorio de los datos muestra que hay ciertas columnas que presentan valores perdidos:

	<code>movies.isnull().sum()</code>
	
	0
adult	0
belongs_to_collection	42055
budget	0
genres	0
id	0
imdb_id	17
overview	995
popularity	4
production_companies	4
release_date	88
revenue	4
runtime	268
title	4
vote_average	4
vote_count	4
keywords	1
director	918

Ahora hay que determinar cómo imputar estos valores nulos. Hay que señalar que el gran porcentaje de valores nulos en la columna belongs to collection tiene sentido pues es la norma que una película no pertenezca a una colección. Otras columnas pueden ser ignoradas puesto que no van a tener ningún desempeño en los modelos predictivos como pueden ser los imdb id.

Respecto a la imputación de valores perdidos se ha tomado la siguiente estrategia para cada una de las columnas:

- Belongs to collection: se rellenan con: {"id": None, "name": "No Collection", "poster\_path": null, "backdrop\_path": null}
- Overview: se rellenan con unknown.
- Popularity, Revenue, Vote average y vote count: se rellenan con 0.
- Production companies: se rellenan con: {'name': 'No Company', 'id': None}



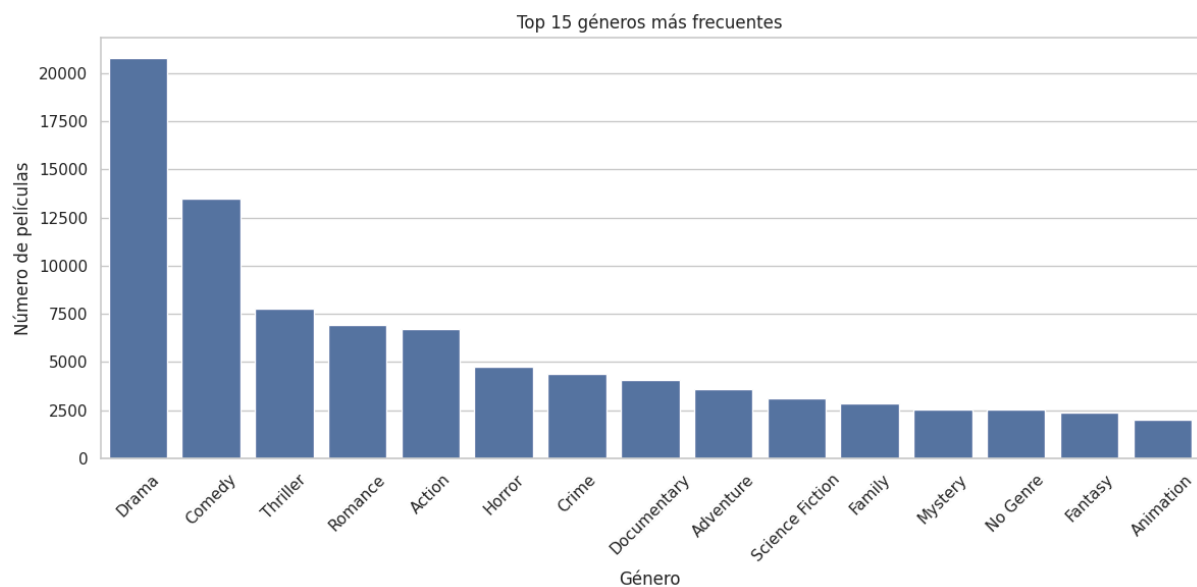
- Release date: se le asigna una fecha aleatoria entre 1900 y 2015.
- Runtime: se rellenan con la media.
- Genero: se rellenan con {'id': None, 'name': 'No Genre'}
- Keywords: se rellena con ['Film']

## 3. Estudio estadístico de los datos

Procedamos ahora a dar algunas métricas sobre los datos disponibles.

### 3.1 Distribución por género

Con el objetivo de conocer la composición temática del conjunto de datos, se ha realizado un análisis de frecuencia sobre los géneros cinematográficos más representados en el dataset.



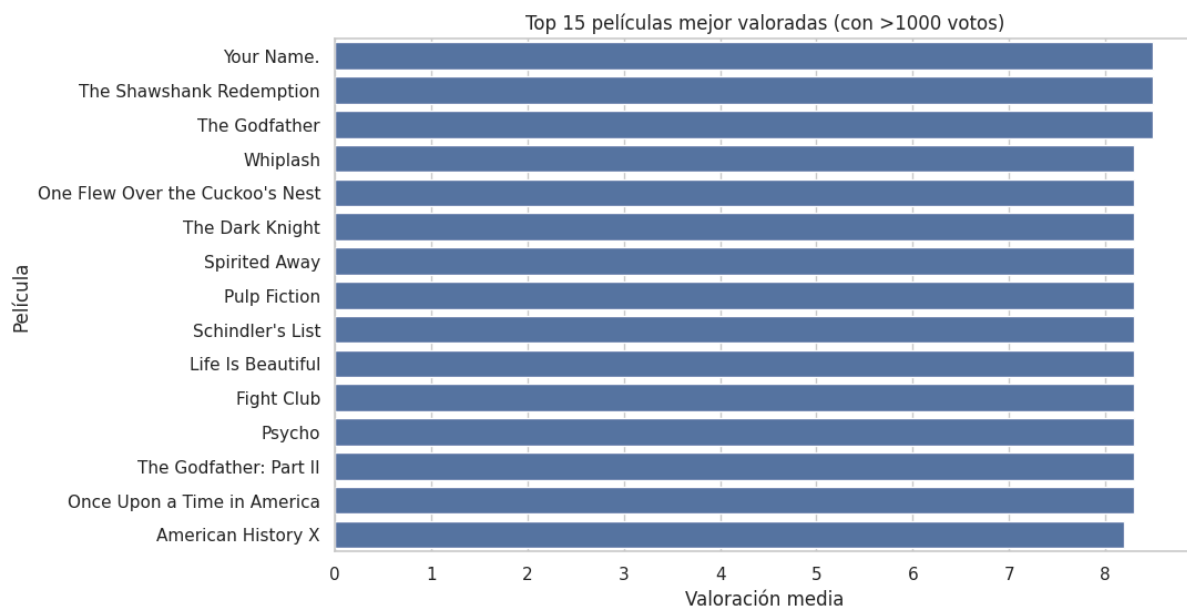
Como se puede observar, los géneros más comunes son drama, con más de 20.000 películas, destaca como el género predominante en la base de datos. Le siguen Comedia y Thriller, con aproximadamente 13.000 y 7.000 películas, respectivamente. Otros géneros como Romance, Action y Horror también tienen una presencia significativa.

Esta distribución es coherente con la tendencia general de la industria cinematográfica, donde el drama y la comedia suelen ser géneros ampliamente producidos. Además, la variedad de géneros menos frecuentes como Misterio, Fantasía o Animación refleja la diversidad temática del dataset, lo que resulta relevante a la hora de construir modelos de recomendación que consideren la heterogeneidad de gustos del público.

## 3.2 Películas mejor valoradas

Uno de los indicadores más relevantes de la calidad percibida de una película es su valoración media por parte del público. Para evitar sesgos debidos a un número reducido de opiniones, se ha considerado únicamente aquellas películas que cuentan con más de 1.000 valoraciones.

La siguiente gráfica muestra las 15 películas con mayor puntuación media dentro del dataset:



Entre las películas mejor valoradas destacan Your Name, una película de animación japonesa que encabeza el ranking con una valoración media superior a 8. Clásicos como The Shawshank Redemption, The Godfather, Pulp Fiction y Schindler's List reflejan el reconocimiento crítico y popular que han mantenido con el paso del tiempo.

También aparecen películas más recientes como Whiplash o The Dark Knight, lo que demuestra la capacidad del dataset para recoger tanto títulos clásicos como contemporáneos.

## 3.3 Popularidad en función del año de estreno

Para analizar cómo ha evolucionado el interés del público a lo largo del tiempo, se ha representado la popularidad media de las películas en función de su año de estreno.

La siguiente gráfica muestra dicha evolución desde finales del siglo XIX hasta la actualidad:



La popularidad permanece baja y estable hasta mediados del siglo XX, con valores próximos a 1. A partir de los años 80 se aprecia una tendencia ascendente que se acentúa notablemente a partir del año 2000. Se observan picos muy marcados en los años recientes (especialmente entre 2015 y 2020), lo que probablemente refleja un mayor volumen de datos y visualizaciones en plataformas digitales.

### 3.4 Nube de palabras de palabras clave

Con el fin de obtener una visión general sobre las temáticas más recurrentes en el conjunto de películas, se ha generado una nube de palabras a partir de las keywords asociadas a cada título.

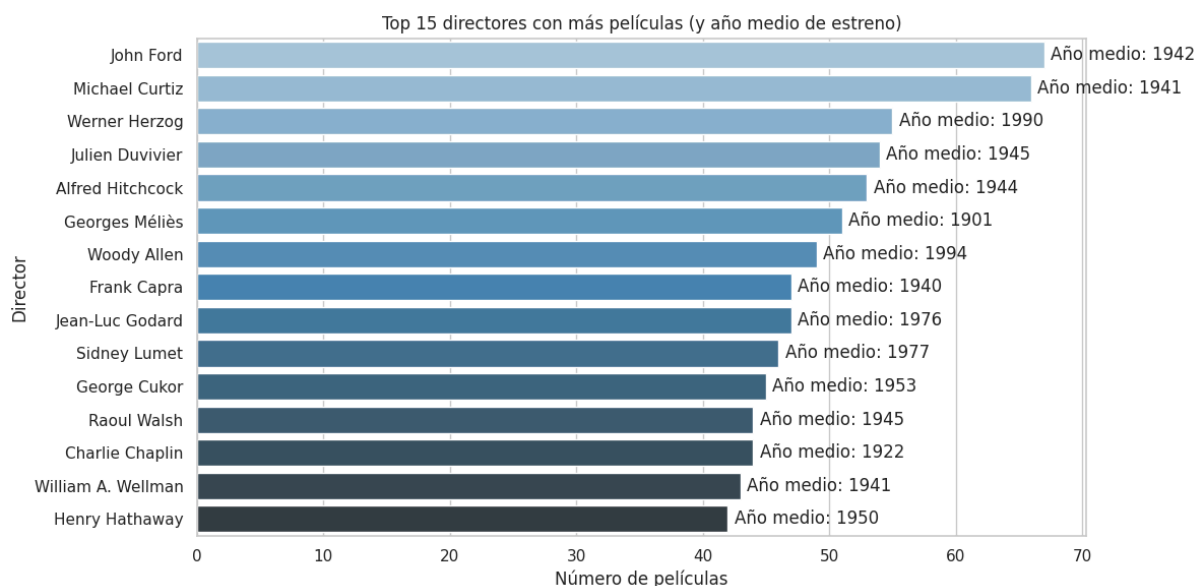
La nube refleja la frecuencia relativa de cada palabra clave, donde el tamaño del texto es proporcional a su aparición en el dataset:



Las palabras clave más destacadas son “woman director” y “independent film”, lo que refleja una mayor visibilidad de películas dirigidas por mujeres y del cine independiente. Términos como “love”, “murder”, “death”, “revenge” o “high school” revelan las temáticas más tratadas en las narrativas cinematográficas. También destacan localizaciones (“new york”, “london”) y géneros específicos (“biography”, “musical”, “sport”, “war film”).

### 3.5 Top de directores con más películas

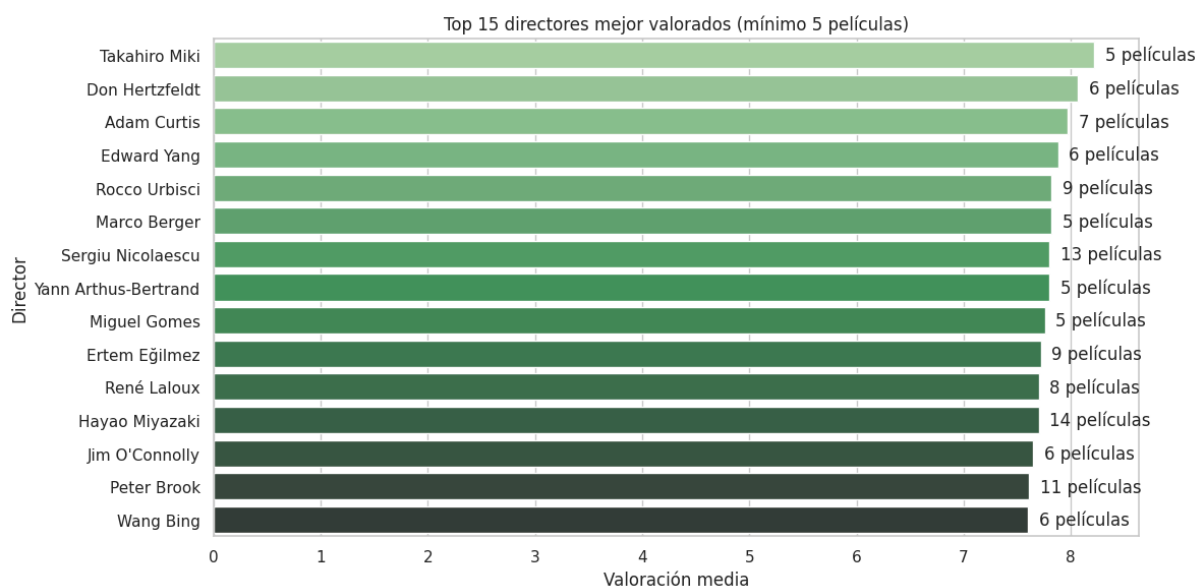
Para identificar a los realizadores más productivos dentro del conjunto de datos, hemos generado un ranking con los 15 directores con mayor número de películas registradas, indicando también el año medio de estreno de sus obras.



El director con mayor presencia es John Ford, con más de 60 películas, seguido por Michael Curtiz y Werner Herzog. La mayoría de los directores del top desarrollaron su carrera entre los años 1920 y 1970, como reflejan los años medios de estreno. Destaca también la presencia de directores clásicos como Alfred Hitchcock, Charlie Chaplin o Georges Méliès, lo que indica una importante representación de cine histórico en el dataset.

### 3.6 Top directores con mejor valorados

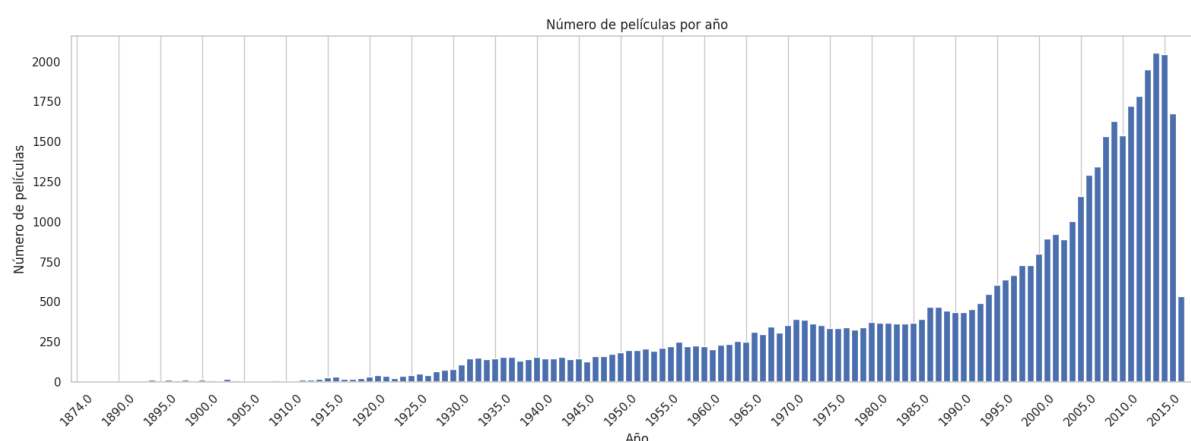
Para evaluar la calidad percibida del trabajo de los directores, también hemos generado un ranking con los 15 directores mejor valorados, calculando la media de puntuaciones obtenidas por sus películas. Se ha fijado un mínimo de 5 películas por director para garantizar robustez en los resultados.



Lideran la clasificación directores como Takahiro Miki, Don Hertzfeldt y Adam Curtis, todos ellos con puntuaciones medias cercanas a 8 sobre 10. Llama la atención la presencia de figuras del cine de animación y documental como Hayao Miyazaki o Peter Brook, así como representantes del cine europeo y asiático menos comercial. Este ranking destaca a autores con una trayectoria más selectiva pero altamente valorada, que pueden no aparecer en el top de directores con más trabajos.

### 3.7 Número de películas por año

Para analizar la evolución del volumen de producción cinematográfica a lo largo del tiempo, se ha realizado un recuento del número de películas estrenadas por año, desde finales del siglo XIX hasta la actualidad.



El gráfico muestra un crecimiento progresivo en la producción de películas desde principios del siglo XX, con una aceleración notable a partir de los años 90. El máximo histórico se alcanza en torno al año 2017, con más de 2000 películas registradas. A

partir de 2020, se observa una caída abrupta, que podría estar asociada al impacto de la pandemia de COVID-19 en la industria. Aún así el crecimiento sostenido desde los años 70 refleja la expansión del cine a nivel global, el abaratamiento de los medios de producción y la aparición de nuevas plataformas de distribución.

## 4. Análisis predictivo

A partir de las variables extraídas y tratadas durante las etapas previas, se construyen modelos que permiten anticipar el comportamiento de una película en términos de ingresos económicos (revenue) o valoración media (vote\_average). Para ello, se han empleado tanto modelos clásicos como técnicas avanzadas de aprendizaje automático, incluyendo árboles de decisión, random forest y modelos de boosting como XGBoost.

Este análisis permite no solo identificar qué técnicas son más eficaces para anticipar el éxito de una película, sino también conocer qué variables (sinopsis, género, director, presupuesto, etc.) tienen mayor peso en dicha predicción. Se trata, en definitiva, de ofrecer una herramienta orientada a la toma de decisiones basada en datos.

### 4.1 Objetivos del análisis

El objetivo principal del análisis predictivo es evaluar si es posible anticipar el éxito de una película antes de su lanzamiento mediante técnicas de ciencia de datos. Para ello, se abordan dos tipos de problemas:

- **Regresión:** el objetivo fue estimar variables continuas asociadas al rendimiento de una película como:
  - **revenue:** ingresos generados por la película.
  - **vote\_average:** puntuación media otorgada por los usuarios.

Para ello se construyeron modelos que intentan predecir un valor numérico lo más cercano posible al valor real observado. Esta tarea es especialmente relevante para simular escenarios previos al lanzamiento, donde aún no se dispone de datos como número de votos o ingresos reales.

- **Clasificación:** determinación de si una película puede considerarse un "éxito comercial", utilizando como umbral de referencia un ingreso superior a cierta cantidad. Para llevarlo a cabo se reformuló el problema del éxito comercial como una clasificación binaria:

*¿Será una película un éxito?  $\rightarrow 1$  si  $revenue > \$100$  millones, 0 en caso contrario.*

Esto permite establecer un sistema de detección temprana de éxitos, útil para la toma de decisiones estratégicas (inversión, distribución, marketing). La clasificación se evaluó con métricas como accuracy, matriz de confusión, precisión/recall y AUC-ROC.



## 4.2 Preparación de los datos

En el desarrollo de estos modelos se utilizaron diversas variables con el objetivo de enriquecer la capacidad de aprendizaje del sistema. Entre las más relevantes destacan los campos narrativos, como la sinopsis y el eslogan publicitario de cada película, los cuales se integraron en una única columna denominada `full_description` para capturar de forma conjunta el contexto semántico y promocional de la obra. A este contenido textual se sumaron otras variables de carácter categórico y numérico, como el género (representado en texto plano), el nombre del director principal y, en algunos modelos más completos, el presupuesto de la película, su nivel de popularidad, la pertenencia a una colección determinada, la principal productora involucrada y el año de estreno. Este conjunto de variables ofreció una combinación equilibrada entre datos textuales y estructurados, permitiendo así experimentar con modelos híbridos tanto basados en contenido como en aspectos colaborativos o estadísticos.

El tratamiento de los datos fue un paso fundamental para garantizar la calidad del modelo. En primer lugar, se depuró el conjunto eliminando aquellas películas que no contaban con sinopsis o que presentaban ingresos igual a cero, ya que tales registros aportan poco valor informativo y podían inducir ruido en el entrenamiento. También se descartaron valores atípicos extremos, como películas con ingresos superiores a los 1.000 millones de dólares, con el fin de evitar que estos casos desproporcionados sesgan las métricas y el ajuste de los modelos.

En cuanto a la ingeniería de características, se aplicaron técnicas como la vectorización TF-IDF sobre los géneros y el uso de codificación one-hot para representar al director. Asimismo, se exploró una versión más sofisticada basada en la codificación semántica de los textos mediante el modelo preentrenado BERT (all-MiniLM-L6-v2), lo que permitió obtener representaciones más ricas y contextuales del contenido narrativo de las películas. Finalmente, los datos fueron divididos en conjuntos de entrenamiento y prueba siguiendo una proporción 80/20 mediante la función `train_test_split`, garantizando así una evaluación adecuada y realista del rendimiento de los modelos desarrollados.

## 4.4 Modelos utilizados

Para evaluar el rendimiento predictivo en distintos escenarios, se aplicaron diversos algoritmos de aprendizaje supervisado, abarcando desde modelos básicos hasta técnicas más sofisticadas. El punto de partida fue la regresión lineal, empleada como modelo baseline para detectar posibles relaciones lineales entre las variables. Aunque sencilla y útil como referencia inicial, su capacidad resulta limitada ante relaciones no lineales o estructuras de datos complejas, como las que surgen del tratamiento de texto vectorizado.

Entre los modelos más interpretables se incluyó el modelo Random Forest, compuesto por un conjunto de árboles de decisión, mostró un mejor desempeño al manejar de forma eficiente tanto variables categóricas como texto vectorizado y valores atípicos. Este enfoque obtuvo resultados aceptables, alcanzando un coeficiente de determinación  $R^2$  de aproximadamente 0.76 en la predicción de ingresos (revenue) al usar todas las variables estructurales disponibles.

Para mejorar aún más el rendimiento, se implementó XGBoost, un algoritmo de boosting que optimiza los errores de modelos previos y permite un control más preciso del sobreajuste. Este modelo fue el que arrojó mejores resultados cuando se utilizaron representaciones textuales avanzadas, como TF-IDF o BERT, destacando especialmente en contextos donde predominaban variables semánticas como sinopsis, géneros y director. Finalmente, se desarrolló una variante que combinó Sentence-BERT (all-MiniLM-L6-v2) con XGBoost. En este caso, se transformaron las descripciones narrativas (sinopsis y tagline) en vectores semánticos, los cuales se enriquecieron con variables categóricas para entrenar el modelo. Esta combinación demostró ser la más eficaz cuando se disponía exclusivamente de texto y contexto, obteniendo un  $R^2$  de 0.34 para revenue y 0.12 para vote\_average, lo que confirma la relevancia del contenido narrativo como fuente informativa en modelos de predicción.

## 4.5 Evaluación de modelos

### 4.5.1 Regresión lineal

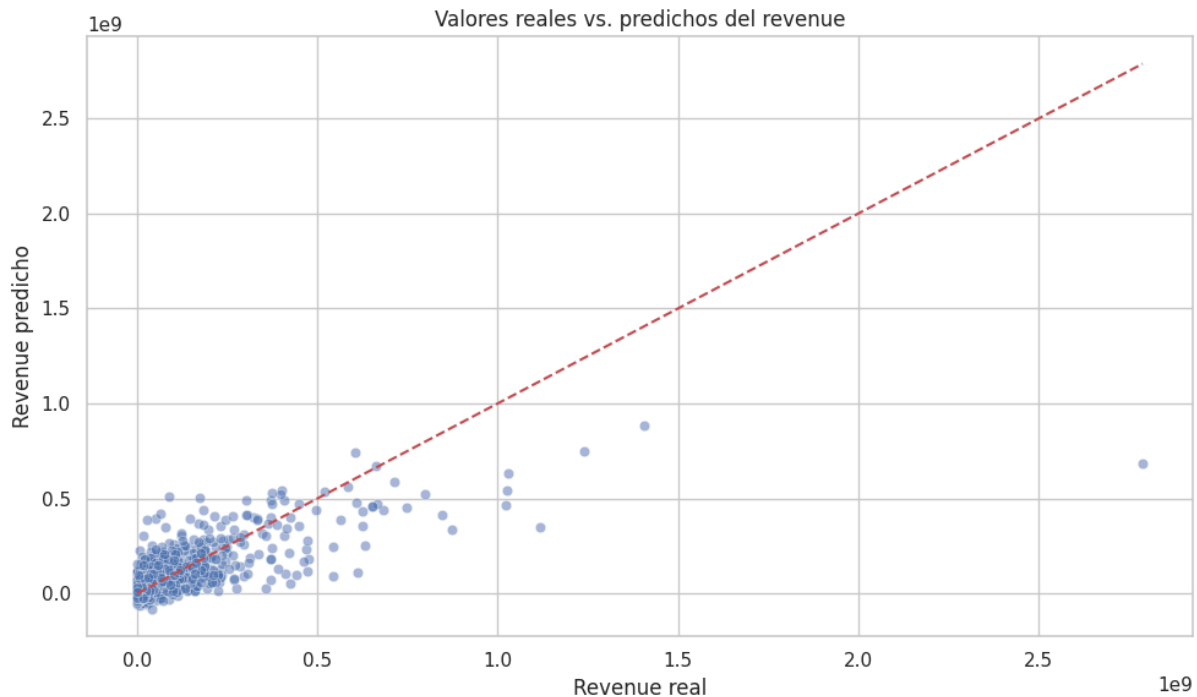
En el caso de la regresión lineal aplicada a la predicción del revenue, se han obtenido los siguientes resultados:

- RMSE (Root Mean Squared Error): \$98.4 millones

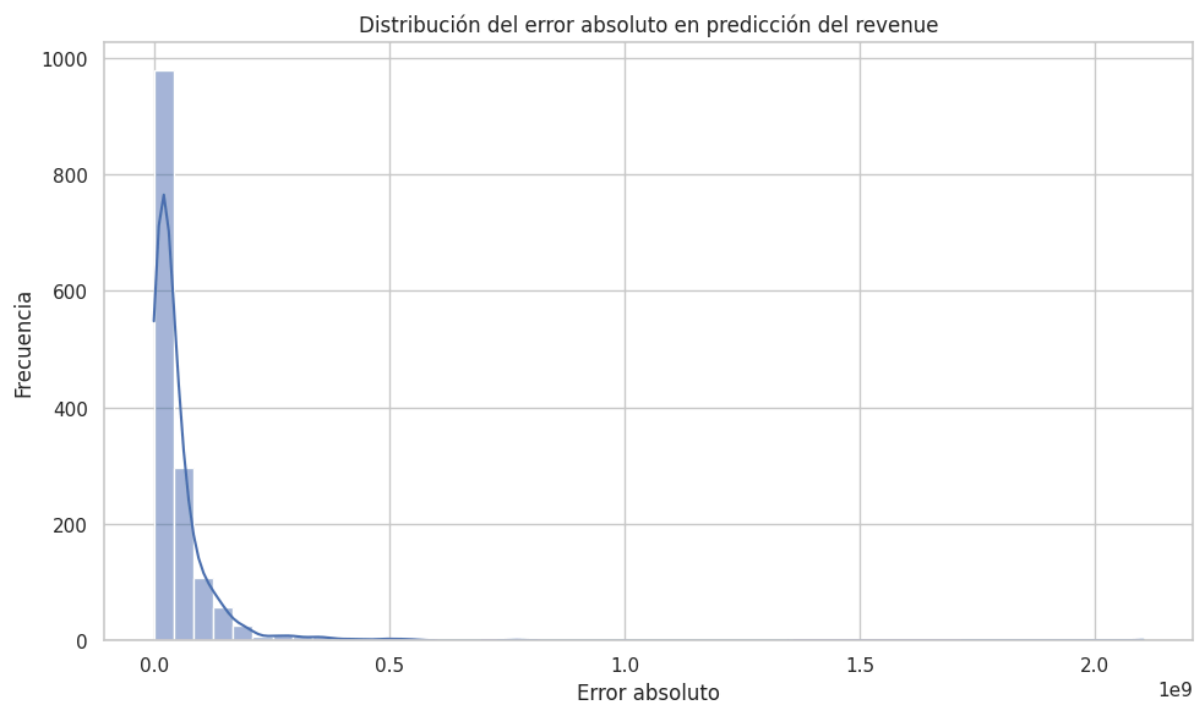
Representa el error cuadrático medio en unidades monetarias. Indica que el modelo, en promedio, se desvía unos 98 millones del valor real. Aunque elevado, es razonable dada la gran dispersión de ingresos en la industria cinematográfica.

- $R^2$  (Coeficiente de determinación): 0.5484

Significa que el modelo es capaz de explicar un 54.84% de la varianza en los ingresos. Para un modelo lineal sin enriquecimiento semántico, es un resultado sólido.



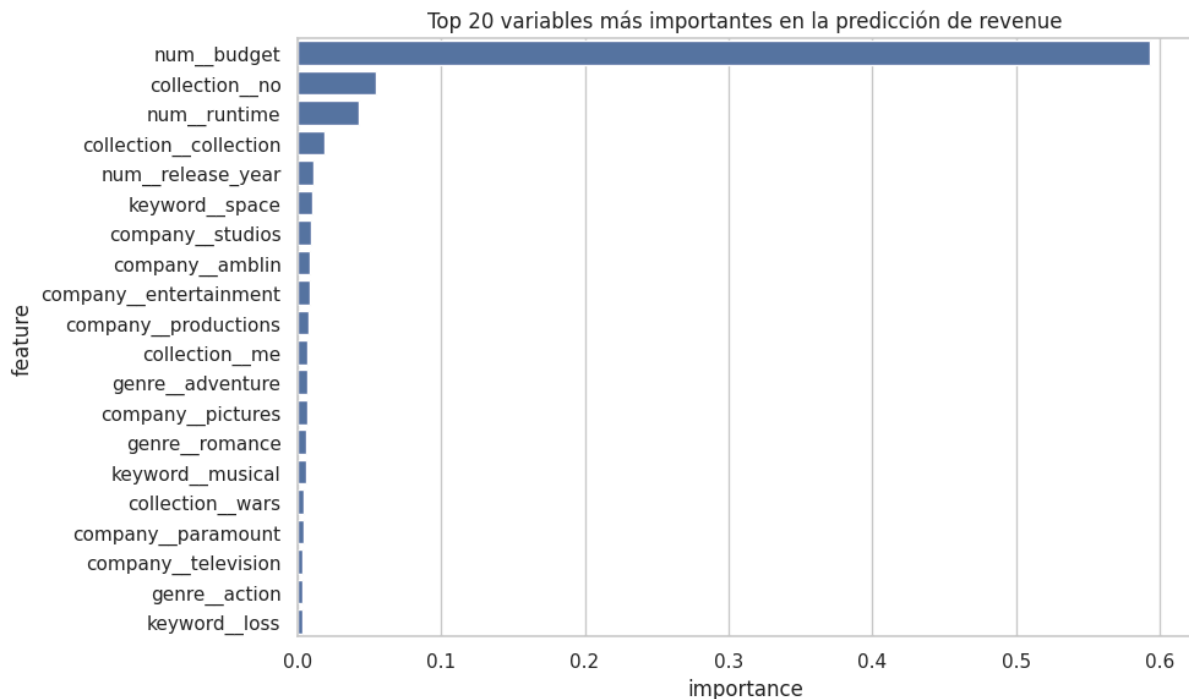
El gráfico de dispersión muestra cómo se alinean las predicciones del modelo respecto a los valores reales. La línea diagonal representa el ajuste perfecto ( $y = x$ ). La mayoría de los puntos se agrupan cerca de la diagonal, indicando un ajuste razonable. Existen errores de predicción más pronunciados en películas con ingresos extremadamente altos.



En este gráfico se representa la frecuencia de los errores absolutos entre las predicciones y los valores reales. La mayoría de los errores son pequeños, lo que confirma que el modelo suele acertar en películas de ingresos medios o bajos. Sin

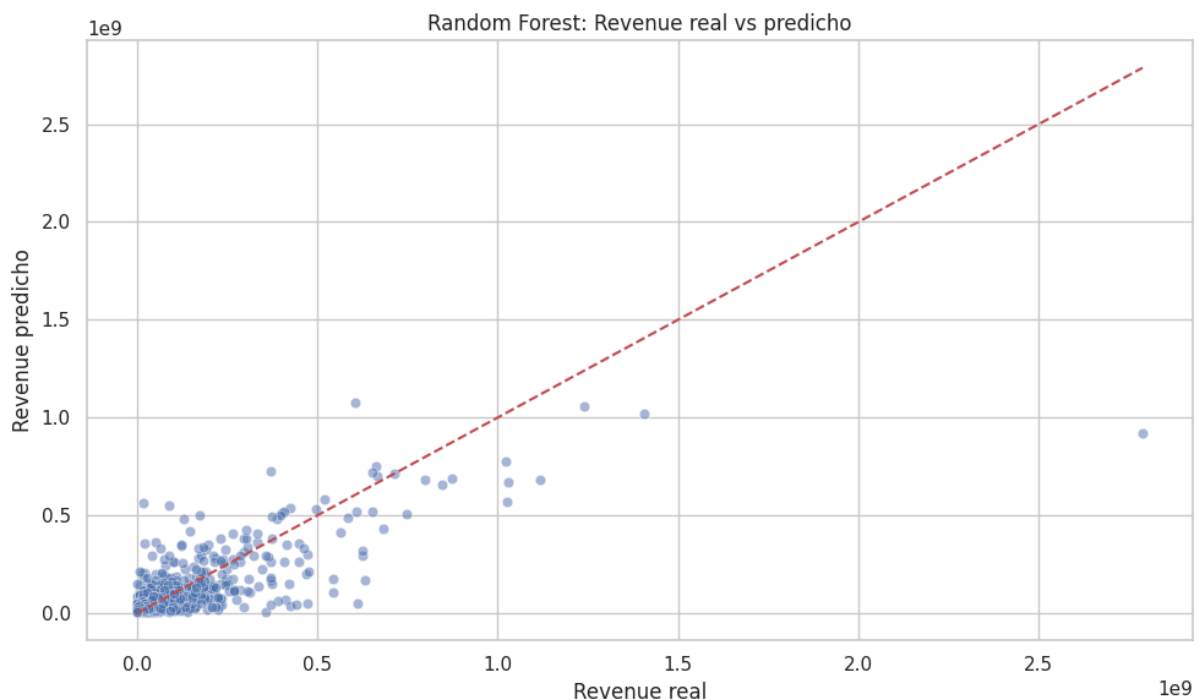
embargo, hay una cola derecha prolongada: casos donde el modelo subestima o sobreestima blockbusters.

## 4.5.2 Random Forest



1. **num\_budget (presupuesto):**  
Es, con diferencia, la variable más importante. Lo esperable: en cine, a mayor inversión, mayor potencial de ingresos.
2. **collection\_no y collection\_collection:**  
Identifican si la película pertenece a una franquicia concreta o no. Las colecciones son predictoras clave del éxito comercial (Marvel, Harry Potter, etc.).
3. **num\_runtime:**  
Aporta contexto sobre la duración: películas muy cortas tienden a recaudar menos.
4. **num\_release\_year:**  
Captura tendencias temporales: más estrenos recientes pueden tener mayor recaudación.
5. **Resto de variables:**  
Palabras clave (keyword\_space, keyword\_musical, etc.) Compañías (company\_amblin, company\_television) y géneros (genre\_romance, genre\_action) tienen un peso mínimo en la predicción. Refleja que el modelo prioriza factores estructurales (presupuesto, saga) frente al contenido.

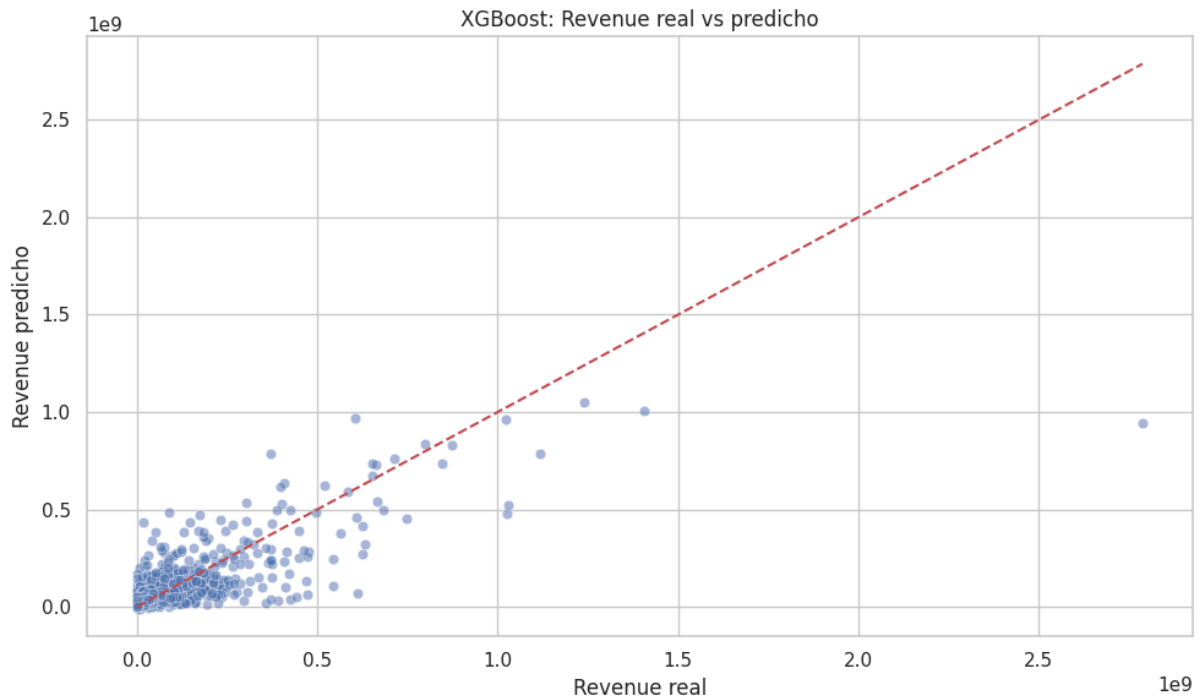
Los resultados muestran un RMSE de \$88.9 millones y un  $R^2$  de 0.6315, datos bastante aceptables teniendo en cuenta las limitaciones del modelo Random Forest.



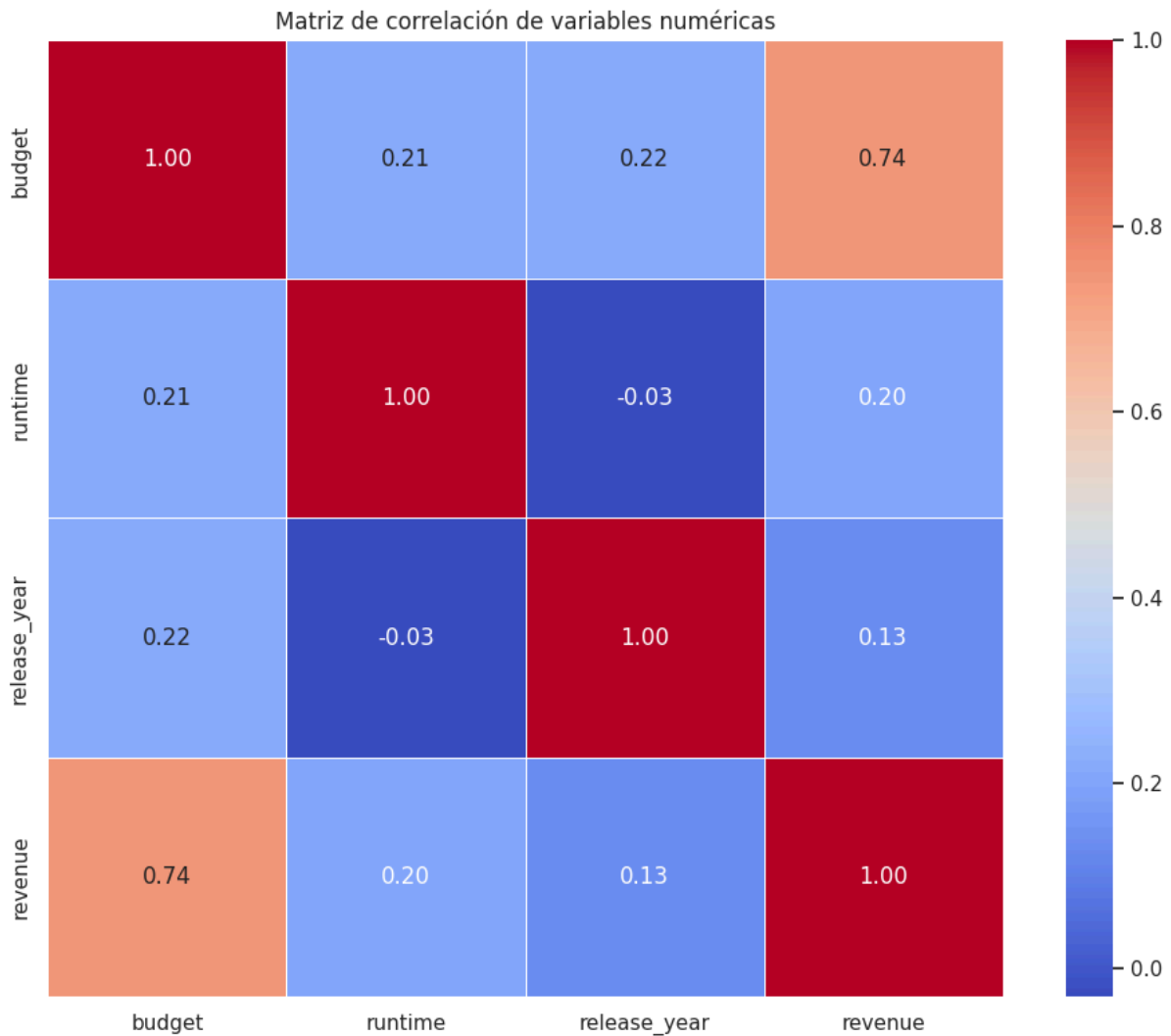
El gráfico muestra que el ajuste global es muy sólido: la nube de puntos se alinea bien con la diagonal ( $y = x$ ). El modelo predice bien la mayoría de las películas dentro de un rango razonable. Sin embargo, algunos casos extremos de revenue alto (blockbusters) aún muestran errores notables, lo cual es natural por su menor frecuencia en los datos, de la misma manera que pasaba en el modelo anterior.

### 4.5.3 XGBoost

El modelo tiene un error medio de  $\pm 88M$ , similar al de Random Forest. La varianza explicada es del 63.63%, lo que lo convierte en el modelo con mejor ajuste hasta ahora, aunque la mejora sobre Random Forest es mínima.



La nube de puntos está bastante bien alineada con la diagonal ( $y = x$ ), especialmente en ingresos bajos y medios. Como en modelos anteriores, el XGBoost subestima algunos blockbusters, aunque el ajuste general es muy bueno. Mejora ligeramente en consistencia en ingresos intermedios, probablemente gracias a su capacidad de regularización y mayor expresividad.



La matriz de correlación confirma lo que ya sabíamos, que la variable budget es, con diferencia, la variable más correlacionada linealmente con revenue. Esto justifica por qué aparece también como la más importante en Random Forest y XGBoost.

## 4.6 Predicción de ingresos para una nueva película

Una vez entrenado y validado el modelo predictivo (en este caso, el algoritmo XGBoost Regressor), es posible utilizarlo para estimar el revenue esperado de nuevas películas, incluso antes de su estreno. Esto resulta especialmente útil para estudios de mercado, planificación presupuestaria o decisiones de inversión.

Se ha construido un ejemplo sintético de película, con los siguientes atributos clave:

```
new_movie = {
    'budget': 100000000,
    'runtime': 120,
    'release_year': 2024,
    'genre_text': 'Action Adventure Sci-Fi',
    'company_text': 'Marvel Studios',
    'keyword_text': 'superhero marvel avengers fight alien',
    'collection_name': 'Avengers Collection'
}
```

El modelo `xgb_model` entrenado previamente ha sido utilizado para predecir el ingreso de esta película ficticia:

```
new_movie_df = pd.DataFrame([new_movie])

predicted_revenue = xgb_model.predict(new_movie_df)

print(f"Predicted revenue: ${predicted_revenue[0]:,.0f}")
```

Predicted revenue: \$331,210,048

El modelo estima que, con estas características, la película generaría más de \$331 millones en ingresos. Esta predicción es coherente con los valores observados en películas de gran presupuesto, pertenecientes a franquicias exitosas (como el universo Marvel). La predicción considera simultáneamente el efecto del presupuesto, la colección, la combinación de géneros y palabras clave asociadas al argumento.

Este tipo de análisis puede ayudar a evaluar decisiones estratégicas antes de rodar o distribuir una película. La fiabilidad de la predicción dependerá de la cobertura y diversidad del conjunto de entrenamiento, y de lo representativo que sea el ejemplo. Si bien este enfoque no reemplaza la opinión humana o el contexto cultural, ofrece una herramienta objetiva complementaria al análisis de negocio.

## 4.7 Predicción de ingresos y valoración a partir de texto (BERT + XGBoost)

El objetivo de este modelo es otra idea que tuvimos de evaluar el potencial comercial y la recepción crítica de una película utilizando solo información textual: su sinopsis (overview), su eslogan publicitario (tagline), los géneros asignados (`genre_text`) y el nombre del director. Esta aproximación resulta especialmente útil en fases tempranas



de desarrollo o promoción, cuando aún no se dispone de datos numéricos como presupuesto o popularidad.

El primer paso fue combinar overview y tagline para construir una única descripción completa. Además de eliminar películas sin texto válido y aquellas sin revenue o vote\_average junto a limitar y filtrar los outliers eliminando películas con ingresos superiores a \$1.000 millones. En adición a lo anterior, se vectorizaron los géneros mediante TF-IDF, con un max\_fratures de 300 y se codificó el director on OneHotEncoder.

El siguiente paso era la codificación semántica con BERT, para lo cual se usó el modelo all-MiniLM-L6-v2 importado de SentenceTransformers. Este modelo es un modelo BERT optimizado para incrustar frases, con el objetivo de generar vectores semánticos a partir de la descripción textual completa de cada película.

```
Python
model_st = SentenceTransformer('all-MiniLM-L6-v2')
text_embeddings = model_st.encode(df_embed['full_description'], show_progress_bar=True)
```

Así, cada sinopsis quedó representada como un vector de 384 dimensiones y podemos pasar a la construcción del modelo predictivo. Para lograr este objetivo, se concatenaron los vectores semánticos (text\_embeddings), el vector de géneros (genre\_vecs) y el director (dir\_vecs) para formar la matriz de entrada final. Se entrenaron dos modelos separados con XGBoost Regressor:

- Uno para predecir el revenue.
- Otro para predecir la vote\_average.

Los parámetros en común fueron:

- n\_estimators=300
- max\_depth=8
- learning\_rate=0.1

En cuanto a los resultados, obtuvimos un RMSE de \$105,9 millones y un  $R^2$  de 0.3431 para la predicción del revenue y un RMSE de 0.9843 (en escala de 0 a 10) junto a un  $R^2$  de 0.1176 para predecir el vote\_average en este caso.

Si interpretamos estos datos, podemos concluir que El modelo consigue capturar más del 34% de la varianza del revenue solo a partir del texto y contexto. En cuanto a la valoración media, el modelo es más limitado, aunque logra reducir el error promedio a 0.98 puntos.

Una vez contamos con este modelo entrenado y con resultados más que razonables, se pudo pasar al siguiente paso y el interesante de este estudio, crear una función `predecir_pelicula_bert()` para estimar `revenue` y `vote_average` a partir de una descripción textual, géneros en texto plano y el nombre del director. Por ejemplo para una película nueva con los siguientes datos:

```
Python
descripcion="A secret agent embarks on a time-bending mission to
prevent global catastrophe.",
genero="Action Sci-Fi",
director="Christopher Nolan"
```

Esta función fue capaz de predecir que la película tendría un `revenue` estimado de \$132.6 millones y una valoración estimada de 6.04/10. El sistema solo por su descripción, detectó que la película encaja con producciones de acción complejas dirigidas por cineastas como Nolan, y ha estimado ingresos y valoración coherentes con ese perfil. Obviamente los resultados de estas predicciones aumentan con más datos, como el presupuesto de la película o incluso la fecha de estreno, pero queríamos demostrar el poder que pueden tener datos tan básicos como la sinopsis o el director de un rodaje.

En conclusión, este enfoque basado únicamente en texto permite realizar predicciones preliminares antes de conocer presupuesto o recepción del público. Además de explorar el impacto del estilo narrativo, el género y el director en el rendimiento de una película y combinarse con modelos colaborativos para construir sistemas de recomendación enriquecidos podrían mejorar los resultados.

## 4.8 Resultados de las predicciones en películas nuevas

Con el objetivo de aplicar los modelos entrenados a casos reales o hipotéticos, se han definido una serie de películas nuevas (no presentes en el dataset original), sobre las cuales se han realizado estimaciones de ingresos utilizando distintos enfoques predictivos. Estas películas han sido las siguientes:

```

new_movie_soul = {
    'budget': 150000000,
    'runtime': 100,
    'release_year': 2020,
    'genre_text': 'Animation Comedy Drama',
    'company_text': 'Walt Disney Pictures Pixar Animation Studios',
    'keyword_text': 'afterlife jazz musician soul pianist purpose',
    'collection_name': 'No Collection'
}

new_movie_parasite = {
    'budget': 11400000,
    'runtime': 132,
    'release_year': 2019,
    'genre_text': 'Drama Thriller',
    'company_text': 'CJ Entertainment Barunson E&A',
    'keyword_text': 'class divide poverty family infiltration manipulation',
    'collection_name': 'No Collection'
}

new_movie_joker = {
    'budget': 55000000,
    'runtime': 122,
    'release_year': 2019,
    'genre_text': 'Crime Drama Thriller',
    'company_text': 'Warner Bros DC Films Village Roadshow Pictures',
    'keyword_text': 'joker mental illness clown society chaos',
    'collection_name': 'No Collection'
}

new_movie_oppenheimer = {
    'budget': 100000000,
    'runtime': 180,
    'release_year': 2023,
    'genre_text': 'Drama History',
    'company_text': 'Syncopy Universal Pictures',
    'keyword_text': 'manhattan project atomic bomb scientist war biography',
    'collection_name': 'No Collection'
}

```

Para el caso del revenue los resultados han sido los siguientes:

Película	Regresión Lineal	Random Forest	XGBoost
Soul	\$457,422,162	\$601,444,828	\$568,395,392
Parásitos	\$20,370,788	\$28,078,551	\$19,250,160
Joker	\$99,223,389	\$108,428,485	\$146,585,488
Oppenheimer	\$272,840,989	\$241,682,976	\$268,675,296

En general, XGBoost tiende a producir predicciones más optimistas que Linear Regression, y con mayor precisión según el rendimiento previo observado. Películas de gran presupuesto como Soul y Oppenheimer reciben ingresos estimados superiores a \$250M, en línea con lo esperado para producciones de alto nivel técnico y con estudios reconocidos detrás. Parásitos, aunque de bajo presupuesto, obtiene ingresos relativamente altos en Random Forest, lo que podría deberse a la importancia del contenido y crítica social en su éxito, aunque sin ser reflejado completamente en los datos estructurales. En el caso de Joker, los tres modelos coinciden en estimaciones superiores a los \$100M, concordando con su perfil de éxito comercial.

Para las predicciones a partir de texto con BERT, se han introducido estas descripciones de las películas:

```
JSON
{
  "Soul": {
    "descripcion": "A jazz musician, stuck in a mediocre job, finally gets his big break but finds himself transported out of his body and must find his way back with the help of an infant soul.",
    "genero": "Animation. Comedy. Drama. Jazz. Afterlife. Soul. Pixar",
    "director": "Pete Docter"
  },
  "Parásitos": {
    "descripcion": "Greed and class discrimination threaten the newly formed symbiotic relationship between the wealthy Park family and the destitute Kim clan.",
    "genero": "Drama. Thriller. Satire. Social Class. Family. Korean Cinema",
    "director": "Bong Joon-ho"
  },
  "Joker": {
    "descripcion": "In Gotham City, mentally troubled comedian Arthur Fleck embarks on a downward spiral of social revolution and bloody crime to become the infamous criminal known as Joker.",
    "genero": "Crime. Drama. Thriller. Mental Illness. Society. DC Comics",
    "director": "Todd Phillips"
  },
  "Oppenheimer": {
    "descripcion": "In wartime, the brilliant American physicist J. Robert Oppenheimer leads the Manhattan Project to build the atomic bomb. Shocked by its destructive power, he questions the moral consequences.",
    "genero": "Drama. History. Biography. World War II. Nuclear. Scientists",
    "director": "Christopher Nolan"
  }
}
```

Este modelo demuestra capacidad para diferenciar entre temáticas de alto impacto comercial (historia, conflictos bélicos, ciencia) y otras más nicho como cine independiente o biográfico. Estos han sido los resultados:

Película	Revenue estimado (BERT)	Valoración media (BERT)
Soul	\$128,880,728	6.75 / 10
Parásitos	\$105,552,488	6.48 / 10
Joker	\$113,180,336	6.49 / 10
Oppenheimer	\$324,343,392	7.01 / 10

Es verdad que viendo los resultados el modelo BERT tiende a subestimar ligeramente el revenue real para películas con gran éxito (como Soul o Joker), probablemente por no tener acceso a datos como presupuesto o productora. A pesar de ello, ofrece predicciones coherentes y razonables a partir de texto puro, lo cual es especialmente útil en fases tempranas de producción o marketing. En cuanto a valoración, las predicciones oscilan entre 6.4 y 7.4, con puntuaciones lógicas que reflejan el impacto crítico o el atractivo del argumento y estilo narrativo.

## 4.9 Conclusiones del análisis predictivo

A lo largo de esta sección se han desarrollado distintos modelos de predicción con el objetivo de estimar dos variables clave en la industria cinematográfica: los ingresos económicos (revenue) y la valoración media del público (vote\_average). Para ello, se han utilizado tanto enfoques clásicos de regresión como modelos más avanzados de aprendizaje automático y procesamiento del lenguaje natural.

Los resultados obtenidos permiten extraer las siguientes conclusiones generales:

- El presupuesto (budget) es, con diferencia, la variable estructural más correlacionada con los ingresos, lo cual es esperable dado que las grandes producciones suelen contar con estrategias comerciales más potentes. Esto se ha confirmado en todos los modelos basados en datos estructurados.
- Modelos como XGBoost han ofrecido el mejor rendimiento general, explicando más del 63% de la varianza de los ingresos y superando a Random Forest y Regresión Lineal. Estos modelos resultan especialmente eficaces cuando se dispone de todas las variables (presupuesto, duración, año, etc.).
- En escenarios donde solo se dispone de información textual previa al estreno (como sinopsis, género y director), el modelo BERT + XGBoost ha demostrado ser una alternativa viable. Aunque su rendimiento es algo más limitado, ha logrado capturar hasta un 34% de la varianza de ingresos únicamente a partir del contenido narrativo.
- Al aplicar los modelos a películas reales como Soul, Joker o Parásitos, las predicciones han sido razonables y coherentes con el perfil de cada producción,

mostrando que los modelos generalizan bien incluso fuera del conjunto de entrenamiento.

- El enfoque semántico basado en BERT permite estimar tanto ingresos como valoración de forma preliminar, lo que lo convierte en una herramienta útil para evaluaciones tempranas de guiones o campañas promocionales, antes incluso de definir el presupuesto o el reparto completo.

En definitiva, este bloque ha demostrado que es posible construir modelos predictivos con un grado aceptable de precisión a partir de datos históricos, y que la incorporación de técnicas de NLP y representaciones semánticas permite ampliar las capacidades del análisis más allá de las variables estructuradas tradicionales. Estas herramientas pueden ser de gran utilidad para la toma de decisiones estratégicas en el ámbito del cine y el entretenimiento.

## 5. Sistemas de recomendación

### 5.1 Introducción a los sistemas de recomendación

Un sistema de recomendación es un sistema que proporciona sugerencias sobre elementos (o acciones) que, dentro de un dominio, pueden ser interesantes para el usuario [Resnik & Varian].

Los sistemas de recomendación han cobrado gran importancia desde hace un década con el auge de las plataformas de contenido en streaming como Netflix o Spotify y redes sociales como Tik Tok o Youtube. Estos servicios basan gran parte de su modelo de negocio en la retención de usuarios a los que sirven contenidos multimedia. Disponer de un buen sistema de recomendación es clave para poder mejorar esta retención, además de aportar un grado de calidad adicional al usuario.

Aunque en nuestro caso estamos trabajando con películas, un sistema de recomendación se puede aplicar a otros muchos servicios:

- Compra de artículos.
- Visualización o acceso.
- Valoración explícita por medio de votos.

Hay tres estrategias principales a la hora de crear un sistema de recomendación:

- **Basadas en contenido:** se nos recomiendan artículos parecido a los que le han gustado anteriormente.
- **Basadas en filtrado colaborativo:** se nos recomiendan artículos que les han gustado a usuarios similares, según las compras o votos dados por nosotros.
- **Híbridos:** combinan ambas estrategias.

También hay diferencias en cómo se puede implementar un modelo:

- **Basados en memoria:** Utilizan toda la matriz de votos para identificar usuarios o ítems con patrones de votos similares y luego recomendar.
- **Basados en modelo:** Utilizan técnicas de aprendizaje automático para construir un modelo abreviado, identificando patrones de comportamiento, que luego se empleará para realizar predicciones.

### 5.2 Redes neuronales para sistemas de recomendación

Una red neuronal artificial es un modelo computacional inspirado en el funcionamiento del cerebro humano, diseñado para reconocer patrones complejos en datos. Está compuesta por unidades básicas llamadas neuronas, organizadas en capas interconectadas. Cada neurona recibe una serie de entradas numéricas, aplica una transformación lineal ponderada, y posteriormente una función de activación no lineal que permite a la red modelar relaciones complejas y no triviales.

Las redes neuronales se estructuran generalmente en tres tipos de capas: la capa de entrada, que recibe los datos iniciales; una o varias capas ocultas, donde se realiza el procesamiento intermedio; y la capa de salida, que produce el resultado final (por ejemplo, una clasificación o una predicción numérica). Durante el entrenamiento, los pesos de las conexiones entre neuronas se ajustan mediante un algoritmo de optimización, típicamente el descenso del gradiente, con el fin de minimizar el error entre las predicciones y los valores reales.

Este tipo de modelo se ha convertido en una herramienta fundamental en la inteligencia artificial moderna, gracias a su capacidad de aprendizaje automático, adaptabilidad a distintos tipos de datos y rendimiento en tareas que van desde la visión por computador hasta la generación de lenguaje natural. En el contexto de los sistemas de recomendación, las redes neuronales permiten construir modelos que van más allá de la simple coincidencia entre usuarios y elementos, incorporando contextos, descripciones y relaciones no lineales entre variables.

Tradicionalmente, los sistemas de recomendación se han apoyado en técnicas de filtrado colaborativo, como el uso de matrices de utilidad y algoritmos basados en la similitud entre usuarios o productos. Sin embargo, este enfoque presenta limitaciones significativas, como la dificultad para tratar con usuarios nuevos (problema del cold start), la falta de contexto semántico en las recomendaciones y la incapacidad de aprovechar descripciones ricas del contenido.

El uso de redes neuronales en recomendación surge como respuesta a estas limitaciones. Gracias a su capacidad para aprender representaciones latentes de alta calidad y su flexibilidad para combinar múltiples tipos de entrada (numéricos, categóricos y textuales), las redes neuronales permiten integrar tanto las interacciones históricas entre usuarios y productos como la información intrínseca de los ítems (por ejemplo, género, sinopsis, año, director).

## 5.3 Sistemas de recomendación implementados

Se han implementado 3 sistemas de recomendación distintos basados en distintos modelos y utilidades disponibles. Todos los sistemas siguen un enfoque híbrido dado la naturaleza del problema y de los datos disponibles.

### 5.3.1 Sistema con el módulo surprise y SVD

El módulo scikit-surprise es un módulo de Python para crear sistemas de recomendación. Este módulo ofrece distintos algoritmos y posibilidad de personalización del modelo implementado.



Para este modelo se hace una evaluación inicial de distintos algoritmos para elegir el mejor resultado:

- **SVD:** RMSE = 0.9014
- **KNNBasic:** RMSE = 0.9841
- **NMF:** RMSE = 0.9461

Tras esto se hace una búsqueda de mejores parámetros para SVD, mejorando así el RMSE a 0.8962.

Este algoritmo se basa en la factorización de matrices y trata de modelar la matriz de ratings **usuario** × **ítem** como la multiplicación de matrices más pequeñas, es decir:

$$R \approx U \cdot V^T$$

En el sistema de recomendación se pretende factorizar **R** que es la matriz de ratings. En esta matriz, las filas son los usuarios, las columnas son los ítems y cada elemento son los ratings.

Una de las desventajas de este sistema es que es puramente colaborativo y no tienen en cuenta datos como género o director. Para solventar esto se ha añadido una sección adicional para poder tener en cuenta estos datos. Así el sistema de recomendación funciona de la siguiente manera:

- Crea un vector que describe el contenido de la película.
- Crea vector que representa el perfil de contenido del usuario.
- Se calcula cuánto se parece cada ítem candidato al perfil de contenido del usuario.
- Para cada película candidata se predice qué rating le daría el usuario (basado en el histórico de ratings de otros usuarios).

El algoritmo tiende a funcionar de la siguiente manera: si, por ejemplo, a un usuario le han gustado comedias de Woody Allen sus recomendaciones estarán orientadas a estos gustos.

### 5.3.2 Sistema basado en redes neuronales con PyTorch

Además del sistema implementado con surprise, se ha desarrollado un modelo de recomendación propio utilizando esta vez redes neuronales artificiales con la librería PyTorch. El objetivo principal de este sistema es predecir la puntuación que un usuario daría a una película a partir únicamente de las interacciones previas, siguiendo el enfoque de **filtrado colaborativo neuronal**.

Este tipo de modelo es especialmente útil cuando se dispone de un histórico suficientemente amplio de valoraciones y permite aprender patrones complejos entre usuarios y películas gracias a la capacidad de representación de las redes neuronales. La arquitectura propuesta sigue la lógica del modelo conocido como **Neural Collaborative Filtering (NCF)**, y se estructura en las siguientes capas:

- **Embeddings:** Se generan vectores latentes para cada usuario y cada película mediante capas de embedding. Estos vectores actúan como representaciones abstractas que resumen las características de preferencia (usuarios) y contenido (películas).
- **Capa oculta:** Los vectores de usuario y película se concatenan y se introducen en una red neuronal de tipo MLP (Multilayer Perceptron), con varias capas densas y activación ReLU, que permite capturar relaciones no lineales.
- **Capa de salida:** Finalmente, se utiliza una neurona con activación lineal para predecir la puntuación que el usuario daría a esa película.

Para entrenar la red se ha utilizado el conjunto `ratings_small.csv`, que contiene registros del tipo (userId, movieId, rating). Las etapas de preparación han sido:

- Codificación de los identificadores con LabelEncoder para obtener índices enteros.
- División del conjunto de datos en entrenamiento y validación (80/20).
- Creación de objetos TensorDataset y uso de DataLoader para facilitar el procesamiento por lotes.

Para el entrenamiento, se ha usado la función de pérdida (MSE o error cuadrático medio), ya usado en modelos de predicciones anteriores, el optimizado Adam con un learning rate de 0.001, un batch size de 64 y 10 épocas. Durante el entrenamiento se ha observado una convergencia estable del error, y una mejora progresiva en la capacidad predictiva del modelo.

El modelo final alcanzó un **MAE** cercano a 0.68, lo que significa que el error medio en las predicciones fue inferior a un punto sobre una escala de 0 a 5. Este resultado es bastante competitivo teniendo en cuenta que el modelo sólo se alimenta de los identificadores y no incluye información semántica adicional (como sinopsis o géneros).

Una vez entrenado, este modelo permite predecir la puntuación que un usuario daría a una película aún no vista, además de generar recomendaciones ordenadas según puntuación estimada y filtrar películas ya valoradas por el usuario para ofrecer solo contenido nuevo.

### 5.3.3 Sistema de recomendación con LightFM

LightFM es una implementación en python de un conjunto de algoritmos de recomendación. Esta biblioteca implementa Factorization Machines, un tipo de algoritmo de aprendizaje supervisado que se puede utilizar, entre otras aplicaciones, para ranking.

El modelo implementado con esta librería funciona de la siguiente manera:

- Se crea una matriz de interacciones de tipo binaria. Si el usuario dio rating  $\geq 4$  se considera que le gustó la película e interacción = 1. Si no, interacción = 0.
- Para cada película se crea una *feature* (una cadena de texto) con director, género y palabras clave, y luego se vectoriza con TF-IDF. Así, para cada ítem, se tiene un vector numérico de features. Esto permite que LightFM tenga embeddings de ítems que reflejen su contenido, además de su comportamiento en ratings.
- LightFM aprende embeddings, tanto de usuarios como de contenido.
- Como función de optimización se usa WARP (Weighted Approximate-Rank Pairwise loss) que directamente para un ranking top-N. Con esto, se pretende maximizar la probabilidad de que los ítems con interacción positiva estén en las primeras posiciones del ranking

## 5.4 Resultados y comparación de los sistemas de recomendación

La ejecución y evaluación de los sistemas de recomendación se ha hecho mediante la creación de una serie de usuarios aleatorios y someter a evaluación las recomendaciones de los sistemas.

A la hora de evaluar el modelo se usan medidas típicas:

- **Precision@10**: porcentaje de ítems relevantes en el top-10.
- **Recall@10**: porcentaje de ítems relevantes que aparecen en el top-10.
- **RMSE**: raíz cuadrada del error cuadrático medio. Mide cómo de lejos están las predicciones de los valores reales, en promedio

Los resultados han sido los siguientes

Sistema	Precision@10	Recall@10	RMSE
Surprise	0.0000	0.0000	0.8291
PyTorch	0.0000	0.0000	0.2883
LightFM	0.2600	0.1121	—

Lo primero que llama la atención de estos resultados es la nulidad o falta de datos en algunos sistemas frente a otros. Esto se debe a que tanto Surprise como PyTorch tratan de predecir la calificación que un usuario dará a una película y en base a esto hacer la recomendación, teniendo en cuenta a usuarios parecidos al evaluado. Es decir, tratan de predecir el **Rating**. Sin embargo, LightFM lo trata de ordenar las películas para cada usuario y dar un ranking con las mejores primero. Esta condición de mejores se calcula según qué valoraciones le dieron otros usuarios similares.

Por este motivo, no tiene sentido calcular valores de RMSE para LightFM ni los otros dos sistemas dan puntuación en Precision ni Recall. Es por esto que, en este caso, se tendrían que evaluar estos modelos en base a la experiencia de los usuarios o en base a conocimiento experto.

Sí se puede decir que el sistema de PyTorch ha dado un mejor resultado que el de Surprise, aunque ambos son bastante buenos.

## 6. Conclusiones del proyecto

A lo largo de este trabajo se han abordado dos grandes líneas de análisis con un enfoque complementario: por un lado, la predicción del éxito de una película antes de su estreno, y por otro, la recomendación personalizada de contenido a usuarios concretos.

En la parte de análisis predictivo, se han explorado modelos clásicos (regresión lineal), árboles de decisión (Random Forest, XGBoost), y modelos semánticos basados en BERT. Estos modelos han permitido estimar valores como la recaudación (revenue) o la puntuación media (vote\_average) a partir de distintas variables, incluyendo tanto metadatos estructurados como descripciones textuales. Entre ellos, XGBoost ha sido el modelo más robusto al trabajar con datos estructurados, mientras que BERT ha demostrado utilidad para realizar estimaciones tempranas a partir únicamente de texto, aunque con mayor margen de error.

Respecto a los sistemas de recomendación, es posible decir con certeza que PyTorch es mejor que Surprise. Sin embargo, es difícil de decir qué modelo es el que mejor funciona. No sólo porque sigan estrategias de recomendación ligeramente distintas, aunque ambos sean modelos híbridos, sino porque, al estar hablando de recomendación de películas, la subjetividad de los usuarios tiene un papel importante.

Como punto adicional, a la hora de construir un sistema de recomendación es importante disponer de un dataset de reseñas lo suficientemente grande, pues si no, los sistemas de recomendación no son capaces de poder dar una respuesta lo suficientemente personalizada para cada usuario, un problema del que adolece especialmente Surprise. Se disponía de él, pero nuestros sistemas de cómputo no han dado la talla.

## 7. Bibliografía

- Material de teoría de la asignatura
- Material de teoría de la asignatura de Inteligencia Computacional
- Material de teoría de la asignatura de Gestión de Información en la Web
- User Guide. (s. f.). Scikit-learn. —  
[https://scikit-learn.org/stable/user\\_guide.html](https://scikit-learn.org/stable/user_guide.html)
- Pandas documentation — pandas 2.3.0 documentation. (s. f.).  
<https://pandas.pydata.org/docs/>
- NumPY Documentation. (s. f.). — <https://numpy.org/doc/>
- Factorization Machines —  
<https://ieeexplore.ieee.org/document/5694074>
- XGBoost Documentation — xgboost 3.0.2 documentation. (s. f.).  
<https://xgboost.readthedocs.io/en/stable/>
- PyTorch documentation — PyTorch 2.7 documentation. (s. f.).  
<https://docs.pytorch.org/docs/stable/index.html>
- Hug, N. (s. f.). Home. Surprise. <https://surpriselib.com/>
- Welcome to LightFM's documentation! — LightFM 1.16 documentation.  
(s. f.). <https://making.lyst.com/lightfm/docs/home.html>
- Matplotlib documentation — Matplotlib 3.10.3 documentation. (s. f.).  
<https://matplotlib.org/stable/index.html>
- SentenceTransformers Documentation — Sentence Transformers  
documentation. (s. f.). <https://sbert.net/>