

*Personalización de preferencias*  
**Perfiles de usuarios**  
**en base a preferencias positivas y negativas**

**GRUPO 14:**

**4260487, DAMIANO IZZI RODRIGO SEBASTIAN**  
**3124845, GONZALEZ CARABAJAL ANA VIRGINIA**  
**4130833, LOPEZ LOPEZ MARCELO DANIEL**  
**4803794, TECHERA GÓMEZ TAMARA DALMAR**

# Tabla de contenido

[Introducción](#)

[Objetivos](#)

[Sistema de Recomendaciones](#)

[Tipos de Sistemas de Recomendación](#)

[Sistemas de recomendación Basados en Contenido](#)

[Sistemas de Recomendación Colaborativos](#)

[Sistemas de recomendación Basados en Conocimiento](#)

[Híbrido](#)

[Utilidad de los Sistemas de Recomendación](#)

[Realimentación en Sistemas de Recomendación](#)

[Elección del conjunto de datos para el Sistema de Recomendación](#)

[Ejemplos de Sistemas de Recomendaciones](#)

[Sistemas de Recomendación Colaborativos](#)

[Categorías de los algoritmos de Filtrado Colaborativo](#)

[Ejemplos reales de Sistemas de Recomendación Colaborativos](#)

[Sobre el Sistema](#)

[Conjunto de datos](#)

[Datos utilizados](#)

[Descripción de las tablas](#)

[Requerimientos funcionales de la aplicación](#)

[Diseño del sistema](#)

[Arquitectura](#)

[Herramientas utilizadas](#)

[Algoritmos utilizados](#)

[Algoritmo basado en usuarios](#)

[Similitud](#)

[Correlación de Pearson](#)

[Cálculo de predicción de puntajes: Average + adjustment](#)

[Votos positivos y negativos](#)

[Weighed Sum](#)

[Resultados](#)

[Trabajo a futuro](#)

[Conclusión](#)

[Referencias](#)

## Introducción

El uso de perfiles, es un componente fundamental para la personalización de cualquier aplicación. La mayoría de los sistemas se basan en objetos que al usuario le interesan (preferencias positivas) pero no en los objetos que no le interesan (preferencias negativas).

Este informe se centrará en la personalización de preferencias, a partir de perfiles de usuarios en base a preferencias positivas y negativas. Utilizar las preferencias negativas, puede incrementar la separación entre las consultas similares y disímiles.

La formulación más común del problema de recomendación se basa en la noción de ratings, es decir, los sistemas de recomendación estiman ratings de ítems o productos que aún no se han consumido por los usuarios, en base a las votaciones de los elementos ya consumidos.

En el presente informe se proponen diferentes técnicas que utilizan perfiles de usuario, que explotan las preferencias de usuario negativas y positivas para producir ese perfil.

Se implementará un sistema, el cual toma en cuenta a los usuarios con gustos semejantes al usuario que solicita la recomendación.

Se utilizan los datos que proporciona la web <http://www.grouplens.org>, para realizar estudios de investigación en el ámbito de los sistemas de recomendaciones colaborativos.

Los datos están en formato texto por lo que hay que convertirlos a un formato de base de datos manejable. Para esto, utilizamos una base de datos MySQL, y se implementó una aplicación en lenguaje Java, en la que se evalúan y comparan distintos algoritmos de filtrado colaborativo basados en usuario.

## Objetivos

Investigar sobre las técnicas existentes de recomendaciones por preferencias negativas, junto a la utilización de diferentes algoritmos de filtrado colaborativo.

Se implementarán y compararán dos técnicas, en las que toman en cuenta las preferencias negativas, y se evaluará el impacto de la inclusión de preferencias negativas en los algoritmos de filtrado colaborativo. Para ello, se implementará un sistema de recomendación con una arquitectura cliente/servidor y con una interfaz web.

## Sistema de Recomendaciones

El ser humano necesita información para tomar decisiones de cualquier tipo, pero muchas veces se encuentra que la información que tiene disponible es demasiado amplia o inconexa, entonces se tiene una sobrecarga de información y es difícil de extraer la que es verdaderamente relevante. Es necesario filtrar esta información relevante diseminada en grandes volúmenes de información, y es en esta situación donde se puede aplicar los sistemas de recomendación.

Formalmente, un sistema de recomendación, tiene como principal tarea seleccionar ciertos objetos de acuerdo a los requerimientos del usuario.

El sistema utiliza las opiniones de los usuarios de una comunidad para ayudar a usuarios de esa comunidad a encontrar contenidos de su preferencia entre un conjunto sobrecargado de posibles elecciones.

Estos sistemas necesitan una enorme cantidad de información sobre usuarios y objetos para poder realizar recomendaciones de calidad. También se tienen, los sistemas de recomendación híbridos, los cuales recogen los mejores aspectos de dos o más de los sistemas de recomendación para conseguir resultados todavía mejores a la hora de realizar sus recomendaciones.

### Tipos de Sistemas de Recomendación

- **Sistemas de recomendación Basados en Contenido**

Son aquellos en el cual las recomendaciones son realizadas basándose solamente en un perfil, creado a partir del análisis del contenido de los objetos que el usuario ha evaluado en un pasado. Extraen características de los objetos y las comparan con el perfil del usuario para predecir las preferencias de los usuarios sobre tales objetos.

- **Sistemas de Recomendación Colaborativos**

Son aquellos en los que las recomendaciones se realizan basándose solamente en los términos de similitud entre los usuarios de intereses similares. Para obtener recomendaciones de calidad es necesario utilizar un buen algoritmo de filtrado colaborativo. Estos algoritmos se pueden agrupar en dos categorías: los algoritmos basados en *memoria o usuario* y los basados en *modelos o ítem*.

- **Sistemas de recomendación Basados en Conocimiento**

Realizan inferencia entre las necesidades y preferencias de cada usuario para sugerir recomendaciones. A diferencia de otros sistemas de recomendación, no dependen de grandes cantidades de información sobre objetos puntuados (basados en contenidos) y usuarios particulares (colaborativos) sino que lo único que necesitan es tener un conocimiento informal de las necesidades del usuario.

- **Híbrido**

Los sistemas híbridos entre: los basados en contenido y los colaborativos, guardan las preferencias del usuario y las combinan con los objetos más relevantes para realizar las recomendaciones. También existen los sistemas híbridos entre:

- los basados en conocimiento y los colaborativos
- los basados en contenido y conocimiento
- los colaborativos y redes sociales.

## **Utilidad de los Sistemas de Recomendación**

Los sistemas de recomendación resultan de vital importancia para el marketing personalizado, ya que reducen el tiempo de búsqueda de los productos, consiguen con mayor efectividad las búsquedas y, por lo tanto, una mayor satisfacción en los clientes.

Para lograr con los objetivos, todos los sistemas de recomendación llevan a cabo dos tareas:

- **Predecir:** predecir una serie de objetos, servicios o productos en los que un usuario o cliente particular podría estar interesado.
- **Recomendar los N-mejores objetos:** identifican los N objetos en los que el usuario estará más interesado.

## **Realimentación en Sistemas de Recomendación**

Un sistema de recomendación no debe ser una entidad estática, sino que tiene que evolucionar en el tiempo en cuanto a la calidad de sus recomendaciones y pronóstico en base a la experiencia y nueva información adquiridas.

Para lograr este objetivo se utilizan mecanismos de realimentación entre el sistema y los gustos de los usuarios. Existen dos tipos de mecanismos de retroalimentación: los *implícitos* y los *explícitos*.

- **Realimentación implícita:** el mecanismo de realimentación implícito proporciona información al sistema de recomendación acerca de las preferencias de los usuarios, sin que éstos sean conscientes de esta situación. Por lo tanto este tipo de realimentaciones no son directas, sino que se realizan mediante diversos tipos de medidas como pueden ser: el tiempo de visualización del objeto, el número de veces que el objeto es solicitado, etc.

Esta realimentación tiene el problema de depender demasiado del contexto y de ser excesivamente hipotética (podemos suponer que solicitar la visualización de un objeto muchas veces indica un especial interés por parte del usuario, pero no tiene porqué ser de

esa manera) por lo que no resulta ser la más apropiada para todas las situaciones de recomendación.

- **Realimentación explícita:** un mecanismo con este tipo de realimentación se basa en la acción directa por parte del usuario para indicar que objetos determinados del sistema son de su interés. Esta interacción directa se puede realizar mediante votaciones numéricas o, más sencillo aún, que el usuario diga si el objeto es no de su agrado.

## **Elección del conjunto de datos para el Sistema de Recomendación**

La elección de un adecuado conjunto de datos es algo importante para evaluar la calidad de un sistema de recomendación.

- **Análisis online u offline**

Es importante definir cómo se va a trabajar con los datos: sí de manera online u offline. En el análisis offline se usa una técnica o algoritmo para predecir ciertos valores retenidos de un conjunto de datos y los resultados son analizados mediante una o varias métricas de error. Este análisis tiene la ventaja de ser rápido y económico pero también tiene las desventajas importantes: el problema de la escasez de datos y el problema de obtener sólo como resultado la bondad de la predicción.

El análisis online, permite obtener otros resultados como son la actuación de los usuarios participantes, su satisfacción o su participación. En su defecto es más lento y caro que el análisis offline.

- **Datos reales o sintetizados**

Otra opción sería elegir entre un conjunto de datos reales (recopilados de usuarios reales sobre objetos reales) o un conjunto de datos sintetizados (creados específicamente para el sistema de recomendación, sin ninguna base real). Los datos sintéticos son más fáciles de crear que los reales, ya que no hay que realizar encuestas ni otros métodos para conseguirlos del mundo real, sin embargo es recomendable usar estos últimos y sólo utilizar los sintetizados en las primeras fases de desarrollo del sistema, siendo sustituidos por los reales cuando haya un número importante recopilados.

## Ejemplos de Sistemas de Recomendaciones

- Amazon: Es un sistema de recomendación que mezcla los enfoques basados en contenido y colaborativo. El sistema guarda las preferencias del usuario activo y las combina con objetos relevantes para generar recomendaciones.
- IMDB Recommendation Center: Este sistema de Recomendación está basado en contenido. El usuario introduce la película o show televisivo que más le guste y el sistema ofrece una lista con diez recomendaciones. Como método de realimentación o feedback con el sistema, el usuario puede señalar las recomendaciones con las que no esté de acuerdo y proponer recomendaciones nuevas con lo que el algoritmo se va depurando con la interacción del usuario.

## Sistemas de Recomendación Colaborativos

Un sistema de recomendación colaborativo es aquel en que las recomendaciones se realizan basándose solamente en los términos de similitud entre los usuarios. Estos sistemas, recomiendan objetos que son preferencia de otros usuarios de intereses similares, en vez de recomendar objetos similares a los que le gustaban en un pasado al usuario activo como sucedía con los basados en *contenido*. Este tipo de sistema, se basa en el concepto del *boca a boca* entre usuarios para realizar las recomendaciones.

La base teórica de estos sistemas es bastante sencilla: se forman grupos de usuarios más cercanos, es decir, aquellos cuyos perfiles son más parecidos y a un usuario de un grupo se le recomienda objetos que él no tenga puntuados pero que tengan puntuaciones positivas por parte del resto de los usuarios de ese grupo, es decir, de los más similares a él.

Existen tres pasos fundamentales en el funcionamiento de los sistemas colaborativos:

1. El sistema guarda un perfil para cada usuario con evaluaciones sobre objetos conocidos por él y que pertenezcan a la base de datos sobre la que se trabaje.
2. Se mide el grado de similitud entre los diferentes usuarios del sistema en base a sus perfiles y se crean grupos de usuarios con características afines.
3. El sistema utilizará toda la información obtenida en los pasos anteriores para realizar las recomendaciones. A cada usuario se le recomendará objetos que no haya evaluado, y que hayan sido evaluados de manera positiva por los miembros del grupo en el que esté incluido. Además, se filtra el contenido que haya sido evaluado de forma negativa.

Al igual que los sistemas basados en contenido, los sistemas colaborativos tienen una serie de problemas que se han ido descubriendo conforme se han hecho más populares y más utilizados, como son los problemas de: la escasez, la escalabilidad y del ítem nuevo.

- **Escasez:** los sistemas de recomendaciones colaborativos necesitan de una gran cantidad de datos, muchos usuarios puntuando muchos ítems similares para así poder calcular los grupos de vecinos y, en base a ellos, realizar las recomendaciones. Si en la base de datos se tiene pocos usuarios o pocas puntuaciones por parte de cada usuario, la matriz de puntuaciones será muy escasa y los cálculos de vecindad, predicción y recomendación no pueden ser realizados con la suficiente seguridad y exactitud obteniendo por lo tanto unas recomendaciones de baja calidad.
- **Escalabilidad:** los sistemas de recomendaciones colaborativos usan por forma general algoritmos de cálculo de los k vecinos más cercanos (Knn, K-nearest neighbors) para obtener la similaridad entre usuarios. Estos algoritmos son costosos computacionalmente y su costo crece linealmente cuanto mayor sea el número de usuarios y de ítems, por lo que son base de datos con millones de elementos, al aumentar el número de datos, el sistema sufrirá graves problemas de escalabilidad.
- **Problema del ítem nuevo:** en los sistemas de recomendaciones colaborativos los ítems nuevos, que tienen muy pocas o incluso ninguna puntuación, no van a ser recomendadas prácticamente nunca. De la misma forma, los nuevos usuarios en el sistema recibirán muy pobres predicciones, debido a que ellos han puntuado muy pocos ítems y se hace difícil encuadrarlos en algún grupo de vecinos. Estos dos hechos nos hacen ver que estos sistemas de recomendación requieren un cierto tiempo antes de empezar a hacer predicciones y recomendaciones ciertamente relevantes y acertadas.

Una gran cantidad de experimentos, estudios e investigaciones se han realizado en los últimos años, para encontrar técnicas que reduzcan el impacto de estos problemas en los sistemas de recomendación con filtrado colaborativo.

Para reducir el problema de la escasez, se han intentado la utilización de puntuaciones implícitas, la correlación entre ítems y el filtrado híbrido. Mientras que para tratar de mejorar la escalabilidad se han propuesto la reducción de la dimensionalidad y aproximaciones basadas en modelos. Finalmente, estudios han demostrado que las técnicas de web mining (la aplicación de técnicas de minería de datos para descubrir los patrones de la Web) como los árboles de decisión, son útiles para paliar el problema de los ítems y usuarios nuevos.

## Categorías de los algoritmos de Filtrado Colaborativo

Para obtener un sistema de recomendación colaborativo de calidad es necesario elegir un buen algoritmo de filtrado colaborativo. Estos algoritmos se pueden clasificar dentro de dos categorías:

- **Algoritmos basados en memoria o basados en usuario:** Estos algoritmos utilizan la base de datos completa para generar una predicción. El funcionamiento de estos algoritmos es el siguiente: se utilizan técnicas estadísticas para encontrar un conjunto de vecinos al usuario activo y posteriormente se utilizan una serie de algoritmos que combinan las preferencias de esta vecindad para realizar las predicciones y recomendaciones.
- **Algoritmos basados en modelos o basados en ítem:** Estos algoritmos proporcionan recomendaciones de ítems desarrollando primero un modelo (ya sea mediante bayesianas, clustering, o modelos basados en reglas) de las puntuaciones de los usuarios sobre los ítems.

No se utilizan técnicas estadísticas sino una aproximación probabilística que calcula el valor esperado de una predicción del usuario dados sus puntuaciones sobre otros ítems. Es decir, estos algoritmos miran en el conjunto de ítems que el usuario activo ha puntuado o evaluado y calcula como de similar son estas puntuaciones con respecto al ítem activo con el fin de realizar una predicción para el mismo.

Los sistemas de recomendación colaborativos basados en usuario son los que centrarán el proyecto.

## Ejemplos reales de Sistemas de Recomendación Colaborativos

- **Tapestry** Tapestry, un proyecto de Xerox PARC, está considerado como el primer sistema de recomendación que implementaba filtrado colaborativo. Permitía a sus usuarios encontrar documentos basados en comentarios hechos previamente por otros usuarios.
- **Zagat.com:** Zagat Survey es una empresa americana que se dedica a la edición de todo tipo de guías de restaurantes, hoteles, clubes o tiendas de distintas ciudades de los Estados Unidos y Canadá. Los usuarios registrados pueden votar distintos aspectos del local referido y, además, introducir pequeños comentarios con su experiencia. En base a estas votaciones los responsables de la empresa asignan sus puntuaciones en sus guías anuales y hacen recomendaciones individuales a sus usuarios a través de su web.
- **Filmaffinity.com:** Filmaffinity es un proyecto español de gran proyección que se encarga de recibir puntuaciones de todo tipo de películas y dar recomendaciones a sus usuarios. El nuevo usuario puede empezar a votar las películas que haya visto con la ayuda de unos

tours dirigidos que acomodan de manera inteligente los problemas de escasez y nuevo ítem; posteriormente el sistema calcula las almas gemelas de este nuevo usuario y le recomienda las películas favoritas de estas almas gemelas que no haya votado el usuario.

- **MovieLens:** MovieLens es un sistema de recomendación de películas online basado en filtrado colaborativo. Desarrollado por el GroupLens Research de la Universidad de Minnesota, recolecta puntuaciones sobre películas de sus usuarios y en base a esos datos agrupa los usuarios de similares gustos. Atendiendo a las puntuaciones de todos los usuarios dentro de un grupo se intenta predecir para cada usuario individual su opinión sobre películas que todavía no ha visto.

Los datos sobre sus usuarios y sus puntuaciones son privados pero los investigadores de GroupLens mantienen como públicas dos ejemplos de 100000 y un millón de puntuaciones respectivamente. Estos ejemplos se pueden descargar desde la propia página (<http://www.grouplens.org>)

## Sobre el Sistema

Se utilizará, en el sistema de recomendación a implementar, un filtrado colaborativo basado en usuarios. Se utilizarán datos de la página nombrada anteriormente de grouplens, donde se tienen puntuaciones sobre películas de usuarios. En base a estas agruparán los usuarios con gustos similares. Luego se aplicarán técnicas basadas en perfil de usuario en base a preferencias positivas y negativas. Se procede a:

1. Identificar grupos con los usuarios similares de la base de datos con distintas **medidas de similitud**. Algunas medidas de similitud existentes son:
  - Coeficiente coseno
  - Distancia euclídea
  - **Coeficiente de correlación de pearson.**

En este caso utilizaremos como medida de similitud, el coeficiente de correlación de Pearson.

2. Elegir un algoritmo de filtrado colaborativo. En este caso, se utilizaron algoritmos de predicción basados en usuarios, como el de **weighted sum**.
3. Aplicar técnicas de perfil de usuario, en base a preferencias positivas y negativas

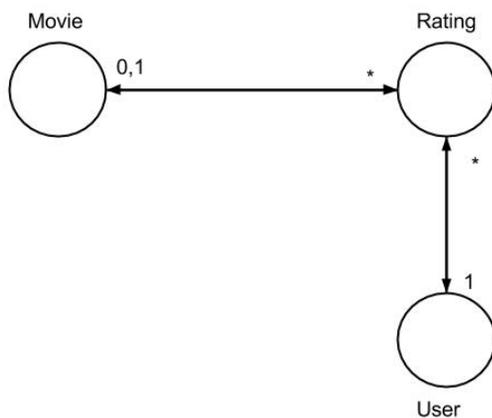
## Conjunto de datos

Se utilizó un conjunto de datos de la página MovieLens, creado por el grupo de GroupLens. Se utilizará como punto de partida, los datos publicados con 100000 puntuaciones.

En esta página, los usuarios votan por diferentes películas con una valoración comprendida entre el 1 y 5. De los usuarios se conoce su sexo, edad, ocupación, código postal, y de las películas su título, su fecha de estreno y los géneros cinematográficos a los que pertenecen. Las películas aparecen junto a su título, su fecha de lanzamiento, su dirección en el sitio web IMDb y los géneros a los que pertenecen.

## Datos utilizados

La base de datos creada, responde al siguiente modelo de datos:



## Descripción de las tablas

- **Géneros:** Descripción del género.
- **Movies:** Título de las películas, fecha de lanzamiento, dirección en el sitio web IMDb y los géneros a los que pertenecen.
- **Ratings:** Puntuación, usuario que realizó la votación y fecha y hora en que la realizó.
- **Users:** Edad, género, ocupación y código postal de los usuarios.

## Requerimientos funcionales de la aplicación

Se realizó una aplicación web, en donde se ingresa el identificador de un usuario cargado en la base y se despliegan las películas recomendadas para ese usuario, en base a las valoraciones realizadas por el mismo, presentes en la base de datos.

- Se utilizan el historial de búsqueda de los usuarios.
- El perfil de usuario es creado en base a las votaciones de películas previas hechas por el mismo.
- Se calculan las preferencias positivas y negativas. Las preferencias positivas son los conceptos y los temas en que el usuario está interesado y las negativas son los conceptos en que el usuario no está interesado en absoluto.
- Luego, se despliegan las recomendaciones, ordenadas por puntaje sugerido de mayor a menor, es decir, se muestran primero las recomendaciones en las cuales posiblemente esté interesado.
- Se da la opción de ingresar la cantidad de películas que se desean que se muestren, y de elegir entre los dos algoritmos presentados, para el cálculo de las recomendaciones.



### Recomendaciones de Películas

Ingrese un código de usuario:

Código de usuario

Cantidad de películas

**Buscar Recomendaciones Weighted Sum**      **Buscar Recomendaciones Weighted Sum Negative**      **Buscar Recomendaciones Weighted Sum Adjustment**

---

Ingrese un usuario.

## Diseño del sistema

### Arquitectura

Se utilizó una arquitectura Cliente/Servidor.

**Base de datos:** Información sobre usuarios, películas y puntuaciones.

**Lógica del negocio:** Aplicación de algoritmos para obtener los usuarios vecinos más similares y así poder recomendar películas.

**Interfaz web:** Ingreso y despliegue de los datos.

### Herramientas utilizadas

Se utilizó para crear el entorno de desarrollo el IDE Eclipse y el lenguaje de programación JavaEE- Para la implementación de la interfaz web, se utilizó JSF que es una tecnología y framework para aplicaciones Java basadas en web que simplifica el desarrollo de las interfaces de usuario. JSF usa JavaServer Pages (JSP) como la tecnología que permite hacer el despliegue de las páginas.

Se utilizó MySQL es un sistema de gestión de la base de datos.

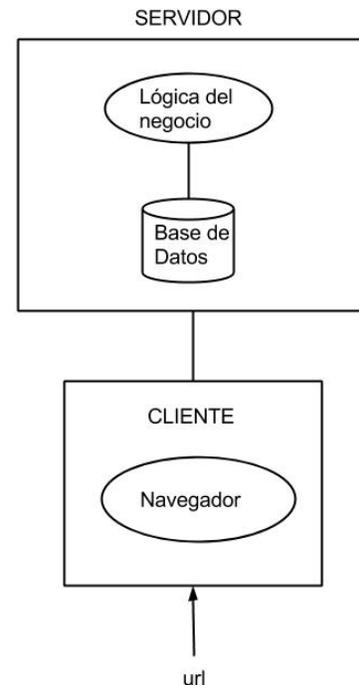
Como servidor de la aplicación se utilizó JBoss. Al estar basado en Java, JBoss puede ser utilizado en cualquier sistema operativo para el que esté disponible la máquina virtual de Java.

### Algoritmos utilizados

#### Algoritmo basado en usuarios

Cuando el usuario activo solicita una recomendación, este algoritmo trata de vincularlo con otros usuarios del sistema. Dos usuarios están vinculados si ambos puntuaron elementos en común. El conjunto de usuarios con los que se vincula al usuario activo se le denomina *vecindario*.

Serán vecinos del usuario activo, aquellos que tienen puntuados en un conjunto de ítems en común con él, sin importar las preferencias sobre éstos.



## Similitud

El concepto de similitud se expresa mediante un valor, que se calcula considerando los elementos puntuados en común. Si ambos puntúan de manera similar estos elementos, se obtiene un valor alto de similitud, mientras que si expresan valoraciones opuestas se obtiene un valor de similitud bajo.

El algoritmo busca calcular las predicciones sobre aquellos elementos que los vecinos conocen, pero que son desconocidos para el usuario activo. A este conjunto de elementos le llamaremos *conjunto de ítems recomendables*, dado que son potencialmente los elementos que pueden llegar a ser recomendados.

Para realizar la recomendación se calcula la predicción del puntaje para cada elemento del conjunto de ítems recomendables. Una vez calculados todos estos puntajes se toma una cantidad arbitraria de elementos, los cuales tienen asociados los mayores valores de puntaje de predicción.

## Correlación de Pearson

Para determinar la similitud entre dos usuarios,  $w(a,i)$  uno de los mecanismos más utilizados es el coeficiente de correlación de Pearson, donde la correlación entre los usuarios  $a$  e  $i$  se calcula como muestra en la fórmula

$$w(a,i) = \frac{\sum_j (v_{a,j} - \bar{v}_a)(v_{i,j} - \bar{v}_i)}{\sqrt{\sum_j (v_{a,j} - \bar{v}_a)^2 \sum_j (v_{i,j} - \bar{v}_i)^2}}$$

En esta fórmula, el índice  $j$  corresponde a los ítems que fueron evaluados por los usuarios  $a$  e  $i$ .

En el numerador podemos ver que, para cada elemento  $j$  de la sumatoria, se multiplican dos términos.

El primer término establece si el usuario activo ( $a$ ) evalúa el elemento  $j$  de manera positiva o negativa, tomando su promedio de votos como valor neutro.

De la misma manera en el segundo término se evalúa el voto del usuario  $i$  con respecto al ítem  $j$ . Al multiplicar estos dos términos se obtendrá un valor negativo, sólo si uno de los usuarios lo puntuó por debajo del promedio de votación y el otro usuario por encima. Este valor negativo provoca una disminución del valor total de similitud  $w$ , lo cual tiene sentido si consideramos el hecho de que los usuarios valoraron de manera opuesta el elemento  $j$ . Análogamente obtendremos un valor positivo en la multiplicación, cuando ambos usuarios valoraron de igual manera el elemento  $j$  en función de sus valores promedio de votación. El denominador simplemente normaliza el valor de similitud.

## Cálculo de predicción de puntajes: Average + adjustment

Se utilizaron dos técnicas para el cálculo de la predicción de puntajes, una de ellas es la presentada en la siguiente fórmula:

Se define un voto  $v_{i,j}$  como el puntaje brindado por el usuario  $i$  para el ítem  $j$ . Se considera además  $\bar{v}_i$  como el puntaje promedio de los votos del usuario  $j$ , y  $w(a,i)$  con el valor de similitud entre el usuario  $a$  y el usuario  $i$ .

$$p_{a,j} = \bar{v}_a + k \sum_{i=1}^n w(a,i)(v_{i,j} - \bar{v}_j)$$

En esta fórmula,  $n$  representa la cantidad de vecinos en el sistema y  $k$  es un factor de normalización de la ecuación, para que el valor que se suma al promedio de votos del usuario activo, sea coherente con la escala de puntajes del sistema.

Analizando la fórmula, se puede observar que el primer término es el valor promedio de votación del usuario activo y el segundo representa la variación del puntaje a predecir con respecto a ese promedio. Esta variación se calcula considerando todos los usuarios que tienen elementos en común con el usuario activo y que además han puntuado el elemento  $j$ .

En este caso se utilizó el valor de  $k$  como:

$$\frac{1}{\sum_{i=1}^n w(a,i)}$$

Utilizando este  $k$ , puede ocurrir que el resultado del puntaje de predicción sea un valor por encima o por debajo del máximo y mínimo puntaje permitido. En este caso, se asignan el máximo y el mínimo puntaje de la escala respectivamente.

## Votos positivos y negativos

En primer lugar es necesario determinar si los votos de los vecinos son positivos o negativos con respecto a el ítem  $j$ . Una posibilidad es comparar ese voto con el valor medio de la escala de votación. En tal caso no se consideran las distintas personalidades de los usuarios. Si en vez de utilizar el valor medio se utiliza el promedio de las votaciones registradas del usuario como

valor neutro, se asume que los votos superiores al promedio son positivos y los menores negativos. De esta forma se tiene en cuenta la tendencia o personalidad del usuario al votar, dado que la escala de puntajes de los usuarios es variada y un voto aceptable para un usuario puede estar contenido en un rango de valores diferente al de otro usuario.

Por ejemplo, considerando una escala de puntajes de 1 a 5, el promedio de votaciones del usuario  $u_1$  igual a 2 y el promedio del usuario  $u_2$  igual a 4. Se quiere saber el agrado o desagrado de una canción que tanto el usuario  $u_1$  como el  $u_2$  le asocia un puntaje igual a 3. Si comparamos contra el promedio de votaciones de cada usuario se concluye que al usuario  $u_1$  le agrada más la canción que al usuario  $u_2$ , dado que para  $u_1$  el valor de 3 supera su promedio de votación (2) mientras que para  $u_2$  no alcanza su promedio (4). Sin embargo en caso de considerar el promedio de la escala de votaciones (3) en vez del promedio de cada usuario, se concluiría que a los dos usuarios le agrada de la misma forma, lo cual no sería real.

Es preciso observar que, si un usuario del sistema puntúa a todos sus ítems con el mismo valor, no será posible realizarle una recomendación: no se puede aplicar la fórmula de coeficiente de correlación de Pearson pues se realiza una división entre cero. Esto tiene sentido dado que si el usuario asigna el mismo puntaje a todos los ítems, no se tiene información sobre sus preferencias y por lo tanto no se le puede realizar ninguna recomendación.

## Weighed Sum

La siguiente técnica utilizada para la predicción de puntajes es el algoritmo de **Weighed Sum**.

$$p_{a,j} = \frac{\sum_{i=1}^n w(a,i) * v_{i,j}}{\sum_{i=1}^n |w(a,i)|}$$

Este método calcula la predicción del puntaje que un usuario  $a$  le daría a un ítem  $j$ . Para ello realiza la suma de todas las evaluaciones de los usuarios sobre ese ítem. Cada una de las valoraciones está ponderada con la correspondiente similaridad  $w(a,i)$  entre el usuario activo y cada uno de los usuarios. Para este caso, también se utilizó la correlación de Pearson para el cálculo de la similaridad.

## Resultados

Se realizaron comparaciones entre ambos algoritmos:

- **Weighted Sum sin preferencias negativas - Average + adjustment**

Para puntajes altos, las recomendaciones brindadas por ambos algoritmos son similares, pero para el algoritmo de weighted sum, se presentan más recomendaciones con el mismo puntaje. Por ejemplo, para el usuario con id 165, se muestran las primeras 15 recomendaciones con puntaje 5.0, mientras que utilizando la técnica de - Average + adjustment, se muestran prácticamente las mismas películas, con puntajes entre 5.94 y 5.2 aprox. Es decir, que éste último método es más efectivo para “rankear” las películas.

### Weighted Sum sin preferencias negativas

Película	Valor
Little City (1998)	5.0000000000000001
Someone Else's America (1995)	5.0
Entertaining Angels: The Dorothy Day Story (1996)	5.0
Sliding Doors (1998)	5.0
Saint of Fort Washington, The (1993)	5.0
Boys, Les (1997)	5.0
Santa with Muscles (1996)	5.0
Flirt (1995)	5.0
Great Day in Harlem, A (1994)	5.0
They Made Me a Criminal (1939)	5.0
Killer (Bulletproof Heart) (1994)	5.0
Marlene Dietrich: Shadow and Light (1996)	5.0
Prefontaine (1997)	5.0
Hugo Pool (1997)	5.0
Star Kid (1997)	5.0

## Average + adjustment

Película	Valor
Flirt (1995)	5.945337620578778
Great Day in Harlem, A (1994)	5.90251572327044
Boys, Les (1997)	5.877083333333333
Saint of Fort Washington, The (1993)	5.6375
Delta of Venus (1994)	5.6231884057971016
Nico Icon (1995)	5.514372248573186
Some Mother's Son (1996)	5.481903470791159
Hugo Pool (1997)	5.478260869565217
Someone Else's America (1995)	5.418250950570342
Sliding Doors (1998)	5.4
Pather Panchali (1955)	5.376581422194311
Star Kid (1997)	5.3431229425597735
Entertaining Angels: The Dorothy Day Story (1996)	5.294117647058823
Boy's Life 2 (1997)	5.2296802962531
Lamerica (1994)	5.22691674155516

Lo mismo sucede para los puntajes más bajos: para el mismo usuario anterior, utilizando la técnica *weighted sum*, se muestran las últimas 43 películas con puntaje 1, mientras que con la otra técnica, las últimas 43 se encuentran rankeadas entre 2.4 y 0.8, de las cuales 25 películas tienen en común con el algoritmo anterior.

Posiblemente sea por la utilización de promedios, ya que en el *weighted sum* no se considera la personalidad del usuario, en cuanto a sus votaciones, y si la calificación es de valor bajo, la considera baja para todos los usuarios. Sin embargo, en la técnica de *average+adjustment* sí se considera, y posiblemente sea esa la razón por la cual no se recomiendan con puntaje bajo varias películas que con la otra técnica sí.

- **Weighted Sum con preferencias negativas**

Con el motivo de tomar en cuenta las preferencias negativas a la hora de hacer la recomendación, para el algoritmo de *Weighted Sum*, se tomó en cuenta una escala de puntuación del 1 al 5, y se consideró como preferencias positivas aquellas superiores al 2. De esta manera, se consideraron opiniones negativas aquellas con puntuación 1 o 2. Para la ejecución del algoritmo, se tomó como -1 el puntaje 2, y -2 el puntaje 1. Se observó que esto era equivalente a cambiar la escala de votación del 1 al 5 a -2 y 3, pudiendo filtrarse más fácilmente las películas con puntajes menores, pero en la práctica el orden de puntuación de las películas a recomendar, era el mismo. Por lo que se consideró que esta técnica no realizaba un gran aporte al algoritmo original.

## **Trabajo a futuro**

En el sistema presentado, ya se tienen los datos cargados de la página movielens, pero se deja planteado, como trabajo a futuro, la incorporación de:

- Opción para crear una cuenta de usuario.
- Presentar las películas categorizadas, para una mejor búsqueda para el usuario.
- Mostrar más información acerca de las películas.
- Poder ingresar comentarios y permitir que el usuario pueda calificarlas, para ir creando el perfil de usuario.
- Sacar estadísticas, sobre los perfiles de los usuarios, relacionando los datos de la cuenta y las películas que prefieren o no prefieren (preferencias negativas).
- Profundizar en la implementación y/o actualización de algoritmos con el motivo de dar una recomendación cada vez más cercana al deseo del usuario.
- Estudiar aspectos relacionados con la credibilidad de las recomendaciones de los algoritmos, para considerar o no la validez de dichas recomendaciones.

## Conclusión

El trabajo realizado en este proyecto se enfocó en el estudio de los sistemas de recomendación, analizando particularmente la técnica de filtrado colaborativo. Además, se intentó enfocar en las preferencias negativas brindadas por los usuarios. En ese sentido, se analizaron los algoritmos más usados, encontrándonos con poca documentación acerca de las preferencias negativas.

Hemos visto que los sistemas colaborativos son capaces de realizar recomendaciones de ítems a usuarios, basándose en la idea de que a un usuario le gustará un ítem si a otros usuarios con gustos parecidos les gustó. Este tipo de sistemas permite realizar predicciones sobre elementos de difícil tratamiento y con características no inmediatas, sin necesidad de analizarlos, simplemente utilizando las valoraciones previas que otros usuarios realizaron acerca de ellos.

Para el cálculo de similitud entre usuarios, se utilizó el coeficiente de correlación de Pearson. Si bien este algoritmo produce buenos resultados, cuando es aplicado sobre grandes cantidades de información, tiene en la performance su principal limitación.

Se evaluaron y compararon los diferentes métodos de filtrado colaborativo basados en usuario. Los resultados obtenidos muestran que el uso de las preferencias positivas y negativas de los usuarios mejora la calidad de los aciertos obtenidos, en comparación a aquellos que solo utilizan las preferencias positivas. Además las preferencias negativas en los perfiles de usuario también ayudan a separar consultas similares y disímiles en grupos distantes.

## Referencias

[1] Sistema de Recomendación de Alquiler de Películas

[http://sinbad2.ujaen.es/cod/archivosPublicos/pfc/pfc\\_fernando\\_siles.pdf](http://sinbad2.ujaen.es/cod/archivosPublicos/pfc/pfc_fernando_siles.pdf)

[2] Desarrollo de una Herramienta Business Intelligence para el Análisis de Redes Sociales Almacenadas en Grafos.

<http://repositorio.unican.es/xmlui/bitstream/handle/10902/3108/Noelia%20Ruiz%20Martinez.pdf?sequence=1>

[3] *A survey on retrieving Contextual User Profiles from Search Engine, 2012*

1G.Ravi, Malla Reddy College of Engineering and Technology, Hyderabad.

[4] *Web Usage Mining: Personalization based on User Positive and Negative Preferences*

[5] *Object oriented implementation of Concept-Based User Profiles from Search Engine Logs,*

Nagaveni Budati, G.Appa Rao, Uppe Nanaji

[6] Sistema de Recomendación de Música

<http://www.fing.edu.uy/inco/grupos/pln/prygrado/InformePGMusica.pdf>