

# Implementación de dos Sistemas de Recomendación sencillos

## (revisado en Mayo del 2018)

Hernández Arias Santiago (1631281), Vallecilla Cuella Cristian Camilo (1628790), Campiño Taborda Christian Camilo (1632081)

**Abstract**—this paper looks to show how a recommender system can be implemented by means of simple mathematical processes which involve numerical methods. Two recommender systems were implemented, one through matrix factorization and the other using the Pearson correlation coefficient. The recommender systems try to predict the ratings a user would give to elements based on his/her existing records and behavior. We hope to show how the obtained results are different to each other and respond to the differences in the implementations and the theory behind them.

**Index Terms**—Recommender System, matrix factorization, latent features, Pearson correlation coefficient



## 1 INTRODUCCIÓN

EL acelerado crecimiento de la cantidad de información disponible en Internet y el cada vez mayor número de usuarios han convertido la sobrecarga de datos en un serio problema que puede perjudicar significativamente los tiempos de acceso a los recursos y elementos de interés disponibles en la red. Junto a ello se encuentran los nuevos retos relativos a la priorización y personalización de la información. En este contexto, los sistemas de recomendación han cobrado mucha importancia recientemente.

Los sistemas de recomendación son sistemas cuyo propósito es el filtro de la información. Se encargan de aislar porciones precisas y reducidas de datos a partir del conjunto total de información. Esto lo hacen basándose en múltiples criterios relacionados tanto con los usuarios como con los elementos. Los sistemas de recomendación examinan sus características y su comportamiento con el fin de predecir si un usuario favorecerá un determinado elemento de acuerdo con la información observada y tratada.

La importancia de los sistemas de recomendación surge del hecho de que pueden incrementar los beneficios de proveedores y usuarios de la información. Son herramientas efectivas en distintos ámbitos, desde el comercio virtual hasta la investigación científica, puesto que a través de ellos se pueden aumentar ventas, reducir costos de aislar y seleccionar objetos para los usuarios, minimizar el tiempo empleado en la observación de catálogos, entre muchas otras ventajas. Con todo esto en cuenta, resulta menester la búsqueda de nuevas técnicas que soporten los sistemas de recomendación y el análisis de las que ya existen, con miras a ofrecer recomendaciones acertadas y relevantes a la creciente base de usuarios de los sistemas

informáticos [1]. En este trabajo se busca exponer la teoría detrás de dos sistemas de recomendación sencillos, mostrar resultados de su utilización y describir los distintos resultados que producen.

## 2 PROCEDIMIENTO

La información utilizada en este trabajo corresponde a la base de datos de Book-Crossing, la cual contiene información acerca de 278.858 usuarios, 271.379 libros y 1.149.780 puntuaciones [2].

La primera consideración ha sido la reducir el número de datos a tratar. Con ello en mente, de la tabla de puntuaciones se extrajo la información perteneciente a aquellos usuarios que aportaron cincuenta o más calificaciones. Además, se eliminó la información de libros con menos de setenta calificaciones. Esta medida se tomó en aras de trabajar solo con la información más diciente disponible y a la vez reducir la potencia computacional requerida para procesar los datos. Después de estos filtros, la tabla de puntuaciones quedó con calificaciones de 1101 usuarios y 264 libros. Es importante notar que las 1.149.780 calificaciones originales fueron reducidas en número a 8788. Esto evidencia que las puntuaciones iniciales, en gran mayoría, correspondían a usuarios con muy pocas calificaciones realizadas o a libros revisados en muy pocas ocasiones.

Los dos sistemas de recomendación implementados y probados utilizan cada uno una técnica distinta.

### 2.1 Sistema de recomendación basado en factorización de la matriz de puntuaciones

El modelamiento de la información se ha hecho como se ilustra en la **Figura 1**.

	D1	D2	D3	D4
U1	5	3	-	1
U2	4	-	-	1
U3	1	1	-	5
U4	1	-	-	4
U5	-	1	5	4

Figura 1. Matriz de puntuaciones.

Las filas de la matriz de puntuaciones representan a los usuarios y las columnas a los elementos calificados, en este caso libros. El número en cada entrada es la puntuación o rating que el usuario le ha otorgado al libro respectivo. Los espacios marcados con “-” significan que el usuario no ha calificado el libro. La **Figura 1** es solo una ejemplificación, puesto que las puntuaciones de la base de datos varían entre enteros del 1 al 10, y en la práctica los espacios “-” fueron sustituidos por ceros. Así pues, el objetivo primario del procedimiento consiste en predecir las calificaciones en las entradas sin número, es decir, pronosticar el valor que el usuario daría en caso de puntuar el libro que aún no valoró.

Las predicciones de este procedimiento se hacen a través de la factorización de la matriz de puntajes original. Se busca entonces hallar dos matrices cuyo producto sea la matriz original. La idea tras la factorización como método para este primer sistema de recomendación es el descubrimiento de los factores latentes. Los factores latentes representan las interacciones entre entidades distintas. Son puntos en común que pueden hacer similares a varios usuarios o a varios elementos. También pueden ser descritos como características derivadas de las cualidades explícitas de las entidades. En el caso de los libros, un factor latente puede ser el género literario, otro puede ser el autor y uno más el contexto social de cuando fue publicado. Así, los factores latentes son características puntuales que pueden explicar o impersonar un parecido entre dos elementos. Debe anotarse que el número de factores no puede superar nunca el número de usuarios o elementos, puesto que esto significaría que no existen puntos comunes entre ellos y por tanto no tendría sentido realizar recomendaciones, pues todos los usuarios y elementos serían únicos y sin parecidos entre ellos [3].

En términos generales, este primer método de recomendación puede ejemplificarse como sigue: en la **Figura 1**, los usuarios U1 y U2 calificaron al libro D1 con 4 y 5. El usuario U1, además, calificó al libro D2 con 3. El sistema tenderá a predecir que U2, que no calificó aún a D2, lo haría con un puntaje cercano a 3, derivado de su similitud con el usuario U1 al momento de puntuar al libro D1.

Para la factorización de la matriz, se usan dos matrices iniciales P y Q. k es el número de factores latentes esti-

mados. P tiene un número de filas igual al número de usuarios y un número de columnas igual al número de factores latentes (1101x2). Q tiene un número de filas igual al número de libros y un número de columnas igual al número de factores latentes (264x2). La matriz P contendrá una serie de números aleatorios entre 0 y 1, los cuales indicarán la fuerza que tiene la relación entre los usuarios y el factor latente correspondiente, lo mismo será para la matriz Q, pero su relación será entre los libros y los factores.

Una vez inicializadas ambas matrices se procederá a realizar la primer matriz de predicciones, esto se hará mediante el producto punto de la fila i-ésima de P con la fila i-ésima de Q. Lo anterior se expresa mejor mediante la siguiente ecuación:

$$\hat{r}_{ij} = p_i^T q_j = \sum_{k=1}^k p_{ik} q_{kj}$$

A continuación se efectúa un análisis del error. Como se ha de saber, la matriz de predicciones no será igual a la matriz original, pues entre ambas existirá un error respecto al valor real con el valor predicho:

$$e_{ij}^2 = (r_{ij} - \hat{r}_{ij})^2 = (r_{ij} - \sum_{k=1}^K p_{ik} q_{kj})^2$$

La diferencia entre ambos valores se eleva al cuadrado porque no se tiene certeza de si la calificación estimada será mayor o menor que la original. Lo principal ahora es reducir la magnitud del error, para ello se hace uso del método del gradiente descendiente. Cabe resaltar que el error a minimizar es respecto de los ratings que ya existían, no de los ratings a predecir pues estos inician en 0. El propósito es calcular el gradiente respecto a la matriz P y respecto a la matriz Q, luego se busca minimizar iterativamente el error avanzando en la dirección contraria a la que el gradiente crece. Los gradientes se calculan de la siguiente forma:

$$\begin{aligned} \frac{\partial}{\partial p_{ik}} e_{ij}^2 &= -2(r_{ij} - \hat{r}_{ij})(q_{kj}) = -2e_{ij} q_{kj} \\ \frac{\partial}{\partial q_{ik}} e_{ij}^2 &= -2(r_{ij} - \hat{r}_{ij})(p_{ik}) = -2e_{ij} p_{ik} \end{aligned}$$

Para determinar la velocidad con la que el gradiente decrecerá utilizamos un coeficiente denominado como paso ( $\alpha$ ), el valor del paso utilizado es de 0.00002. El paso debe tener un valor pequeño para que las iteraciones nos permitan llegar al mínimo y no oscilar alrededor del mismo. Una vez el error llegue a un valor aceptado, salimos del bucle y la última matriz de predicciones será la mejor, pues donde ya existían los ratings el error será mínimo y donde no existían ya se dispondrá de unos ratings predichos y consistentes con la recomendación esperada.

Al final del procedimiento se hace una regularización al cálculo del error. Para ello se utiliza un parámetro beta

( $\beta$ ), el cual permitirá que las magnitudes de los vectores factor-usuario y factor-libro provean un valor más aproximado al rating real sin la necesidad de contener números muy grandes. El cálculo del error con el parámetro beta incluido se realiza de la siguiente forma:

$$e_{ij}^2 = (r_{ij} - \sum_{k=1}^K p_{ik}q_{kj})^2 + \frac{\beta}{2} \sum_{k=1}^K (\|P\|^2 + \|Q\|^2)$$

Finalmente, una vez se indique el usuario al cual se le realizarán las recomendaciones, se accede a la matriz de puntajes que resulta del producto entre P y Q después del proceso de factorización iterativo. Esta matriz se caracteriza porque los valores originales y distintos de cero (las puntuaciones iniciales) cambian muy poco, mientras que los ceros son sustituidos por las puntuaciones pronosticadas. Esto gracias a la minimización del error que tiene en cuenta solo los valores iniciales distintos de cero. Para ilustrar esto, se emplea la **Figura 2**:

	D1	D2	D3	D4
U1	4.97	2.98	2.18	0.98
U2	3.97	2.40	1.97	0.99
U3	1.02	0.93	5.32	4.93
U4	1.00	0.85	4.59	3.93
U5	1.36	1.07	4.89	4.12

Figura 2. Matriz de puntuaciones con los puntajes ya predichos, después de factorizar y producto de P y Q

Como puede observarse, las puntuaciones originales varían muy levemente. Como se mencionó que ocurriría, el sistema predijo un puntaje de 2.4 para el usuario U2 y el libro D2, basándose en su similitud con el usuario U1 y su calificación al libro D2. Las pruebas de la implementación y los resultados se expondrán en la **sección 3**.

## 2.2 Sistema de recomendación basado en filtro colaborativo basado en items mediante coeficiente de correlación de Karl Pearson

Con la matriz de puntuaciones, se calcula la similitud que existe entre los items y basado en ellas aunadas a la puntuación que un usuario específico ha dado a libros parecidos, se realizan recomendación prediciendo nuevamente puntajes de libros no calificados.

En este procedimiento las similitudes son halladas con la correlación de todas las columnas (valoraciones de los usuarios a cada libro) entre sí, mediante el cálculo del coeficiente de la correlación de Karl Pearson. Éste muestra el grado de correlación o similitud que existe entre cada una de las columnas. El valor del coeficiente se calcula con la siguiente expresión:

$$r = \frac{\sum (x - \bar{x})(y - \bar{y})}{\sqrt{\sum (x - \bar{x})^2} \sqrt{\sum (y - \bar{y})^2}}$$

X e Y son las columnas a las cuales se les encuentra el coeficiente de correlación. Los coeficientes calculados son almacenados en una matriz como se muestra en la **Figura 3**:

	Libro 1	Libro 2	Libro 3	Libro 4	Libro 5
libro 1	1	0,6	0,45	0,65	0,89
libro 2	0,6	1	0,7	-0,45	0,7
libro 3	0,45	0,7	1	0,8	-0,09
libro 4	0,65	-0,45	0,8	1	0,1
libro 5	0,89	0,7	-0,09	0,1	1

Figura 3. Ejemplo de matriz de coeficientes de correlación

En la anterior figura se observa un ejemplo de cómo cada celda representa el grado en que cada libro se parece a los demás. Si el coeficiente r es 1, la similitud es perfecta y los elementos son idénticos, de ahí que la diagonal sean unos. Valores entre 0 y 1 indican similitudes normales. Si r es 0, no existe similitud. Si r está entre -1 y 0 hay similitud negativa, es decir que los elementos son diferentes y si r es -1 los elementos son totalmente diferentes [4].

Después de calcular estas similitudes, se define el usuario al que se le desea ofrecer recomendación. Sus calificaciones son extraídas y multiplicadas cada una con su fila correspondiente en la matriz de similitudes. Así, la calificación que el usuario da al libro 1 se multiplica por los valores de la fila 1. Las similitudes negativas o nan (resultado que no pudo calcularse, generalmente ocurre cuando ambas columnas contienen en mayor cantidad la calificación de cero) no son tenidas en cuenta. Para cada libro se guarda la media de valores obtenidos por medio de estas multiplicaciones y se recomiendan aquellos cuya media sea superior.

Una consideración valiosa es que este Segundo procedimiento es menos costoso computacionalmente puesto que la multiplicación por la matriz de similitud implica solo las puntuaciones del usuario al que se le desea recomendar, en contraste con el primer procedimiento que conlleva más cálculos y que al final realiza predicciones para todos los usuarios. Esto se refleja más adelante en los tiempos de ejecución.

## 3 RESULTADOS

En esta sección se busca mostrar los resultados obtenidos al aplicar los métodos de recomendación implementados a la base de datos conseguida tras los filtros realizados que fueron descritos al principio de la **Sección 2**. Debido a que la matriz de puntajes original es una matriz muy

dispersa de dimensión 1101x264, resulta dispendioso el buscar ejemplos puntuales de puntajes similares entre libros y usuarios, por lo que se mostrarán ejemplos de resultados con matrices similares a la de la **Figura 1** para evidenciar con sencillez el trabajo del método. Respecto a la información real, se mostrarán las recomendaciones como tal conseguidas y una discusión lacónica de los mismos.

### 3.1 Resultados del primer método – factorización de la matrices de puntajes

Si se utiliza la matriz de la **Figura 1**:

```
Matriz R:
5 3 0 1
4 0 0 1
1 1 0 5
1 0 0 4
0 1 5 4
```

*Figura 4. Matriz de puntuaciones R original*

A través del método descrito, se consiguen las matrices P y Q cuyo producto aproxima a R:

```
Matriz P:
0.572394 0.959767
0.854361 0.814215
0.936799 0.581373
0.174163 0.195568
0.23312 0.591268

Matriz Q:
0.528086 0.497346
0.374616 0.63991
0.196457 0.706133
0.210127 0.632814
```

*Figura 5. Matrices P y Q que factorizan a R, obtenidas a partir del primer procedimiento*

El producto de P y Q es la nueva matriz R de puntajes con los valores predichos en donde originalmente habían ceros:

```
Matriz Nueva R:
4.99039 2.94881 3.73319 0.998241
3.96748 2.35427 3.16694 0.99515
1.05767 0.857971 5.47666 4.95726
0.970195 0.758641 4.45319 3.96941
1.68027 1.17494 4.91838 4.0446
```

*Figura 6. Nueva matriz R de puntuaciones, con puntajes predichos*

Se observa como las puntuaciones originales discrepan muy poco de las nuevas, y en donde habían ceros ahora hay puntajes predichos.

Ahora se muestran las pruebas de ejecución del primer procedimiento con la tabla de puntuaciones real. Se realizó el proceso de recomendación con cuatro usuarios diferentes, en cada caso variando el número de factores latentes (k) entre dos valores, 2 y 5.

**Primer usuario: usuario con ID 68436**

**Libros calificados originalmente:**

booktitle	bookrating	userid
The Two Towers (The Lord of the Rings, Part 2)	8	68436
The Return of the King (The Lord of the Rings, Part 3)	8	68436
Fahrenheit 451	9	68436
The Andromeda Strain	8	68436

**k=2:**

```
Recommendations for user: 68436
-----
Running with Scissors: A Memoir - Predicted score: 9.68016
Big Stone Gap: A Novel (Ballantine Readers Circle) - Predicted score: 9.60056
A Child Called "It": One Childs Courage to Survive - Predicted score: 9.48861
Enders Game (Ender Wiggins Saga (Paperback)) - Predicted score: 9.46427
STONES FROM THE RIVER - Predicted score: 9.37894
Recommend: success

real    3m9,364s
user    0m38,435s
sys      0m2,566s
```

**k=5:**

```
Recommendations for user: 68436
-----
The Witching Hour (Lives of the Mayfair Witches) - Predicted score: 9.96246
I Know This Much Is True (Oprahs Book Club) - Predicted score: 9.92186
The Handmaids Tale - Predicted score: 9.70789
The Alchemist: A Fable About Following Your Dream - Predicted score: 9.31557
What Looks Like Crazy On An Ordinary Day - Predicted score: 9.22411
Recommend: success

real    3m31,499s
user    1m4,434s
sys      0m2,652s
```

**Segundo usuario: usuario con ID 278418**

**Libros calificados originalmente:**

booktitle	bookrating	userid
Dance upon the Air (Three Sisters Island Trilogy)	8	278418

**k=2:**

```
Recommendations for user: 278418
-----
Outlander - Predicted score: 9.06785
Enders Game (Ender Wiggins Saga (Paperback)) - Predicted score: 9.04016
Angelas Ashes (MMP) : A Memoir - Predicted score: 8.87701
Prodigal Summer: A Novel - Predicted score: 8.87592
The Princess Bride: S Morgensterns Classic Tale of True Love and High Adventure - Predicted score: 8.8304
Recommend: success

real    3m5,889s
user    0m37,129s
sys      0m2,818s
```

k=5:

```
Recommendations for user: 278418
-----
The Notebook - Predicted score: 8.18201
The Nanny Diaries: A Novel - Predicted score: 8.13576
Animal Farm - Predicted score: 8.03164
Face the Fire (Three Sisters Island Trilogy) - Predicted score: 8.00767
The Princess Bride: S Morgensterns Classic Tale of True Love and High Adventure - Predicted score: 7.99964
Recommend: success

real    3m34,028s
user    1m4,845s
sys      0m2,764s
```

**Tercer usuario: usuario con ID 86720**  
**Libros calificados originalmente:**

booktitle	bookrating	userid
The Divine Secrets of the Ya-Ya Sisterhood: A Novel	6	86720
Tell No One	8	86720

k=2:

```
Recommendations for user: 86720
-----
The Princess Bride: S Morgensterns Classic Tale of True Love and High Adventure - Predicted score: 8.93068
Good Omens - Predicted score: 8.84185
Holes (Yearling Newbery) - Predicted score: 8.73726
Seabiscuit: An American Legend - Predicted score: 8.70246
The Return of the King (The Lord of the Rings, Part 3) - Predicted score: 8.69793
Recommend: success

real    3m8,597s
user    0m38,079s
sys      0m2,692s
```

k=5:

```
Recommendations for user: 86720
-----
Watership Down - Predicted score: 9.4184
The Hours : A Novel - Predicted score: 9.396
The Horse Whisperer - Predicted score: 8.66262
The Clan of the Cave Bear (Earths Children (Paperback)) - Predicted score: 8.56224
Cold Mountain - Predicted score: 8.49028
Recommend: success

real    3m34,564s
user    1m4,593s
sys      0m2,817s
```

**Cuarto usuario: usuario con ID 218552**  
**Libros calificados originalmente:**

booktitle	bookrating	userid
Divine Secrets of the Ya-Ya Sisterhood: A Novel	5	218552
Fast Food Nation: The Dark Side of the All-American Meal	7	218552
Four To Score (A Stephanie Plum Novel)	6	218552
White Oleander : A Novel	5	218552
The Fellowship of the Ring (The Lord of the Rings, Part 1)	5	218552
The Da Vinci Code	8	218552
A Time to Kill	6	218552
The Client	8	218552
Message in a Bottle	3	218552
2nd Chance	7	218552
Ishmael: An Adventure of the Mind and Spirit	5	218552
Two for the Dough	8	218552
Shes Come Undone (Oprahs Book Club)	5	218552
Angels & Demons	8	218552
Girl, Interrupted	4	218552
ANGELAS ASHES	5	218552

k=2

```
Recommendations for user: 218552
-----
Outlander - Predicted score: 7.81976
The Sweet Potato Queens Book of Love - Predicted score: 7.72859
Dreamcatcher - Predicted score: 7.56772
The Cider House Rules - Predicted score: 7.51946
Snow Falling on Cedars - Predicted score: 7.09724
Recommend: success

real    3m6,686s
user    0m38,181s
sys      0m2,795s
```

k=5

```
Recommendations for user: 218552
-----
The Poisonwood Bible: A Novel - Predicted score: 8.20777
Into Thin Air : A Personal Account of the Mt. Everest Disaster - Predicted score: 7.64821
The Sweet Potato Queens Book of Love - Predicted score: 7.57036
The Prince of Tides - Predicted score: 7.54353
The Bonesetters Daughter - Predicted score: 7.52331
Recommend: success

real    3m33,403s
user    1m5,202s
sys      0m2,583s
```

En cada proceso de recomendación se escogieron los libros con los cinco puntajes predichos más altos para ser recomendados. Como puede observarse, la variación en el número de factores latentes hace que los resultados de la recomendación discrepen significativamente. Una explicación de este fenómeno es que a mayor número de factores latentes, el sistema de recomendación puede establecer más puntos en común entre los usuarios y/o los libros. Así, puede estimarse que cuando se utilizan 5 factores latentes, los libros recomendados podrían parecerse más entre sí en términos de poseer más cosas en común, como género, trama, contexto social al momento de su publicación, entre otras.

### 3.2 Resultados del segundo método – Sistema de recomendación basado en filtro colaborativo basado en ítems mediante coeficiente de correlación de Karl Pearson

Ahora se muestran las recomendaciones conseguidas aplicando la segunda implementación, para los mismos usuarios del primer procedimiento:

**Primer usuario: usuario con ID 68436:**

```
Recommendations for user: 68436
-----
The Fellowship of the Ring (The Lord of the Rings, Part 1) - Predicted score: 1.4671
Beloved (Plume Contemporary Fiction) - Predicted score: 1.19178
The Hobbit : The Enchanting Prelude to The Lord of the Rings - Predicted score: 1.13112
Rising Sun - Predicted score: 1.03067
Digital Fortress : A Thriller - Predicted score: 1.02499
Recommend: success

real    2m42,582s
user    0m14,194s
sys      0m2,649s
```



### Segundo usuario: usuario con ID 278418:

```
Recommendations for user: 278418
-----
Face the Fire (Three Sisters Island Trilogy) - Predicted score: 4.85845
Heaven and Earth (Three Sisters Island Trilogy) - Predicted score: 4.69289
Four To Score (A Stephanie Plum Novel) - Predicted score: 0.931707
The Divine Secrets of the Ya-Ya Sisterhood: A Novel - Predicted score: 0.850459
Three To Get Deadly : A Stephanie Plum Novel (A Stephanie Plum Novel) - Predicted score: 0.7982
Recommend: success

real    2m59,591s
user    0m14,184s
sys     0m2,625s
```

### Tercer usuario: usuario con ID 86720:

```
Recommendations for user: 86720
-----
The Saving Graces: A Novel - Predicted score: 1.47808
When the Wind Blows - Predicted score: 1.31144
What Looks Like Crazy On An Ordinary Day - Predicted score: 0.967853
B Is for Burglar (Kinsey Millhone Mysteries (Paperback)) - Predicted score: 0.965464
A Walk to Remember - Predicted score: 0.884143
Recommend: success

real    2m44,624s
user    0m14,233s
sys     0m2,592s
```

### Tercer usuario: usuario con ID 218552:

```
Recommendations for user: 218552
-----
High Five (A Stephanie Plum Novel) - Predicted score: 0.803516
Three To Get Deadly : A Stephanie Plum Novel (A Stephanie Plum Novel) - Predicted score: 0.763141
The Pelican Brief - Predicted score: 0.745367
Seven Up (A Stephanie Plum Novel) - Predicted score: 0.663789
Hot Six : A Stephanie Plum Novel (A Stephanie Plum Novel) - Predicted score: 0.618156
Recommend: success

real    2m42,664s
user    0m14,184s
sys     0m2,551s
```

Las recomendaciones obtenidas con la segunda implementación difieren de las conseguidas con la primera implementación. Esto refuerza la idea de que los resultados varían mucho en función del método empleado para recomendar. Como anotación, en este caso se nota el uso de la matriz de similitudes: para el usuario 68436 aparecen dos recomendaciones parecidas: un libro del Hobbit y uno del Señor De Los Anillos.

Como fue mencionado en la descripción de los procedimientos, el tiempo de ejecución es menor para la segunda implementación, con una media de 30 segundos menos por recomendación.

## 4 CONSIDERACIONES FINALES

La implementación de dos sencillos sistemas de recomendación ha permitido observar cómo la aplicación de métodos numéricos permite tratar volúmenes grandes de información, modelados a través de matrices, para conseguir datos nuevos. En este caso específico, en el primer procedimiento el gradiente descendiente prueba ser una herramienta potente en el ámbito de la factorización de matrices por medio de iteraciones, en cuanto hace posible minimizar el error cada vez más en cada iteración yendo en la dirección opuesta del gradiente

calculado. El método numérico del gradiente descendiente es en este caso el núcleo del procedimiento, ya que su aplicación lleva al cálculo de las matrices P y Q cuyo producto incorpora las predicciones buscadas.

Los sistemas de recomendación constituyen importantes instrumentos no solo para la obtención de recomendaciones para usuarios, sino también para el descubrimiento de información. Los procedimientos implementados, en especial el primero, muestran cómo a través de cálculos sencillos se pueden obtener valores ausentes anteriormente que exhiban congruencia con la información explícita provista por usuarios.

Las recomendaciones obtenidas a través de sistemas de recomendación varían en gran medida conforme lo hacen los mecanismos utilizados y los parámetros de los mismos. Asimismo, es de notar la validez de la noción de que la efectividad de los mismos está íntimamente ligada con la cantidad de factores que se tienen en cuenta. Por ello, sistemas de recomendación que incluyen en sus cálculos factores más que solo los explícitos (ej: puntuación dada por usuarios), como la ubicación y las condiciones climáticas, pueden alcanzar grados más altos de precisión en virtud de la incorporación de estas condiciones de naturaleza variable.

## REFERENCIAS

- [1] Sciencedirect.com, 'Recommendation systems: Principles, methods and evaluation', 2015. [Online]. Disponible: <https://www.sciencedirect.com/science/article/pii/S1110866515000341>. [Recuperado el 25/05/2018].
- [2] [www.informatik.uni-freiburg.de](http://www.informatik.uni-freiburg.de), 'Book-Crossing Dataset'. [Online]. Disponible: <http://www2.informatik.uni-freiburg.de/~cziegler/BX/>. [Recuperado el 05/05/2018].
- [3] <http://www.quuxlabs.com>, 'Matrix Factorization: A Simple Tutorial and Implementation in Python', 2010. [Online]. Disponible: <http://www.quuxlabs.com/blog/2010/09/matrix-factorization-a-simple-tutorial-and-implementation-in-python/>. [Recuperado el 05/05/2018].
- [4] <http://blog.findemor.es/>, 'Sistemas de recomendación en Python', 2018. [Online]. Disponible: <http://blog.findemor.es/2018/02/sistemas-de-recomendacion-en-python/>. [Recuperado el 18/05/2018].