Evaluation of standard SVD-based techniques for Collaborative Filtering

Manolis Vozalis Angelos Markos
Konstantinos Margaritis

Department of Applied Informatics, University of Macedonia,
156 Egnatia Street, P.O. Box 1591, 54006, Thessaloniki, Greece

Abstract

In this paper, we present and evaluate three SVD-based algorithms that can be used for the least-squares approximation of the user-item ratings matrix in the context of Collaborative Filtering (CF). The algorithms are adaptations of three standard techniques for matrix factorization: Standard Singular Value Decomposition (sSVD), Principal Component Analysis (PCA) and Correspondence Analysis (CA). PCA and CA can be described as singular value decompositions of appropriately transformed matrices. CA is conceptually similar to PCA and treats rating data using a consistent weighting scheme. To our knowledge, CA has not been previously used in the context of CF. The main purpose of this paper is to test how the corresponding CF variations behave under rating data sets of different sparsity and size. A series of experiments were executed, aiming to evaluate the three algorithms in terms of prediction accuracy. Experimental results on two benchmark data sets, MovieLens and Jester, indicated that CA-Cf outperforms SVD-Cf and PCA-Cf in terms of Mean Absolute Error (MAE) for small numbers of retained dimensions, regardless of the different data set characteristics. Nevertheless, SVD-Cf displayed the overall highest accuracy.

Keywords: Singular Value Decomposition, Collaborative Filtering, Principal Component Analysis, Correspondence Analysis

1. Introduction

Collaborative Filtering (CF) is a popular approach employed by Recommender Systems (RSs), a term used to describe intelligent techniques that generate personalized recommendations. The premise of CF is that users who have agreed in the past tend to agree in the future. A common approach to collaborative prediction is to fit a factor model to the original rating data, and use it in order to make further predictions [1, 2]. The goal of a factor model is to uncover latent features that explain user preferences. That is achieved by approximating the observed user preferences in a low dimensionality space. In this paper, we will focus on methods based on Singular Value Decomposition (SVD), a well-known technique for approximating matrices of a given rank.

Several studies described and evaluated the performance of various matrix factorization techniques in the context of Collaborative Filtering, including both plain and similarity/neighborhood based approaches [3, 4, 5, 6]. In [5], an SVD approach is proposed that incorporates demographic information and ratings to enhance the plain CF algorithm. Goldberg et al. utilized Principal Component Analysis (PCA) as a preprocessing step in a recommendation procedure, followed by a recursive rectangular clustering method [3]. Alternatively, PCA can be iteratively performed until convergence and users are clustered based on their scores in the reduced space [7]. In [4] a complex scheme is described for combining predictions generated by SVD extensions applied to the Netflix data set. In addition to a straightforward optimization approach for the general low-rank approximation problem, many authors proposed sophisticated algorithms to cope with the missing ratings problem [7, 4, 6].

In this paper, we describe and compare three CF algorithms that can be used for the low-rank approximation of the user-item ratings matrix, in terms of least squares. The algorithms are adaptations of three baseline techniques for fitting a factor model to the data: Standard Singular Value Decomposition (sSVD), Principal Component Analysis (PCA) and Correspondence Analysis (CA). PCA is the basis for linear least-squares approximation of a data matrix, while CA can be regarded as a particular case of non-linear, weighted PCA [8]. CA considerably expands the scope of a PCA-type analysis in its ability to handle a wide range of data. The main contribution of this paper lies in the application of CA in the context of CF. Notice that the crucial step of all these methods is the SVD of an appropriately transformed matrix. In other words, the three CF algorithms described in the following sections share the SVD as an algorithmic engine for dimension reduction and prediction generation.

Our purpose is to test how the three algorithms behave in terms of accuracy under two publicly available data sets of different size and sparsity. For each algorithm we implement a direct rating prediction scheme based on the reduced or reconstructed user-item ratings matrix. This matrix contains the predicted likeliness of each item by each user.

The paper is organized as follows: Section 2 is devoted to the brief presentation of the three standard SVD-based matrix factorization techniques. The CF versions of each method are thoroughly described in Section 3. The efficiency of each approach is demonstrated in Section 4 through a set of experiments on two real-world ratings data sets. The paper concludes in Section 5.

2 A family of SVD-based methods

In this section we present three SVD-based low-rank approximation techniques which were selected for the implementation of the proposed approaches: *Standard Singular Value Decomposition*, *Principal Component Analysis* and *Correspondence Analysis*. We will focus more on the latter method, since it is the first time, to our knowledge, that CA is utilized in the context of collaborative filtering.

2.1 Standard Singular Value Decomposition

Standard *Singular Value Decomposition* (sSVD) is based on a well known matrix factorization method which takes an $m \times n$ matrix $\mathbf{A}$, with rank r, and decomposes it as follows [9]:

$$\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathbf{T}}$$

$\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices with dimensions $m \times m$ and $n \times n$ respectively. $\mathbf{S}$, called the *singular matrix*, is an $m \times n$ diagonal matrix whose diagonal entries are non-negative real numbers.

The initial r diagonal entries of $\mathbf{S}$ $(s_1, s_2, \dots, s_r)$ have the property that $s_i > 0$ and $s_1 \geq s_2 \geq \dots \geq s_r$. Accordingly, the first r columns of $\mathbf{U}$ are eigenvectors of $\mathbf{A}\mathbf{A}^{\mathbf{T}}$ and represent the left singular vectors of $\mathbf{A}$, spanning the column space. The first r columns of $\mathbf{V}$ are eigenvectors of $\mathbf{A}^{\mathbf{T}}\mathbf{A}$ and represent the right singular vectors of $\mathbf{A}$, spanning the row space. If we focus only on these r nonzero singular values, the effective dimensions of the SVD matrices $\mathbf{U}$, $\mathbf{S}$ and $\mathbf{V}$ will become $m \times r$, $r \times r$ and $r \times n$ respectively.

The sSVD provides a robust computational method for the low-rank approximation of the original data matrix, $\mathbf{A}$ in terms of the 2-norm and Frobenius norm [9] and can be a particularly useful technique in the context of Recommender Systems [10]. By retaining the $k \ll r$ largest singular values of $\mathbf{S}$ and discarding the rest, we reduce the data dimensionality and expect to capture the underlying latent structure of the original data.

2.2 Principal Component Analysis

Principal Component Analysis, or PCA, is a multivariate data analysis technique aiming at the low-rank least-squares approximation of a data matrix and has been extensively used in survey analysis, medical imaging, lossy data compression and feature extraction [11, 3, 7]. PCA reduces data dimensionality by optimally projecting highly correlated data along a smaller number of orthogonal dimensions. Its objective is to find a (linear) transformation of the original variables to a set of new uncorrelated variables (the principal components) such that a very high proportion of the variation of the old variables is captured by relatively few of the new ones. A comprehensive overview of the theory and applications of PCA can be found in [11].

In practice, classical PCA involves the calculation of the eigenvalue decomposition of the data covariance matrix. The eigenvalues of the covariance matrix indicate the amount of variance along the direction given by the corresponding eigenvector. The SVD offers an alternative viewpoint to some aspects of the PCA theory. The PCA solution can be found by first computing the SVD of the original mean centered data matrix $\mathbf{A}$, i.e., $\mathbf{A} = \mathbf{USV^T}$ with $\mathbf{U^T U} = \mathbf{V^T V} = \mathbf{I}$, where $\mathbf{U}$ contains the variables' loadings for the principal components and $\mathbf{S}$ has the corresponding variances along the diagonal [11]. A reduction to k dimensions is obtained by projecting the original data matrix on the subspace consisting of eigenvectors corresponding to the largest k eigenvalues of the covariance matrix.

2.3 Correspondence Analysis

Correspondence Analysis (CA) is mainly considered as a non-linear multivariate data analysis method, suitable for exploring the associations between two or more, non-metric, categorical variables without a priori hypotheses or assumptions. CA was originally used for the analysis and visualization of data arriving typically from fields of social sciences and biometrics [8]. However, the method has recently attracted the attention of the engineering community from a number of disciplines, including machine learning, image retrieval and data mining [12, 13, 14]. Recently, the method has been described as a preprocessing step for pattern recognition [15]. In practice, CA can be performed to analyze almost any type of tabular data after suitable transformation or recoding [8]. The only input requirement of the method is a matrix with non-negative entries, with at least one non-zero entry to each row and each column.

CA can be described as a particular case of weighted PCA [16, 12]. Similar to PCA, the rows or columns of a data matrix are assumed to be points in a high-dimensional space in which distance is measured by a weighted Euclidean metric and the points themselves have differential weights, called "masses". The method aims to highlight both visible and hidden relations in the data structure by mapping the original data onto lower-dimensional maps, so that the principal dimensions (usually two or three) capture the most variance possible. These dimensions can be considered as latent constructs or new composite quantitative variables, with metric properties, that summarize the original multi-dimensional information. In that sense, CA can be viewed as a method that quantifies qualitative data, with a simultaneous dimensionality reduction [16, 8].

In the CA context, the SVD provides a straightforward mechanism of approximating the so-called standardized residuals matrix with another matrix of lower rank, via non-linear weighted least squares [8, 17]. All numerical results of the method are obtained directly from the SVD.

3 SVD-based Collaborative Filtering Algorithms

In this section we will describe how the aforementioned matrix factorization techniques can be combined with collaborative filtering in order to make direct prediction generation both scalable and effective.

3.1 SVD-Cf

Following Sarwar et al. [18], we describe the sSVD algorithm in a CF context in order to capture latent relationships between users and items that allow us to compute the predicted likeliness of a certain item by a user.

Step 1. Data representation

1. Define the original user-item matrix, $\mathbf{R}$, of size $m \times n$, which includes the ratings of m users on n items. r_{ij} refers to the rating of user u_i on item i_j.
2. Preprocess user-item matrix $\mathbf{R}$ in order to impute the missing data. The preprocessing is described as follows:
   (a) Compute the average of each row, $\bar{r}_i$, where $i = 1, 2, ..., m$, and the average of each column, $\bar{c}_j$, where $j = 1, 2, ..., n$, from the user-item matrix, $\mathbf{R}$.
   (b) Replace all missing values with the corresponding *column* average, $\bar{c}_j$, which leads to a new filled-in matrix, $\mathbf{R}_f$.
   (c) Subtract the corresponding *row* average, $\bar{r}_i$, from $\mathbf{R}_f$, and obtain the row centered matrix $\mathbf{A}$ i.e. the row mean of $\mathbf{A}$ is 0.

Step 2. Low-rank approximation

Compute the SVD of $\mathbf{A}$ and keep only the first k diagonal entries from matrix $\mathbf{S}$ to obtain a $k \times k$ matrix, $\mathbf{S}_k$. Similarly, matrices $\mathbf{U}_k$ and $\mathbf{V}_k$ of size $m \times k$ and $k \times n$ are generated. The reduced or reconstructed matrix is denoted as $\mathbf{A}_k$.

Step 3. Prediction generation

The predicted rating for user u_i on item i_j is given by:

$$pr_{ij} = \bar{r}_i + \mathbf{U}_k \sqrt{\mathbf{S}_k^{\mathbf{T}}}(i) \sqrt{\mathbf{S}}_k \mathbf{V}_k^{\mathbf{T}}(j),$$

The second part of the equation gives the corresponding element of the reduced matrix $\mathbf{A}_k$. The prediction is generated by adding the mean of the appropriate row, $\bar{r}_i$, to this element.

3.2 PCA-Cf

PCA is implemented in a CF framework, in a way similar to the one described in the previous section.

Step 1. Data representation

1. Impute the missing values in the original user-item matrix, $\mathbf{R}$, with the corresponding *row* average, $\bar{r}_i$, which leads to a new filled-in matrix, $\mathbf{R}_f$.
2. Subtract the corresponding *column* average, $\bar{c}_j$, from $\mathbf{R}_f$, and obtain the column centered matrix $\mathbf{A}$ i.e. the column mean of $\mathbf{A}$ is 0.

Step 2. Low-rank approximation

Compute the SVD of $\mathbf{A}$ and keep only the first k eigenvalues. This is equivalent to the eigenvalue decomposition of the covariance matrix $\frac{1}{m-1}\mathbf{A}^{\mathbf{T}}\mathbf{A}$ [11]. The reduced or reconstructed matrix is denoted as $\mathbf{A}_k$.

Step 3. Prediction generation

The predicted rating for user u_i on item i_j is given by:

$$pr_{ij} = \bar{c}_j + \mathbf{U}_k \sqrt{\mathbf{S}_k^{\mathbf{T}}}(i) \sqrt{\mathbf{S}}_k \mathbf{V}_k^{\mathbf{T}}(j),$$

The second part of the equation gives the corresponding element of the reduced matrix $\mathbf{A}_k$. The prediction is generated by adding the mean of the appropriate column, $\bar{c}_j$, to this element.

3.3 CA-Cf

The theory of CA is based on the generalized SVD of the ratings matrix in a weighted least squares sense.

Step 1. Data representation

Impute the missing values in the original user-item matrix, $\mathbf{R}$, with the corresponding *column* average, $\bar{c}_j$, which leads to a new filled-in matrix, $\mathbf{R}_f$.

Step 2. Low-rank approximation

Compute the SVD of $\mathbf{A} = \mathbf{D}_q^{-1/2}(\mathbf{R}_f - \mathbf{q}\mathbf{w}^{\mathbf{T}})\mathbf{D}_w^{-1/2}$, which is known as the standardized residuals matrix of $\mathbf{R}_f$ [8], where $\mathbf{q}$ and $\mathbf{w}$ are the vectors with the row and column marginal relative frequencies and $\mathbf{D}_q = diag(\mathbf{q})$, $\mathbf{D}_w = diag(\mathbf{w})$. The reduced or reconstructed matrix, keeping the first k dimensions, is denoted as $\mathbf{A}_k$.

Step 3. Prediction generation

The predicted rating for user u_i on item i_j is given by:

Table 1. Brief description of the three CF algorithms

Algorithm	Low-rank approximation	Prediction generation
SVD-Cf	$\mathbf{A} = \mathbf{R}_f - \bar{\mathbf{r}}\mathbf{1}_m$ †	$pr_{ij} = \bar{r}_i + \mathbf{U}_k\sqrt{\mathbf{S}_k^{\mathbf{T}}}(i)\sqrt{\mathbf{S}}_k\mathbf{V}_k^{\mathbf{T}}(j)$
PCA-Cf	$\mathbf{A} = \mathbf{R}_f - \bar{\mathbf{c}}\mathbf{1}_n$	$pr_{ij} = \bar{c}_j + \mathbf{A}_k = \mathbf{U}_k\sqrt{\mathbf{S}_k^{\mathbf{T}}}(i)\sqrt{\mathbf{S}}_k\mathbf{V}_k^{\mathbf{T}}(j)$
CA-Cf	$\mathbf{A} = \mathbf{D}_q^{-1/2}(\mathbf{R}_f - \mathbf{q}\mathbf{w}^{\mathbf{T}})\mathbf{D}_w^{-1/2}$	$pr_{ij} = q_iw_j + \mathbf{D}_q^{1/2}\mathbf{U}_k\sqrt{\mathbf{S}_k^{\mathbf{T}}}(i)\mathbf{D}_w^{1/2}\sqrt{\mathbf{S}}_k\mathbf{V}_k^{\mathbf{T}}(j)$

† $\bar{\mathbf{r}}$ is the column vector of row averages and $\mathbf{1}_m$ is a row vector of ones of size m

$$pr_{ij} = q_iw_j + \mathbf{D}_q^{1/2}\mathbf{U}_k\sqrt{\mathbf{S}_k^{\mathbf{T}}}(i)\mathbf{D}_w^{1/2}\sqrt{\mathbf{S}}_k\mathbf{V}_k^{\mathbf{T}}(j)$$

The equation gives the corresponding element of the reduced matrix $\mathbf{A}_k$.

A brief description of the three CF algorithms is given in Table 1.

4 Experiments

In this section we first provide a brief description of the various experiments that we executed in order to evaluate and compare the proposed CF methods, and then present and comment on their results.

The experiments were run on two publicly available data sets, MovieLens and Jester. The selection of the data sets was based on their significant differences in size and sparsity, which allowed us to evaluate whether the behavior of the proposed approach is affected by these characteristics. MovieLens is a data set which was first presented by the GroupLens research group [19]. It consists of 100,000 ratings which were assigned by 943 users on 1682 movies. Ratings follow the 1(bad)-5(excellent) numerical scale. The sparsity of the data set is high, at a value of 93.7%. Jester is a data set initially introduced by Goldberg et al [3]. Its size is considerably larger than that of MovieLens, including 4.1 million ratings of 100 jokes from 73,421 users. Apart from their difference in size, Jester is a significantly denser data set. Its sparsity has a value of 44.16%. In both cases, starting from the initial data set, distinct splits of training (80%) and test (20%) data were utilized.

Mean Absolute Error (MAE) was the metric we employed to evaluate the accuracy of the methods [20]. MAE measures the deviation of predictions generated by the RSs from the true rating values, as they were specified by the user.

4.1 MovieLens results

The sole parameter altered during the execution of our experiments on the three SVD-based methods for the MovieLens data set was the number of reduced dimensions, k. Figure 1 depicts the MAE values obtained for different k values, ranging between 1 and 25.

Based on Figure 1, CA-Cf appears to be the most accurate among the three methods, for smaller values of k. Specifically, the results suggest that the CA-Cf algorithm is the best in terms of MAE for k = 2 to 5. However, the SVD-Cf displays the highest accuracy for larger values of k, with the best overall MAE (0.7895) achieved at a k of 12. A series of paired t-tests confirmed the statistical significance of differences between the SVD and PCA ($t(8) = 3.60$, $p < 0.01$), as well as the SVD and CA ($t(8) = 3.88$, $p < 0.01$).

4.2 Jester results

Based on Figure 2, it can be concluded that CA-Cf is the most accurate approach for lower values of k, and specifically when k is between 2 and 5. On the contrary, for larger values of k (k=6 ...25), SVD-Cf seems to be the method with the highest accuracy. It also achieves the best overall MAE, for a value of 0.8218 when k=11. Finally, PCA-Cf has a slightly worse accuracy than SVD-Cf, but significant statistical differences between the two approaches were not detected.

4.3 Comparing the two data sets

The experimental results for both data sets indicate that CA-Cf is the most accurate method for smaller values of retained dimensions, k=2 ... 5. As a result, we conclude that CA-Cf should be the method of choice in case the focus is on the more compact model rather than on the most accurate one. On the other hand, if we are mostly interested in higher accuracy, then

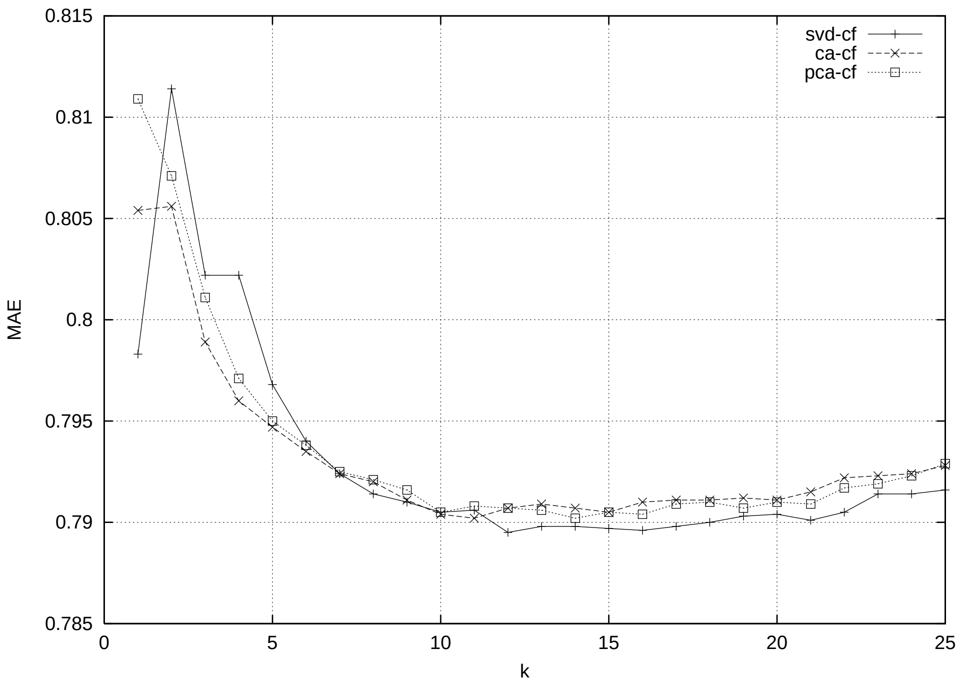

Figure 1. Accuracy of the three SVD-based approaches on the MovieLens data set

the final choice should be between the SVD-Cf and PCA-Cf methods, with the former having a slight edge, as documented by its outperforming the rest of the approaches for the MovieLens experiments.

5 Conclusions

In this paper we described three CF methods based on standard techniques for fitting a factor model to the data in a least squares sense. At the core of the three algorithms lies the SVD of an appropriately transformed user-item ratings matrix. For each method we implemented and evaluated a standard prediction scheme on two well-known rating data sets.

Results indicated that CA-Cf was the most accurate method for small numbers of retained dimensions, while the SVD-Cf approach performed better for the remaining values of k. This is an interesting remark considering the fact that CA has not been previously utilized in the CF framework. The property of CA to treat users and items in a simultaneous manner, utilizing a weighting scheme for categorical data, still needs to be explored. Further research issues include the extension of the proposed methods with some recently introduced SVD-based variations, as well as the description and evaluation of other methods which share the SVD as an algorithmic engine, in a CF context.

References

[1] Thomas Hofmann, "Latent semantic models for collaborative filtering," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 89–115, 2004.

[2] Jasson D. M. Rennie and Nathan Srebro, "Fast maximum margin matrix factorization for collaborative prediction," in *Proceedings of the 22nd international conference on Machine learning (ICML '05)*, Bonn, Germany, 2005, pp. 713–719.

[3] Ken Goldberg, Theresa Roeder, Dhruv Gupta, and Chris Perkins, "Eigentaste: A constant time collaborative filtering algorithm," *Information Retrieval Journal*, vol. 4, pp. 133–151, 2001.

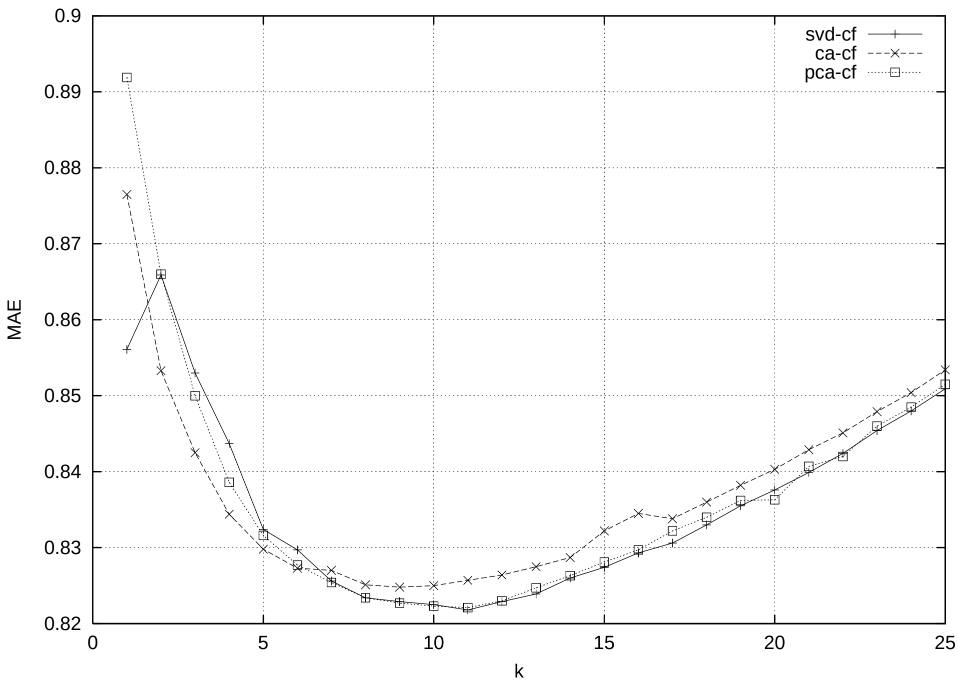

Figure 2. Accuracy of the three SVD-based approaches on the Jester data set

[4] Arkadiusz Paterek, "Improving regularized singular value decomposition for collaborative filtering," in *Proceedings of KDD Cup Workshop at SIGKDD'07, 13th ACM International Conference on Knowledge Discovery and Data Mining*, San Jose, CA, USA, 2007, pp. 39–42.

[5] Manolis G. Vozalis and Konstantinos G. Margaritis, "Using svd and demographic data for the enhancement of generalized collaborative filterin," *Information Sciences*, vol. 177, pp. 3017–3037, 2007.

[6] Sheng Zhang, Weihong Wang, James Ford, Fillia Makedon, and Justin Pearlman, "Using singular value decomposition approximation for collaborative filtering," in *Proceedings of the Seventh IEEE International Conference on E-Commerce Technology (CEC '05)*, Washington, DC, USA, 2005, pp. 257–264.

[7] Dohyun Kim and Bong-Jin Yum, "Collaborative filtering based on iterative principal component analysis," *Expert Systems with Applications*, vol. 28, pp. 823–830, May 2005.

[8] Michael Greenacre, *Correspondence Analysis in Practice (2nd Ed.)*, Chapman & Hall, Boca Raton, 2007.

[9] Gene H. Golub and Charles F. Van Loan, *Matrix Computations*, John Hopkins University Press, 3 edition, 1996.

[10] Badrul M. Sarwar, George Karypis, Joseph A. Konstan, and John T. Riedl, "Item-based collaborative filtering recommendation algorithms," in *10th International World Wide Web Conference (WWW10)*, Hong Kong, 2001, pp. 285–295.

[11] Ian T. Jolliffe, *Principal Component Analysis (2nd Ed.)*, Springer, 2002.

[12] Boris Mirkin, *Clustering For Data Mining: A Data Recovery Approach*, Chapman & Hall, London, 2005.

[13] Pierre Loslever, P. Simon, F. Rousseau, and Jean-Christophe Popieul, "Using space windowing for a preliminary analysis of complex time data in human component system studies. examples with eye-tracking in advertising and car/head movements in driving," *Information Sciences*, vol. 178, pp. 3645–3664, 2008.

[14] Nguyen-Khang Pham, A. Morin, P. Gros, and Le Quyet-Thang, "Factorial correspondence analysis for image retrieval," in *Proceedings of IEEE International Conference on Research - Innovation and Vision for the Future (RIVF'08)*, Ho Chi Minh City, 2008, pp. 269–275.

[15] Fionn Murtagh, A. Ganz, and S. McKie, "The structure of narrative: The case of film scripts," *Pattern Recognition*, vol. 42, pp. 302–312, 2009.

[16] Albert Gifi, *Non-Linear Multivariate Analysis*, John Wiley & Sons Ltd/Chichester, 1996.

[17] Shizuhiko Nishisato, *Multidimensional Nonlinear Descriptive Analysis*, Chapman & Hall/CRC, New York, 2007.

[18] Badrul M. Sarwar, George Karypis, Joseph A. Konstan, and John T. Riedl, "Application of dimensionality reduction in recommender systems - a case study," in *ACM WebKDD 2000 Web Mining for E-Commerce Workshop*, 2000, pp. 82–90.

[19] Badrul M. Sarwar, George Karypis, Joseph A. Konstan, and John T. Riedl, "Analysis of recommendation algorithms for e-commerce," in *2nd ACM Conference on Electronic Commerce*, 2000, pp. 158–167.

[20] Emmanouil Vozalis and Konstantinos G. Margaritis, "Analysis of recommender systems algorithms," in *Proceedings of the Sixth Hellenic-European Conference on Computer Mathematics and its Applications - HERCMA 2003*, 2003, pp. 732–745.