



Undirect knowledge discovery by using singular value decomposition

E. Maltseva¹, C. Pizzuti² & D. Talia²

¹DEIS, Università della Calabria, Rende, Italy.

²ISI-CNR, c/o DEIS, UNICAL, Rende, Italy.

Abstract

Clustering is an undirected knowledge discovery technique based on the partitioning of large sets of data objects into homogenous groups. All objects contained in the same group have similar characteristics. Grouping multivariate data is a difficult data mining task when no domain knowledge on data structure is available. In this paper we describe the use of a well known linear projection technique, called *singular value decomposition (SVD)*, to discover clusters in the pattern space by projecting it into a subspace that constitutes its best approximation and preserves the character of data. Experimental results on real datasets from the UCI Machine Learning repository assess the quality of the clustering obtained.

1 Introduction

Clustering is a data mining task [1] for unsupervised classification that consists in partitioning large sets of data objects into homogenous groups [2, 3, 4, 5]. All objects contained in the same group have similar characteristics. Clustering is an undirected knowledge discovery technique that is used to discover meaningful structures in data when no predefined knowledge is available. Generally, an object is described by a set of m attributes (features) and it can be represented by an m -dimensional vector, called *pattern*. Thus, if the dataset contains n objects, it can be viewed as an $n \times m$ *pattern matrix*. Rows of the matrix correspond to patterns and columns to features. Each pattern can then be considered as a point in an m -dimensional space, where each of the m attributes is one of the axis of the space. Clusters can be described as regions of this space containing points which are close to each other. This perspective of clusters directly reflects the process we use

to detect clusters visually in one, two or three dimensions. However, clustering algorithms must be able to detect such groups in multidimensional spaces. An approach to reduce the intrinsic complexity of finding groups in multivariate data is *linear projection* [5]. It consists of mapping a set of m -dimensional patterns onto a d -dimensional space, where $d < m$. A linear projection expresses the d features as linear combination of the original m features [5]: $y_i = \mathcal{H}x_i$ $i = 1, \dots, n$ where y_i is a d vector, $\mathcal{H}$ is a $d \times m$ matrix and x_i is an m vector. A main characteristic of linear projection algorithms is that they retain the structure of data. If an accurate two or three-dimensional approximation of the pattern matrix can be obtained, a visual perception of the existing groups could be used for many purposes, such as validate the results of clustering algorithms, provide initial seed information for partitional clustering algorithms, or try to define threshold values for the projected features to assign the original patterns to the discovered groups on the basis of the values of the projected features.

The main goal of this work is to use a well known linear projection algorithm named *Singular Value Decomposition (SVD)*, to discover groups in multivariate data and to evaluate the quality of clustering results on real datasets from the UCI Machine Learning repository [6] for which membership class is known. The experiments we have done on nine different datasets demonstrated that the SVD technique has been useful in obtaining a good clustering result for at least five datasets and showed how the SVD approach could be useful for data clustering. Because of space limits we describe here only results obtained on two datasets: *iris plant*, and *dermatology*. The paper is organized as follows. Section 2 introduces the SVD technique and section 3 discusses in detail the experiments we performed on the two datasets. Section 4 summarizes the results and concludes the paper.

2 Singular Value Decomposition

SVD is a powerful technique in matrix computation and analysis that has been introduced by Beltrami in 1873 [7]. More recently it has been used in several applications such as solving systems of linear equations, linear regression [8], pattern recognition [3], statistical analysis [9], data compression [10] and matrix approximation [11]. The SVD is based on linear algebra concepts and is defined as follows. A *singular value decomposition* of an $n \times m$ matrix X is any factorization of the form

$$X = U \times \Lambda \times V^T$$

where U is an $n \times n$ orthogonal matrix, V is an $m \times m$ orthogonal matrix and Λ is an $n \times m$ diagonal matrix with $\lambda_{ij} = 0$ if $i \neq j$. A matrix is said orthogonal if its columns are mutually orthogonal vectors of length 1, that is $x^T x = 1$, where x denotes a column of X . This means that $X^T X = I$, the identity matrix. For a square matrix A , a unit vector u and

a scalar λ that satisfy $Au = \lambda u$ are called *eigenvector* and *eigenvalue* of A . The diagonal elements λ_i are the *eigenvalues* of X . It has been shown that there exist matrices U and V such that the diagonal elements of Λ are sorted: $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m$. The eigenvalues λ_i are called *singular values* of X . The decomposition can equivalently be written as $XV = \Lambda U$. Geometrically this factorization defines a rotation of the axis of the vector space defined by X where V gives the directions, Λ the strengths of the dimensions and $U \times \Lambda$ the position of the points along the new axis. Intuitively, the U matrix can be viewed as a similarity matrix among the rows of X , i.e. the objects of the dataset, the V matrix as a similarity matrix among the columns of X , i.e. the features that describe an object, the Λ matrix gives a measure of how much the data distribution is kept in the new space [5]. The value d to choose for the subspace it should be such that

$$\sum_{i=1}^d \lambda_i / \sum_{i=1}^m \lambda_i \geq 0.95$$

This should assure that projection on few features is a good approximation of the original pattern matrix.

In the data mining area, *SVD* can be used to identify clusters by analyzing the U and V matrices. By visualizing the V matrix in no more than 3 dimensions it can give a guidance on the number of potential clusters. In fact, as already observed, each row of V corresponds to an attribute of the patters. If groups of attributes are similar and, thus, are plotted in the same region, they will probably contribute to form clusters characterized by these features. By visualizing the matrix $U \times \Lambda$ and considering only the first d dimensions, where $d \leq 3$, we obtain a compressed representation of the X matrix that approximates it at the best.

In the following section the experiments carried out by using the *SVD* technique for the analysis of different datasets are reported and discussed. Furthermore, tentative assignments of points to the discovered clusters is done by defining threshold values for the attributes. To asses the validity of the method, the confusion matrix for datasets is also given.

3 Experimental results

3.1 Iris plant

This dataset contains 150 instances of iris flowers that compose 3 groups, where each group refers to a type of iris plant: iris-versicolor, iris-verginica and iris-setosa. Each iris item is characterized by four attributes that give the petal and sepal features. The use of *SVD* on this dataset has shown good clustering results. Starting from the analysis of $\Lambda = [95.95, 17.72, 3.47, 1.88]$ matrix we can observe that one or two columns of U and V are sufficient for representing the dataset. Considering only the first dimension we verified

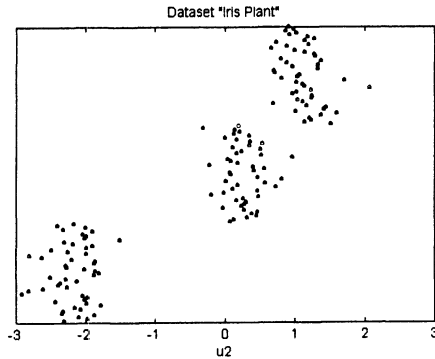


Figure 1: 1-D visualization of the iris dataset (2nd column of $\Lambda \times U$).

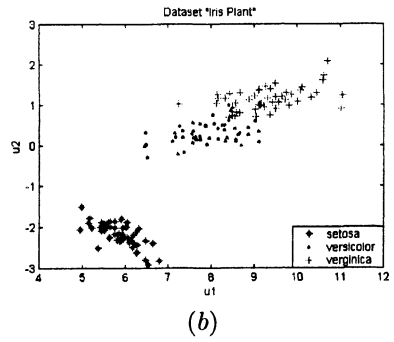
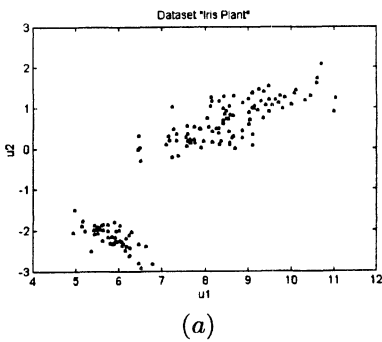


Figure 2: 2-D visualization of the iris dataset: (a) unique symbol, (b) one symbol per class.

that, by visualizing the first column of the matrix $U \times \Lambda$, we cannot distinguish any grouping, whereas by projecting the dataset along the second column of $U \times \Lambda$ (figure 1) it is possible to distinguish at least two clusters. By using a two-dimensional visualization of $U \times \Lambda$ we can see the same grouping of the previous step (figure 2a).

In particular, we have two groups of items. Actually, one of them contains two clusters, as shown in figure 2b, where different symbols are used to visualize the items of different clusters. Figure 2a is a representative visualization of the real features of this dataset. In fact, in it one class is linearly separable from the other two that are not linearly separable from each other. To verify this feature, we used SVD to find possible clusters in the 100 items that are grouped in the top right part of figure 2a.

The two-dimensional visualization of the $U \times \Lambda$ matrix (figure 3) for

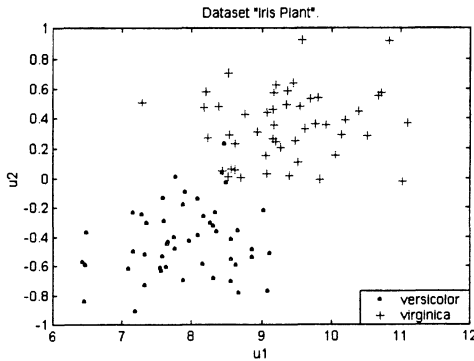


Figure 3: 2-D visualization of a subset the iris dataset.

Table 1. Confusion matrix for the Isir dataset.

Class 1	Class 2	Class 3	← Classified as
50			Class 1
	47	3	Class 2
		50	Class 3

this subset of data shows that it is possible to identify two groups in it. For a better reading of this result, in figure 3 we used two different symbols to visualize the items belonging to the two iris clusters that are not linearly separable from each other, but that can be clearly identified as two groups. On the basis of this result and that showed in figure 2 we defined the following thresholds for the three clusters ($u2 \leq -1$, $0.6 \geq u2 > -1$, $u2 > 0.6$). They allow for the assessment of our approach by estimating the error of SVD classification with respect to real classes. By using these thresholds we obtained the confusion matrix shown in table 1 that shows the difference between the actual classes and the clusters suggested by the SVD.

From table 1 we can see that only three instances are misclassified, thus SVD helped us to obtain a very good clustering of this dataset with an error equal to 0.02. Furthermore, we verified that the best useful visualization for this dataset is two-dimensional.

3.2 Dermatology

This dataset contains 358 instances each one with 33 attributes. Each instance contains the clinical and histopathological information about diagnosis of patients that suffer of erythemato-squamous diseases. All the patients should be grouped into six clusters.

By the analysis of the matrix $\Lambda = [105.440, 55.48, 32.67, 24.81, 19.57, 18.18, \dots]$,

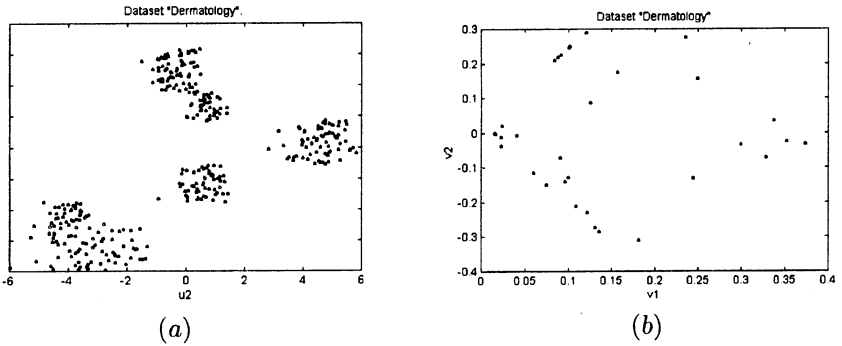


Figure 4: (a) 1-D visualization of $\Lambda \times U$, (b) V matrix of the dermatology dataset.

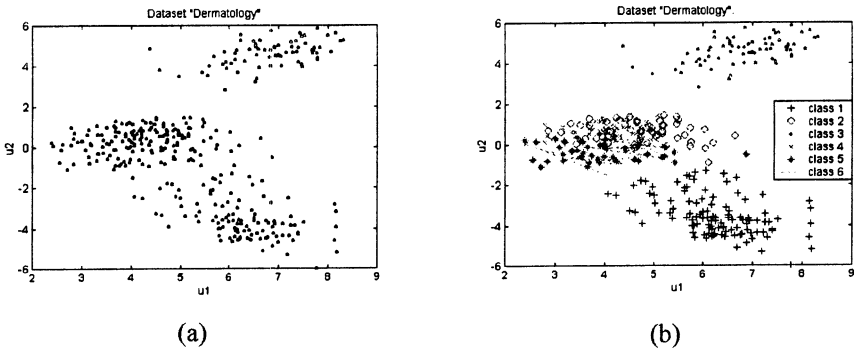


Figure 5: 2-D visualization of the dermatology dataset: (a) unique symbol, (b) one symbol per class.

we can observe that one, two or three columns of U and V could be used to represent the dataset. By visualizing the first two columns of V (figure 4b) we observed that the attributes seem to be separable into about five or six groups. This is a useful information for the clustering process.

By considering only one dimension of $U \times \Lambda$ we verified that by visualizing the first or the third column, the projection of items along these dimensions give a uniform distribution, so we cannot distinguish any grouping. However, by projecting the dataset along the second column of $U \times \Lambda$ (figure 4a) it is possible to distinguish at least three clusters.

Then, by using a two-dimensional visualization of $U \times \Lambda$ (figure 5a) we can see three clusters of items. Indeed the central cluster contains the items of four actual classes (figure 5b). However, at this stage we were not able to

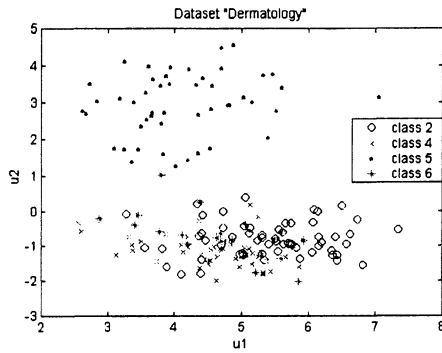


Figure 6: 2-D visualization of the subset of items of classes 2, 4, 5, and 6.

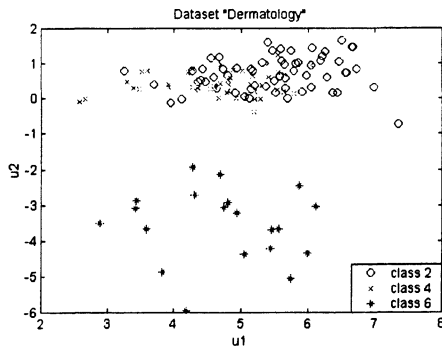


Figure 7: 2-D visualization of the subset of items of classes 2, 4, and 6.

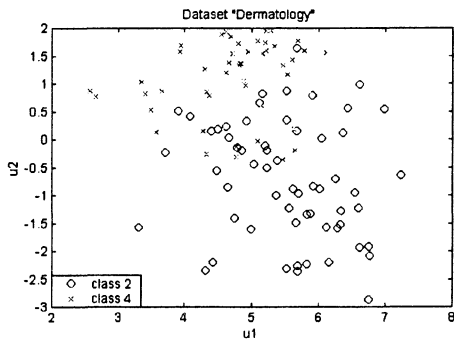


Figure 8: 2-D visualization of the subset of items of classes 2 and 4.

know if each or some of these clusters contained more than a class of items. To try to discover whether the clusters already identified contain different classes of items, we used a step-by-step approach based on the use of SVD on each one of the three clusters.

The SVD analysis on the 111 items that compose the top-right cluster shown in figure 5a resulted in a uniform distribution of the projection of items along the first two columns of the $U \times \Lambda$ matrix. The same result has been obtained applying SVD to the 71 items of the bottom-right cluster of figure 5a. This means that SVD suggests that the two clusters are homogeneous so they do not contain sub-clusters, as really occurs. Then we applied SVD to the 176 items that compose the central cluster. The Λ matrix is equal to [64.21, 24.34, 18.49, 12.30, 9.76, ...], so we used the first two columns of $U \times \Lambda$ to plot the items (figure 6). In the figure we can notice two clusters well separated by the threshold value $u_2 = 1$. Although, in the original figure the items was represented by the same symbol, in figure 6 (and in the next two) we used different symbols for items of different classes to allow the reader to do a better interpretation of the SVD results.

Actually, these results confirmed our hypothesis about the usefulness of SVD in discovering other clusters in subsets of data after the deletion of items that belong to the clusters that in the previous steps have been clearly identified. Proceeding step by step, we applied SVD to the 128 items that compose the bottom cluster of figure 6. Λ is equal to [58.13, 18.14, 13.50, 10.46, 8.97, ...] and figure 7 shows a 2-D visualization based on the first two columns of $U \times \Lambda$. Two groups of items are showed in it separated by the threshold value $u_2 = -1.5$. The bottom group, in particular, contains the items of class 6, whereas the top group consists of items of different classes. In the final step, we applied SVD on the 108 items of this group to verify if SVD is able to discover the two clusters that it contains indeed. As we can see in figure 8, that shows the plot of items using the first two columns of $U \times \Lambda$, SVD was not able to give an evident separation between the items of classes 2 and 4. We tried also to use different components of $U \times \Lambda$ and different dimensions for representing this datasets obtaining the same result.

In summary, at the beginning SVD discovered three clusters on this dataset. Then by a step-by-step deeper analysis of each of these clusters, SVD supported us in discovering 5 of the 6 real classes in which the items of this dataset are grouped. This result could be effectively used as input information for an eventual supervised classification step performed on the dataset with other data mining tools. This experiment is very positive because it demonstrates that SVD offers a user helpful clustering support also in the analysis of complex datasets whose clustering cannot be discovered in a single step.



4 Conclusions

In this paper we reported the experiences done by using the SVD technique for the analysis of different real datasets. In particular, due to lack of space, we discussed the use of SVD on the *iris plant* and *dermatology* databases. However, we tested this methodology also on the following databases of the UCI Machine Learning repository: *ionosphere*, *ecoli*, *Wisconsin breast cancer*, *Wisconsin diagnostic breast cancer*, *liver disorders*, and *congressional voting records*. The often positive and sometime negative results that we obtained in using SVD in such data domains showed that this technique could be useful in unsupervised clustering of data.

The output of the SVD technique is in the matrices Λ , V , and U . Thus when SVD is used for clustering it is necessary a detailed analysis of these matrices to discovery in them values that suggest how data can be visualized and interpreted to extract the structure of clusters from data. The values of the diagonal matrix Λ indicates the number of attributes that could be useful to consider to get the best approximation of data. The higher is the value of an element of Λ , the more significant is the corresponding column of U . On the basis of our experiments, the 1-D visualization of $\Lambda \times U$ allows for selecting the column that permits a better identification of distinct clusters. However, a single dimension is not ever sufficient; in general we experienced that 2-D representations of $\Lambda \times U$ gives the best way to identify clusters. Furthermore a step-by-step approach based on first applying the SVD on subsets of data that composed the clusters discovered at the previous step, and then analyzing the $\Lambda \times U$ matrix for these subsets, allowed us to discover clusters that previously were not visible. Finally, the analysis of the matrix V has also been useful because it suggested possible similarities among the attributes. The experiments on datasets with a high number of attributes demonstrate that the number of groups of attributes showed through the 2-D visualization of V , approximately corresponds to the number of clusters found in the dataset.

In conclusion, this work discussed a practical data mining methodology based on the SVD technique for discovering clusters when no predefined knowledge about data classes is available. The main goal of our work was to examine and evaluate SVD on real datasets to asses the technique. The results on the two datasets described here and on other datasets suggest that SVD is able to find clusters and represent them as groups of items in a d -dimensional space ($d \leq 3$) when these are separable in the original m -dimensional space. In many cases it was possible to define thresholds between classes and use them to validate the classification accuracy of SVD by means of confusion matrices. Finally, it is worth observing that this technique becomes compute intensive when it is used on very large datasets because it operates on very large matrices. The implementation of SVD on parallel computers can solve this problem by offering a high performance data mining tool for clustering.



References

- [1] Fayyad, U.M., Piatetsky-Shapiro, G. & Smith, P. From Data Mining to Knowledge Discovery: an overview. In U.M. Fayyad & al. (Eds) *Advances in Knowledge Discovery and Data Mining*, pp.1-34, AAAI/MIT Press, 1996.
- [2] Everitt, B. *Cluster Analysis*. Heinemann Educational Books Ltd, London, 1977.
- [3] Duda, R.O. & Hart, P.E. *Pattern Classification and Scene Analysis*. Wiley, New York, 1973.
- [4] Kaufman, L. & Rousseeuw, P.J. *Finding Groups in Data: an Introduction to Cluster Analysis*. John Wiley & Sons, 1990.
- [5] Jain, A.K. & Dubes, R.C. *Algorithms for Clustering Data*. Prentice Hall, 1988.
- [6] Blake, C.L. & Merz, C.J. UCI repository of Machine Learning. <http://www.ics.uci/mllearn/MLRepository.html>, Irvine, CA: University of California, Department of Information and Computer Science, 1998.
- [7] Beltrami, E. Sulle Funzioni Bilineari [On Bilinear Functions]. *Giornale di Matematiche ad Uso degli Studenti delle Università*, Vol. 11, pp. 98-106, 1873.
- [8] Strang, G. *Linear Algebra and its Applications*. Academic Press, 1980.
- [9] Jolliffe, I.T. *Principal Component Analysis*. Springer Verlag, 1986.
- [10] Korn, F., Jagadish, H.V. & Faloutsos, C. Efficient Supporting Ad Hoc Queries in Large Datasets of Time Sequences *Proceedings of the ACM Sigmod Conf. on Management of Data*, 1997.
- [11] Press, W. H., Teukolsky, S.A., Vetterling, W.T. & Flannery, B.P. *Numerical Recipes in C: The Art of Computing*. Cambridge University Press, 1992.