Research and Implementation of SVD in Machine Learning

YongchangWang
Communication University of China
Beijing, China
583696995@qq.com

Ligu Zhu
Communication University of China
Beijing, China
zhuligu@cuc.edu.cn

***Abstract*—With the arrival of the era of big data, people's ability to collect and obtain data is becoming more powerful. These data have shown the characteristics of high dimension, large scale and complex structure. High dimensional data has seriously hindered the efficiency of data mining algorithm, we call it "the Dimension disaster ". Therefore, dimension reduction technology has become the primary task of big data mining and machine learning. In this paper, we focus on the method of data reduction, described the category of data dimension reduction. The research status and main algorithms of dimension reduction method are described in detail. This paper briefly introduces the latest research progress of data dimension reduction algorithm, including some popular algorithm such as PCA, KPCA, SVD, etc. The principle of principal component analysis (PCA) is discussed in this article, and the singular value decomposition (SVD) theorem is introduced to solve the problem that the PCA method has a large amount of computation, we also give a comparison of PCA and SVD. Finally, we design and implement some experiments to verify the application of SVD in data analysis and latent semantic indexing.**

Keywords—Big data; Machine Learning; Dimension Reduction; PCA; SVD

I. INTRODUCTION

As the key step of data mining, the dimension reduction technology has been developed for many years, the method of dimension reduction is continuously evolved. In the environment of big data, research on dimensionality reduction has become a hot topic in the field of machine learning [1].

Dimension reduction is the mapping of data to a lower dimensional space such that uninformative variance in the data is discarded, or such that a subspace in which the data lives is detected. Dimension reduction has a long history as a method for data visualization, and for extracting key low dimensional features (for example, the two-dimensional orientation of an object, from its high dimensional image representation) [2]. In some cases the desired low dimensional features depend on the task at hand. Apart from teaching us about the data, dimension reduction can lead us to better models for inference. The need for dimension reduction also arises for other pressing reasons [3].

The subject of dimension reduction is vast, so we use the following criterion to limit the discussion: we restrict our attention to the case where the inferred feature values are continuous. The observables, on the other hand, may be continuous or discrete [4]. Thus this review does not address clustering methods, or, for example, feature selection for discrete data, such as text. This still leaves a very wide field, and so we further limit the scope by choosing not to cover probabilistic topic models (in particular, nonnegative matrix factorization, probabilistic latent semantic analysis, and Gaussian process latent variable models).

The popular methods of dimension reduction at present include: principal component analysis (PCA)[5], Singular value decomposition(SVD)[6],kernel PCA, probabilistic PCA, canonical correlation analysis (CCA), kernel CCA, Fisher discriminant analysis, oriented PCA, and several techniques for su□cient dimension reduction. The paper only research the principle and implementation of PCA and SVD.

The principle of PCA

The central idea of principal component analysis (PCA) is to reduce the dimensionality of a data set consisting of a large number of interrelated variables, while retaining as much as possible of the variation present in the data set. This is achieved by transforming to a new set of variables, the principal components (PCs), which are uncorrelated, and which are ordered so that the first few retain most of the variation present in all of the original variables [7].

Eigenvalues and eigenvectors are the two most important concepts in PCA. We think the matrix as a space, then for a symmetric positive definite matrix A, its eigenvectors are orthogonal to the space formed by A. The larger the eigenvalue is, the more information the corresponding eigenvector can have on this basis. For example, we select the feature values of the 10 largest eigenvalue of an image to reconstruct the image, as fig.1 shows, the graph is decomposed into 10 images(each feature vector corresponds to one image)[8].

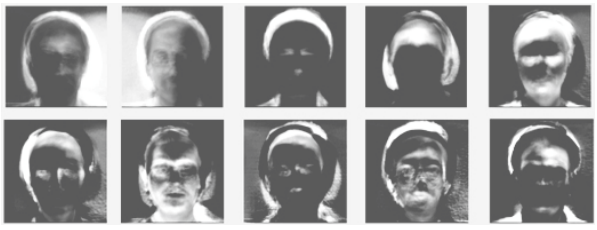

Fig. 1. The 10 images corresponding to the first 10 princple eigenvector

If the 10 images are superimposed, we can get a picture with not much difference to the original one, the space composed by the 10 feature vector contains 95% energy of the original image.

978-1-5090-5507-4/17/$31.00 ©2017 IEEE
ICIS 2017, May 24-26, 2017, Wuhan, China

The idea of PCA is to reduce the data dimension to the dimensions that make the variance of the data distribution maximum. Those dimensions that make the largest variance in the data distribution are the principal components. How to make the maximum variance and how to select these directions will be introduced in the following steps [9].

1. Moving the data center to 0 coordinates and normalize the data, that is, all the sample points X minus the mean and divided by the variance to obtain the normalized data.
2. Assume that the space after dimension reduction is P dimension, and now each sample point is projected to the P dimensional space, it is necessary to multiply the projection matrix P, set the P dimensional space orthogonal basis is U=[U1, U2,..,Up]. The projection matrix is $P=(U^TU)\text{-}1U^T$, U is an orthogonal matrix, so $P=U^T$, sample point Xi is project to the P dimensional space, the coordinates obtained in P dimensional space is $y=P\ x\ (\ I\ )=U^T\ x(\ I\ )$.
3. Now we get the coordinates of each sample point in the lower dimensional space , then we need to compute the variance, obviously, data variance after dimension reduction is $\sum_i y_i^T y^i$, the value y in step2 can be carried into $\sum_i y_i^T y^i$,then the variance of the data distribution can be represented as $U^T\Sigma U$ after computation, in which Σ is the covariance matrix of X.

Now we get a lower dimensional space, the matrix composed by P orthogonal basis of the space is U, the U represent the space. As for each new sample point x, we only need to project x to U, then our goal of dimension reduction can be achieved. Therefore, the vector after dimension reduction is $y=U^T x$.

The principle of SVD

Eigenvalue decomposition is a very good method to extract the characteristics of the matrix, but it is only effective for square matrix, however, most of the matrix is not square in the real world. For example, there are N students, each student has M scores, a N x M matrix would not be square. How can we describe the important features of such ordinary matrix? Singular value decomposition (shortly as VSD) can be used to do this, and singular value decomposition is a method which can be applied to any matrix. Let's look at the following formula:

$$A=U\Sigma VT \qquad (1)$$

If A is a matrix of M x N, U is a matrix of M x M (its vector is orthogonal, the vector in U is called left singular vector), Σ is a matrix of M x N (all elements are 0 in addition to the diagonal elements ,diagonal elements are known as the singular value) ,V^T(the transposition of V) is a matrix of N x N (its vector is orthogonal, the vector in V is called right singular vector), the fig.2 bellow can reflect the multiplication of these matrix.

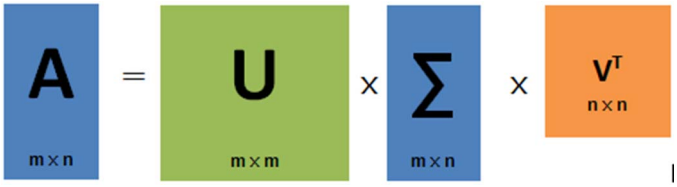

Fig. 2. The multiplication of these matrix

So how do singular values and eigenvalues correspond to each other? First of all, we can have a square matrix through the multiplication of A^T x A. Then we can calculate eigenvalues of the square matrix using the formula $(A^TA)Vi=\lambda_i \times Vi$, the V in this formula is the right singular vectors. In addition, we can also get the following variables $\sigma_i=\sqrt{\lambda_i}$ and $u_i=\frac{1}{\sigma_i}Av_i$:

The variable σ is the singular value said above and the variable u is the left singular vector said above. The singular value is similar with eigenvalue of matrix, in matrix Σ, its values are ordered from large to small and dropped very fast. In many cases, 10% or even 1% of the singular values accounted for the sum of 99% of singular values [10]. In other words, we can also use the singular value of the previous r to the describe the matrix approximately, here we can define partial singular value decomposition as the following formula:

$$A_{m\times n}\approx U_{m\times n}\Sigma_{m\times n}V^T_{r\times n} \qquad (2)$$

The number r is a much smaller than m and n, so the multiplication of the matrix looks like the fig3:

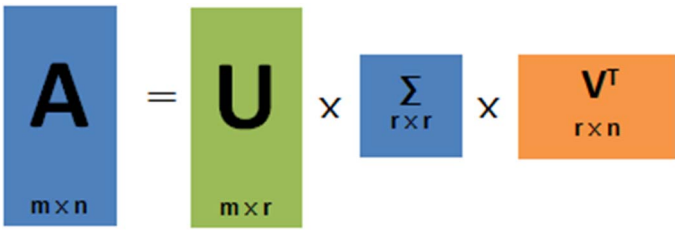

Fig. 3. The multiplication of these matrix

The result of multiplying the three matrices on the right will be a matrix that is close to A, where r is closer to n, the result of the multiplication is closer to A. The sum of the three matrix area is much smaller than area of the original matrix A (from the point view of storage, matrix area is smaller, the size of storage is smaller) [11]. So if we want to use compressed space to represent the original matrix A, we only need to save the three matrices: U, Σ and V.

Comparison of PCA and SVD

After introducing the method to compute the similarity, let's look at how to find the user-item neighbors according to similarity, the common principle of selecting neighbors can be divided into two categories: Figure 1 shows the set of points in two-dimensional space diagram

Look at the fig.4 bellow, if we want to use a straight line to fit these points, which direction shall we choose? Certainly, we will choose the line with signal on the map. If we project these

points simply to the X axis or Y axis, the variance in Y axis and in the X axis is similar (because these points has the trend of 45 degrees in the direction, so the projection to the X axis or Y axis are similar), If we use the original XY coordinates to look at these points, it is not easy to see what the real direction of these points [12]. But if we change the coordinate: make the horizontal axis into the direction of the signal axis and make the vertical axis into the direction of noise, it is easy to see the variance in what direction is large and in what direction is small [13].

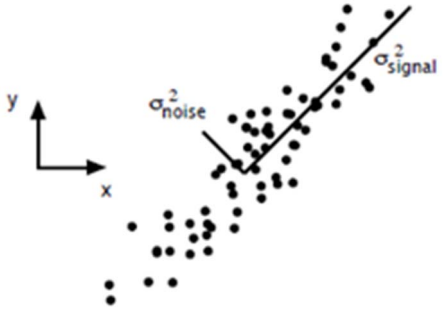

Fig. 4. The graph of the signal and the noise

Assume that we use each row of a matrix to represent a sample and use each column to represent a feature. For a matrix A of m x n ,we can use a matrix P of n x n to change the n dimensional space of A into another n dimensional space. The changes are similar to rotation and stretching in space and it can be described by the following formula .

$$A_{m\times n}P_{n\times n} = \widetilde{A}_{m\times n} \tag{3}$$

By changing a matrix A of a m x n R into a matrix of m x r will make the number of features from n to r ($r < n$), the r is actually a refinement of n features, we called this feature compression, the changes can be described by the following formula:

$$A_{m\times n}P_{n\times r} = \widetilde{A}_{m\times r} \tag{4}$$

But what is the relationship between this and SVD? As we have mentioned before, the singular vectors of SVD also ordered according to the order of their singular value (from big to small).From the point view of PCA, the axis with maximum variance is the first singular vector, the axis with second largest variance is the second singular vector... etc. We recall the following formula of SVD we obtained before:

$$A_{m\times n} \approx U_{m\times r}\Sigma_{r\times r}V^T_{r\times n} \tag{5}$$

We can multiply both sides of the formula by the matrix V, as V is an orthogonal matrix, so the multiplying result of V and its transpose matrix V^T is a unit array I, then we get the formula bellow.

$$A_{m\times n}V_{r\times n} \approx U_{m\times r}\Sigma_{r\times r}V^T_{r\times n}V_{r\times n} \tag{6}$$

$$A_{m\times n}V_{r\times n} \approx U_{m\times r}\Sigma_{r\times r} \tag{7}$$

In contrast to formula 3, we believe that the V is actually P, which is a changing vector. The change from matrix of m x n into matrix of m x r is a compression on column of the matrix. If we want to compress the row of a matrix (from the point view of PCA, the row compression can be deemed as sum similar sample together, or removing the sample that do not have much value) , we also can get a general example of row compression, as the formula8 described:

$$P_{r\times m}A_{m\times n} = \widetilde{A}_{r\times n} \tag{8}$$

Thus we compress a matrix with m rows into a matrix with r rows. As for SVD, we multiply both sides of the formula by the matrix U^T(the transpose matrix of U),then we get the following formula:

$$U^T_{r\times m}A_{m\times n} \approx \Sigma_{r\times r}V^T_{r\times n} \tag{9}$$

Now we get the formula of compression to rows of matrix. It can be seen that PCA actually is a package of SVD, when we realize the SVD, the PCA is also realized. In addition, we can get the two directions of PCA by using SVD, if we decompose the eigenvalue of matrix A^TA , we can only get PCA with one direction.

II. EXPERIMENT

Experiment 1 –application of SVD in data analysis

There is always noise in the data we collected, no matter how sophisticated the equipment is and how good mothed we use, there will always be some errors [14]. As we have mentioned before, bigger singular values corresponds to the principle information of the matrix, so it is reasonable to analyze data with SVD.As an Experiment of SVD, we collect some data described by the following figure.

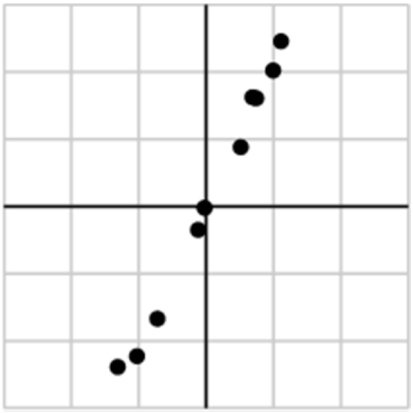

Fig. 5. The original data

We represent these datum in the form of a matrix as bellow:

-1.03	0.74	-0.02	0.51	-1.31	0.99	0.69	-0.12	-0.72	1.11
-2.23	1.61	-0.02	0.88	-2.39	2.02	1.62	-0.35	-1.67	2.46

After singular value decomposition, we get σ1=6.04 and σ2=0.22

Since the first singular value is much larger than the second one, there are some noises in the datum, and the corresponding

values of the second singular values can be ignored. After the decomposition of SVD, the main sample points are retained as shown in the fig.6

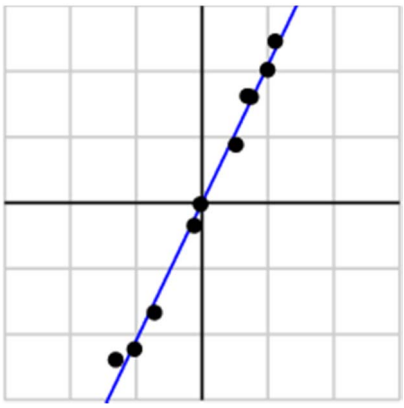

Fig. 6. The remaining data after of SVD

By observing the main sample datum that retained, we believe that the process has some connection with PCA. PCA also uses the SVD to detect dependencies between datum and redundant information [15].

Experiment 2 –application of SVD in Latent Semantic Indexing (LSI)

Let's look at table1 in bellow:

TABLE 1. THE DATA COLLETED

Index Words	Titles					
	T1	T2	T3	T4	T5	T6
book			1	1		
dads						1
dummies		1				
estate						
guide	1					1
investing	1	1	1	1	1	1
market	1		1			
real						
rich						2
stock	1		1			
value				1	1	

This is a matrix, but in particular, the value of this matrix row means in what title a word appear (one row is one-dimensional feature said before) and the value on a column means what words appear in a title. For example, there are four words --guide, investing, market and stock appear in T1, each word appeared once. If we use SVD on this matrix, we can get the following matrix, as shown in fig.7:

book	0.15	-0.27	0.04
dads	0.24	0.38	-0.09
dummies	0.13	-0.17	0.07
estate	0.18	0.19	0.45
guide	0.22	0.09	-0.46
investing	0.74	-0.21	0.21
market	0.18	-0.30	-0.28
real	0.18	0.19	0.45
rich	0.36	0.59	-0.34
stock	0.25	-0.42	-0.28
value	0.12	-0.14	0.23

×

3.91	0	0
0	2.61	0
0	0	2.00

×

T1	T2	T3	T4	T5	T6
0.35	0.22	0.34	0.26	0.22	0.49
-0.32	-0.15	-0.46	-0.24	-0.14	0.55
-0.41	0.14	-0.16	0.25	0.22	-0.51

Fig. 7. The multiplication of matrix according to SVD

Left singular vectors represent some characteristics of the word vector, right singular vectors represent some characteristics of the document, the singular value matrix in middle represent some important relation between one row of left singular vectors and one column of right singular vectors, it's more important if the number is larger.

From this matrix we can also find some interesting things, first of all, the first column of left singular vector represent the frequency of each word appears, although it is not linear, it can be thought of as a general description, such as the value of book is 0.15, means it appear 2 times in corresponding documents, investing is 0.74 means 6 times, rich is 0.36, means 2 times,., etc.

Secondly, the first row in the right singular vector represents the approximation of the number of words in each document. For example, T6 is 0.49, means there are 5 words in T6; T2 is 0.22, means there are 2 words in the T2.

When we look the matrix inversely, we can take the last 2 dimensions of the left singular vector and the right singular vector, and projected them onto a plane, you can get the picture bellow:

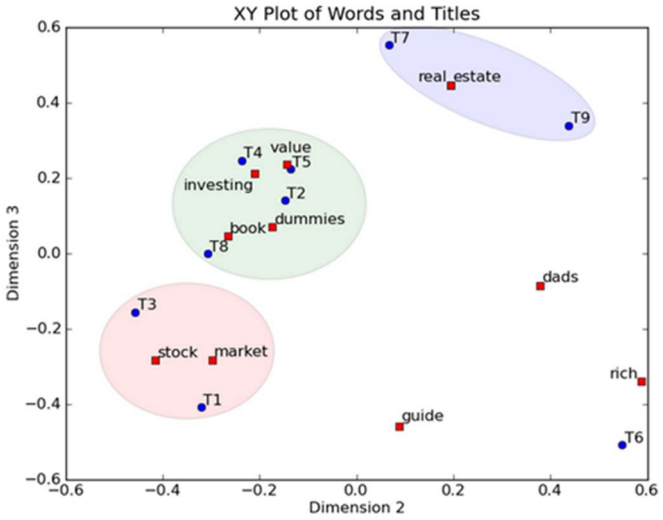

Fig. 8. The result of LSI after SVD

In the diagram, each red dot represents a word, each blue point represents a document, so that we can cluster these words and documents, such as 'stock' and 'market' can be placed in a class, because they always appear together, 'real' and 'estate' can be placed in a class, while the words 'dads' and 'guide'

looks a bit isolated, and we do not merge them. According to the effect of such clustering, we can extract synonyms from the document, so that when users retrieve the document, they can get the service of semantic level (synonym set) to retrieve, but not in the word level before. This can reduce the time of search and the size of storage, because the effect of such compressed document is the same with PCA and SVD. At the same time, we can improve the users' experience, when users enter a word, we can find it in the synonyms set, but we cannot do this in the traditional index.

III. CONCLUSION

PCA is one of the most commonly used methods in data dimension reduction, and the essence of PCA is to select the direction which contains the most information. The choice of the projection direction is based on the maximum variance of the data distribution. SVD can obtain the other direction of the principal component, while PCA can only get the main components of a single direction, SVD can get the same results with PCA, but SVD is usually better than the direct use of PCA, it is more stable, we can say that PCA is a package of SVD. Surely, both PCA and SVD have certain limitations, they can only be used for processing the linear relationship, as for more complex relationships among datum, we may need to use nonlinear methods, such as the introduction of kernel function, then KPCA may be a good choice. In addition, with the advent of the era of big data, methods of dimension reduction are facing greater challenges, perhaps we need to deeply study distributed method (such as the distributed SVD) to solve the problem of dimension reduction for big data in the future.

REFERENCES

[1] J. Y. Kim and Y. S. Kim, "Face Tracking and Recognition in Video with PCA-based Pose-Classification and (2D) 2 PCA recognition algorithm," Journal of Korean Institute of Intelligent Systems, vol. 23, April2012, pp. 423-430.

[2] Z. Dao qiang, Z. Zhi-Hua, "Two-directional two-dimensional PCA for efficient face representation and recognition," Neuro Computing, vol.2,May.1999,pp. 224-231.

[3] Q. Zhao, B. Liang, and F. Duan, "Combination of Improved PCA and LDA for Video-based Face Recognition," Journal of Computational Information Systems,vol.3,May.2010, pp. 273-280.

[4] E. Candes and X. Li. "Robust Principal Component analysis," ´ Journal of Software, vol.4,July.2012,pp221-224.

[5] T. Celik. "Unsupervised change detection in satellite images using principal component analysis and K-means clustering," IEEE Geoscience and Remote Sensing Letters,vol.4,April.2011, pp.121-125.

[6] F .Cong, J.Chen,G. Dong, "Short time matrix series based singular value decomposition for rolling bearing fault diagnosis," Mechanical Systems and Signal Processing, vol.5, June.2010,pp. 218-230.

[7] G. M. Jackson, I M Mason and S A Green Halgh. Principal component transforms of recordings by singular value decomposition. Geophysics, vol.4,June.1991, pp.528-533.

[8] H.Y.Shen,Q.C.Li " Seismic wave fields separation and noise attenuation in frequency domain singular value decom position" Near-Surface Geophysics and Human Activity,vol.2,May.2013, pp.178～ 181.

[9] L.M.Sergio,L.M.Freire, J.Ulrych, " Application of singular value decomposition to vertical seismic profiling", Geophysics, vol.2,May.2011,pp.778-785.

[10] H.Y.Shen, Q.C.Li, "Wave field separation and noise attenuation in frequency domain via single value decomposition(SVD)",Near-Surface Geophysics and Human Actitivy,vol.4,July.2003,pp.178-181.

[11] S.W. Choi, J.H. Park, and I.B. Lee. "Process monitoring using a Gaussian mixture model via principal component analysis and discriminant analysis," Computer and Chemical Engineering,vol.4,April.1999, pp.1377-1387.

[12] P.S. Dhillon, Y. Lu, D. Foster, and L. Ungar. "New subsampling algorithms for fast least squares regression," Advances in Neural Information Processing, vol.2, March.1992,pp.360-368.

[13] Y. Gao, and E. al, "Performance and Power Analysis of High-Density Multi-GPGPU Architectures: A Preliminary Case Study", Proceedings of 17th IEEE International Conference on High Performance Computing and Communications (HPCC-ICESS-CSS 2015), pp. 66-71. IEEE, 2015

[14] W.Q. Meeker and Y. Hong. "Reliability meets big data: opportunities and challenges," Qualify Engineering,vol.2,June.1999, pp.102-116.

[15] H. Shen and J.Z. Huang, "Sparse principal component analysis via regularized low rank matrix approximation," Journal of Multivariate Analysis,vol.5,March.2013 pp. 1015-1034..