e-PGPathshala

Subject : Computer Science

Paper: Machine Learning

Module: Dimensionality Reduction - II

Module No: CS/ML/28

Quadrant I – e-text

Welcome to the e-PG Pathshala Lecture Series on Machine Learning. In the previous module we discussed basics of dimensionality reduction and one technique of dimensionality reduction namely Principal Component Analysis. In this module two other approaches to dimensionality reduction namely Fisher Linear Discriminant and Singular Value Decomposition will be discussed.

Learning Objectives:

The learning objectives of this module are as follows:

- To explain the Fisher Linear Discriminant approach
- To understand the concept of Singular Value Decomposition
- To outline the computation of Singular Value Decomposition

28.1 Introduction

The main idea of Fisher linear discriminant approach is finding the projection to a line such that samples from different classes projected on the line are well separated (Figure 28.1).

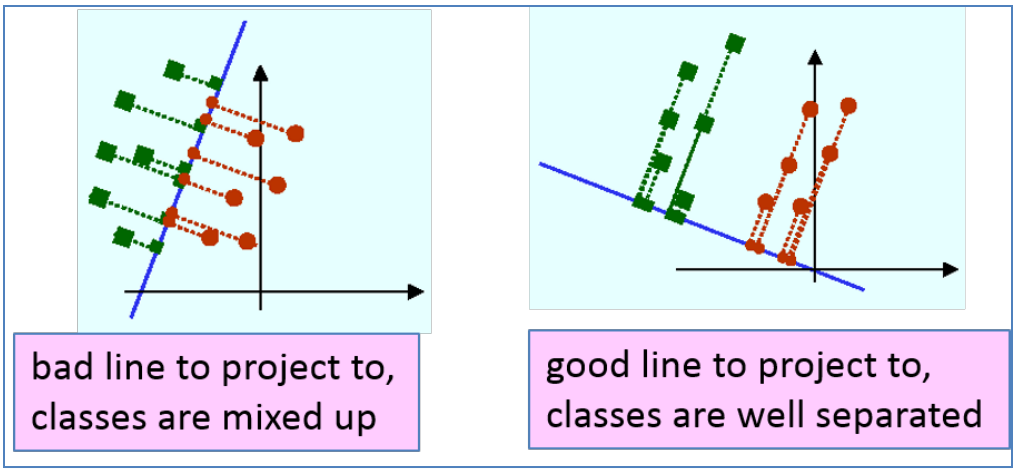

bad line to project to, classes are mixed up
good line to project to, classes are well separated

Figure 28.1 Bad and Good Projections

28.2 The Basis of Fisher Discriminant

Suppose we have 2 classes and d-dimensional samples X_1,...., X_n where n1 samples come from the first class and n2 samples come from the second class. Consider projection on a line, and let the line direction be given by unit vector V. Scalar $V^t X_i$ is the distance of projection of x$_i$ from the origin. Thus $V^t X_i$ is the projection of X_i into a one dimensional subspace (Figure 28.2).

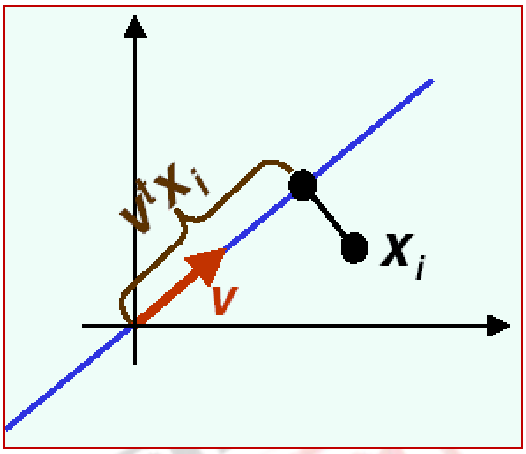


Figure 28.2 $V^t X_i$ – the Projection of X_i

Thus the projection of sample X_i onto a line in direction V is given by $V^t X_i$. How do we measure separation between projections of different classes?. Let $\widetilde{\mu_1}$ and $\widetilde{\mu_2}$ be the means of projections of classes 1 and 2 (Figure 28.3) and let μ_1 and μ_2 be the means of classes 1 and 2 and | $\widetilde{\mu_1}$ - $\widetilde{\mu_2}$ | seems like a good measure.

$$\tilde{\mu}_1 = \frac{1}{n_1} \sum_{x_i \in C1}^{n_1} v^t x_i = v^t \left(\frac{1}{n_1} \sum_{x_i \in C1}^{n_1} x_i \right) = v^t \mu_1$$

$$\textit{similarly,} \qquad \tilde{\mu}_2 = v^t \mu_2$$

Figure 28.3 Means of Projections

Now let us discuss the goodness of |$\widetilde{\mu_1}$ - $\widetilde{\mu_2}$| as a measure of separation. The larger |$\widetilde{\mu_1}$ - $\widetilde{\mu_2}$| , the better is the expected separation. The vertical axes is a better line than the horizontal axes to project to for class separability (Figure 28.4). However $\widehat{\mu_1}$ - $\widetilde{\mu_1}$| >|$\widetilde{\mu_1}$ - $\widetilde{\mu_2}$|.

The problem with |$\widetilde{\mu_1}$ - $\widetilde{\mu_2}$| is that it does not consider the variance of the classes. We need to normalize |$\widetilde{\mu_1}$ - $\widetilde{\mu_2}$| by a factor which is proportional to variance (Figure 28.5).

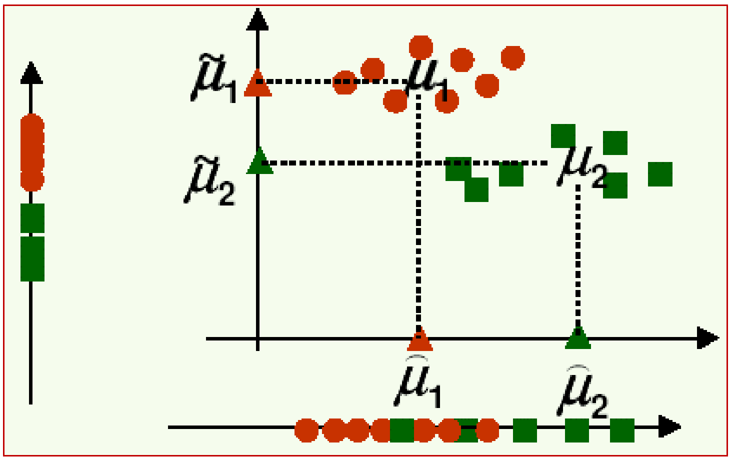


Figure 28.4 The |$\widetilde{\mu_1}$ - $\widetilde{\mu_2}$| Measure of Separation

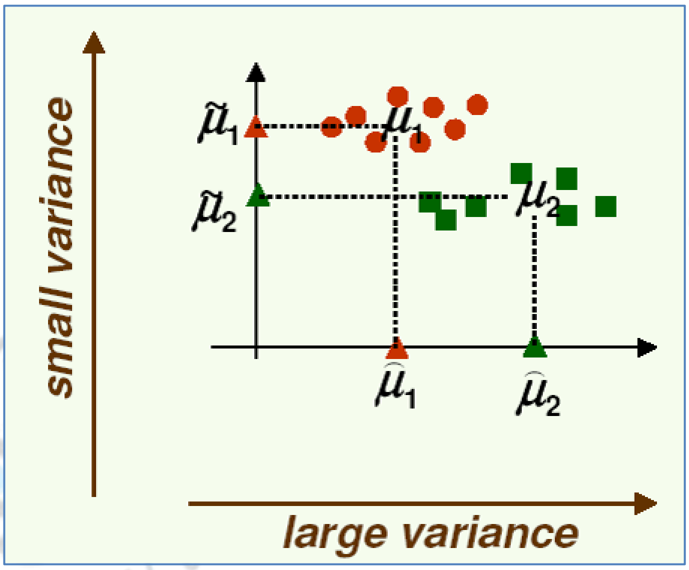


Figure 28.5 Normalized Difference in Means

Let us consider some samples $Z_1,\ldots, Z_n$. Sample mean is as given below

$$\mu_z = \frac{1}{n}\sum_{i=1}^{n} z_i$$

Now let us define their **scatter** as

$$s = \sum_{i=1}^{n} (z_i - \mu_z)^2$$

Thus scatter is just sample variance multiplied by n. In other words, scatter measures the same concept as variance, the spread of data around the mean, only that scatter is just on a different scale than variance.

28.3 Fisher Linear Discriminant

Fisher Solution to finding the projection to a line such that samples from different classes projected on the line are well separated is to normalize $|\widetilde{\mu_1} - \widetilde{\mu_2}|$ by scatter.

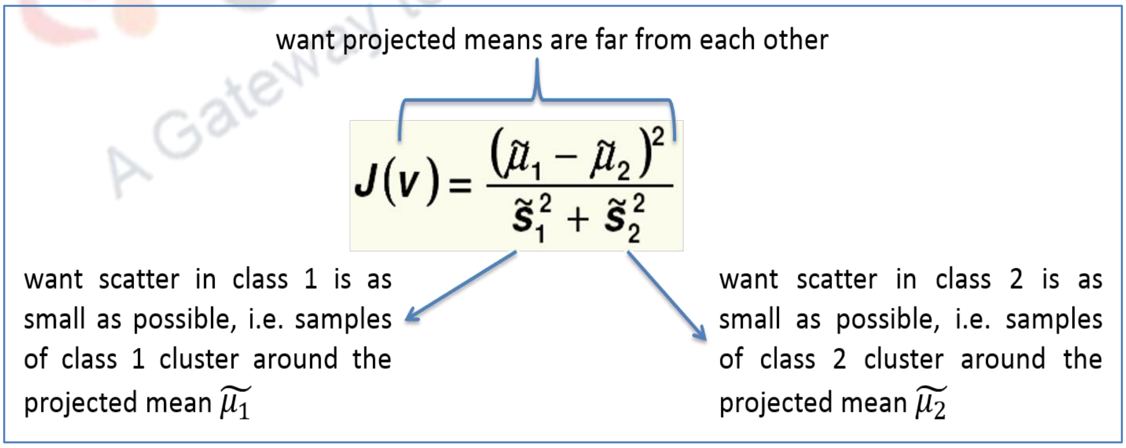
Let $\boldsymbol{y_i} = \boldsymbol{V^t X_i}$, i.e. $\boldsymbol{y_i}$'s are the projected samples and the scatter for projected samples of class 1 is as given below:

$$\tilde{s}_1^2 = \sum_{y_i \in Class\ 1} (y_i - \tilde{\mu}_1)^2$$

Similarly the scatter for projected samples of class 2 is

$$\tilde{s}_2^2 = \sum_{y_i \in Class\ 2} (y_i - \tilde{\mu}_2)^2$$

We need to normalize by both scatter of class 1 and scatter of class 2. Thus Fisher linear discriminant needs to project on line in the direction v which maximizes J(v) (Figure 28.5). Here J(v) is defined such that we want the projected means to be far from each other and the scatter of each class to be as small as possible that is we want the samples of the respective classes to cluster around the projected means. If we find v which makes J(v) large, we are guaranteed that the classes are well separated (Figure 28.6).



want projected means are far from each other

$$J(v) = \frac{(\tilde{\mu}_1 - \tilde{\mu}_2)^2}{\tilde{s}_1^2 + \tilde{s}_2^2}$$

want scatter in class 1 is as small as possible, i.e. samples of class 1 cluster around the projected mean $\widetilde{\mu_1}$

want scatter in class 2 is as small as possible, i.e. samples of class 2 cluster around the projected mean $\widetilde{\mu_2}$

Figure 28.5 Definition of J(V)

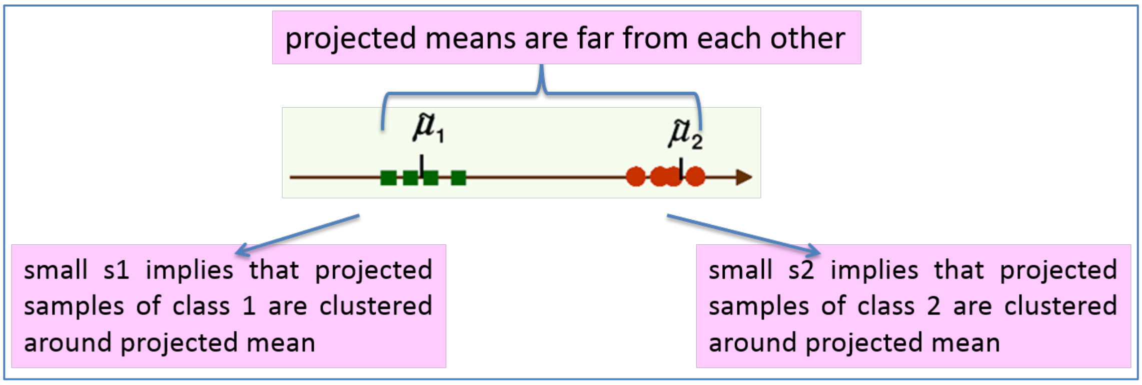

projected means are far from each other
$\tilde{\mu}_1$
$\tilde{\mu}_2$
small s1 implies that projected samples of class 1 are clustered around projected mean
small s2 implies that projected samples of class 2 are clustered around projected mean

Figure 28.6 Well Separated Projected Samples

All we need to do now is to express J explicitly as a function of v and maximize it. This is fairly straightforward but needs application of linear algebra and calculus. We define the separate class scatter matrices S1 and S2 for classes 1 and 2. These measure the scatter of original samples x_i (before projection) as follows:

$$S_1 = \sum_{x_i \in Class\ 1} (x_i - \mu_1)(x_i - \mu_1)^t$$

$$S_2 = \sum_{x_i \in Class\ 2} (x_i - \mu_2)(x_i - \mu_2)^t$$

Now let us consider the samples belonging to two classes as given below:

- Class 1 has 5 samples c1=[(1,2),(2,3),(3,3),(4,5),(5,5)]
- Class 2 has 6 samples c2=[(1,0),(2,1),(3,1),(5,3),(6,5)]

Now let us arrange data in 2 separate matrices as follows:

$$c_1 = \begin{bmatrix} 1 & 2 \\ \vdots & \vdots \\ 5 & 5 \end{bmatrix} \qquad c_2 = \begin{bmatrix} 1 & 0 \\ \vdots & \vdots \\ 6 & 5 \end{bmatrix}$$

It is to be noted that PCA performs very poorly on this data because the direction of largest variance is not helpful for classification (Figure 28.7).

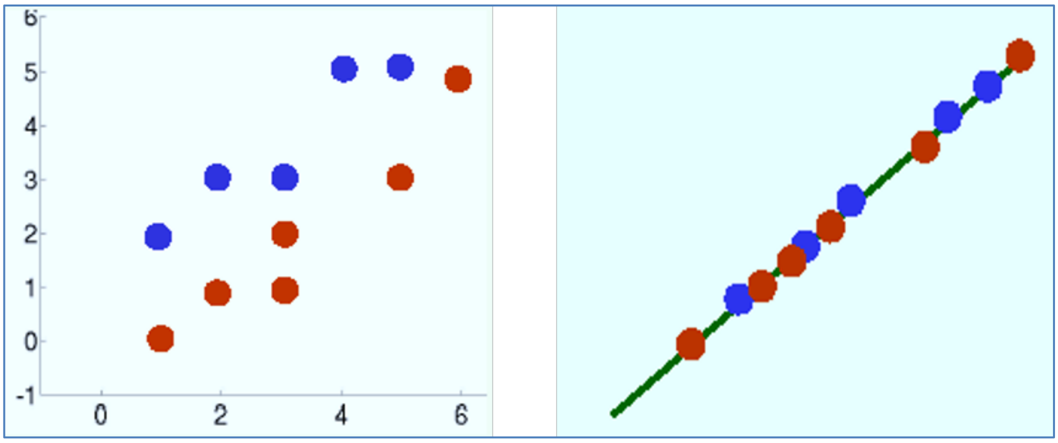


Figure 28.7 PCA based Dimensionality Reduction

Now let us first compute the mean for each class.

M1= mean(c1)=[3 3.6] M2= mean (c2) = [3.3 2]

Now based on these means let us compute the scatter matrices S1 and S2 for each class as follows:

$$S_1 = 4 * cov(c_1) = \begin{bmatrix} 10 & 8.0 \\ 8.0 & 7.2 \end{bmatrix} \qquad S_2 = 5 * cov(c_2) = \begin{bmatrix} 17.3 & 16 \\ 16 & 16 \end{bmatrix}$$

Now the within the class scatter is a follows:

$$S_W = S_1 + S_2 = \begin{bmatrix} 27.3 & 24 \\ 24 & 23.2 \end{bmatrix}$$

This matrix has full rank, and we do not have to solve for Eigen values

The inverse of S_W

$$S_W \text{ is } S_W^{-1} = inv(S_W) = \begin{bmatrix} 0.39 & -0.41 \\ -0.41 & 0.47 \end{bmatrix}$$

Finally, the optimal line direction **v** is as given below:

$$v = S_W^{-1}(\mu_1 - \mu_2) = \begin{bmatrix} -0.79 \\ 0.89 \end{bmatrix}$$

Notice, that as long as the line has the right direction, its exact position does not matter.

Finally the last step is to compute the actual 1D vector y. Let's do it separately for each class

$$Y_1 = v^t c_1^t = \begin{bmatrix} -0.65 & 0.73 \end{bmatrix} \begin{bmatrix} 1 & \cdots & 5 \\ 2 & \cdots & 5 \end{bmatrix} = \begin{bmatrix} 0.81 & \cdots & 0.4 \end{bmatrix}$$

$$Y_2 = v^t c_2^t = \begin{bmatrix} -0.65 & 0.73 \end{bmatrix} \begin{bmatrix} 1 & \cdots & 6 \\ 0 & \cdots & 5 \end{bmatrix} = \begin{bmatrix} -0.65 & \cdots & -0.25 \end{bmatrix}$$

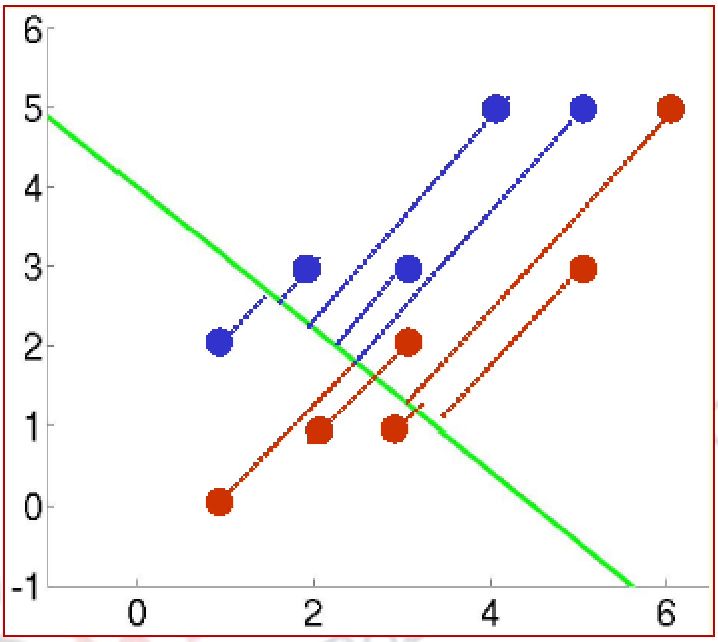


Figure 28.8 Projection based on Fisher Linear Discriminant

28.4 Singular Value Decomposition

Now let us discuss the final method of dimensionality reduction namely Singular Value Decomposition (SVD). SVD can be viewed as a method for transforming correlated variables into a set of uncorrelated ones that better expose the various relationships among the original data items. It is a method for identifying and ordering the dimensions along which data points exhibit the most variation. With SVD, it's possible to find the best approximation of the original data points using fewer dimensions. Hence, SVD is used for data reduction.

Singular Value Decomposition factorizes a real or complex matrix. For an $M \times N$ matrix **A** of rank r there exists a factorization (Singular Value Decomposition = **SVD**) as follows:

$$A = \underset{\boxed{M\times M}}{U}\;\underset{\boxed{M\times N}}{\Sigma}\;\underset{\boxed{V \text{ is } N\times N}}{V^T}$$

Here the columns of ***U*** are orthogonal eigen vectors of ***AA***^{T}, the columns of ***V*** are orthogonal eigen vectors of ***A***^{T}***A*** and Eigen values $\lambda_1 \ldots \lambda_r$ of ***AA***^{T} are the eigen values of ***A***^{T}***A***. An illustration of SVD dimensions and sparseness is as given in Figure 28.9.

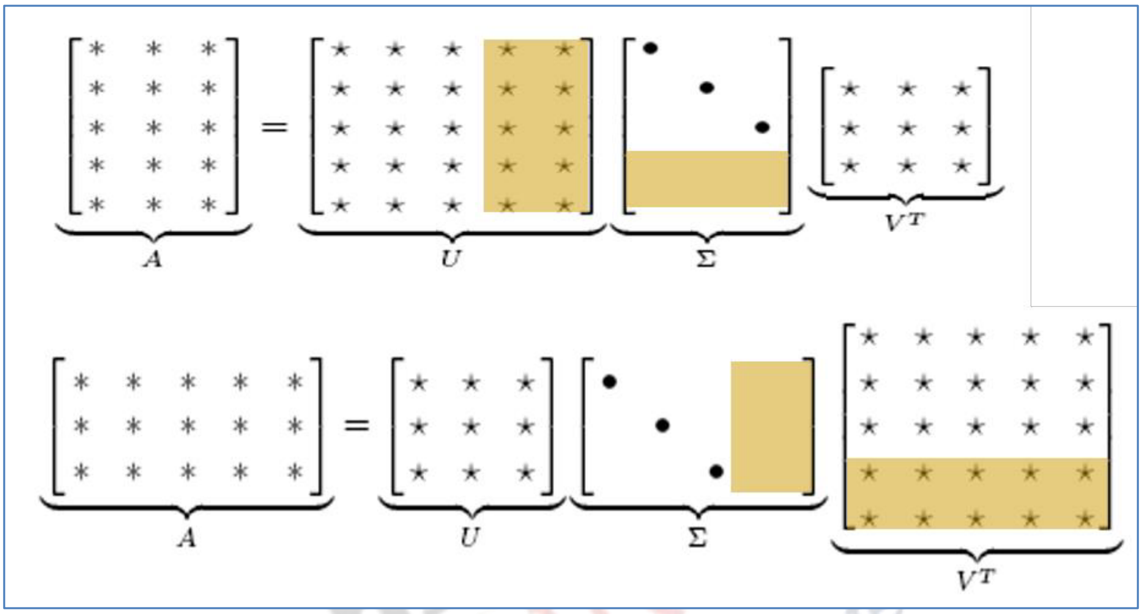


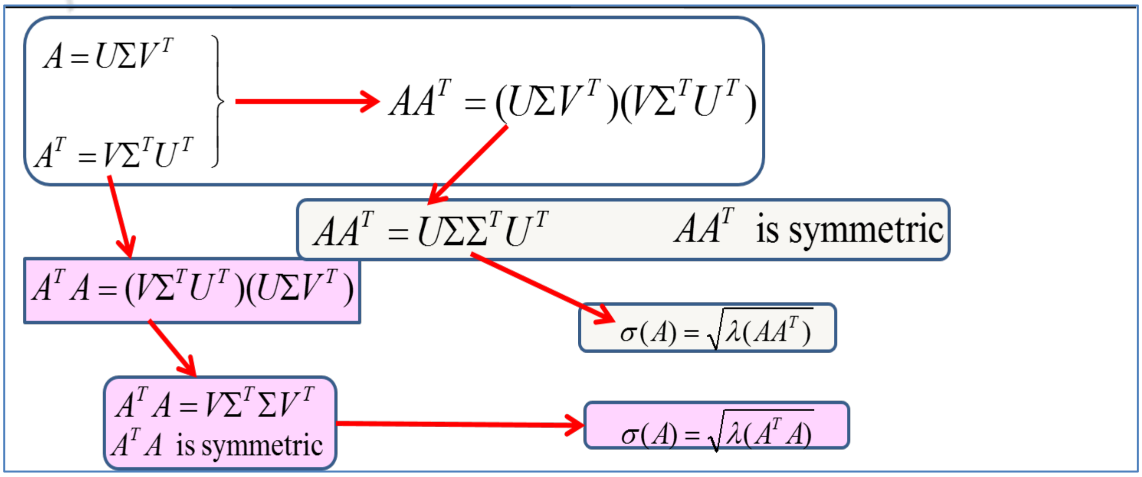
Figure 28.9 SVD Illustration

The corresponding singular values are given below.

$$\sigma_i = \sqrt{\lambda_i}$$

$$\Sigma = diag(\sigma_1 \ldots \sigma_r) \longleftarrow \boxed{\textit{Singular values.}}$$

28.5 Computation of SVD



$A = U\Sigma V^T$

$A^T = V\Sigma^T U^T$

$AA^T = (U\Sigma V^T)(V\Sigma^T U^T)$

$AA^T = U\Sigma\Sigma^T U^T$ AA^T is symmetric

$A^T A = (V\Sigma^T U^T)(U\Sigma V^T)$

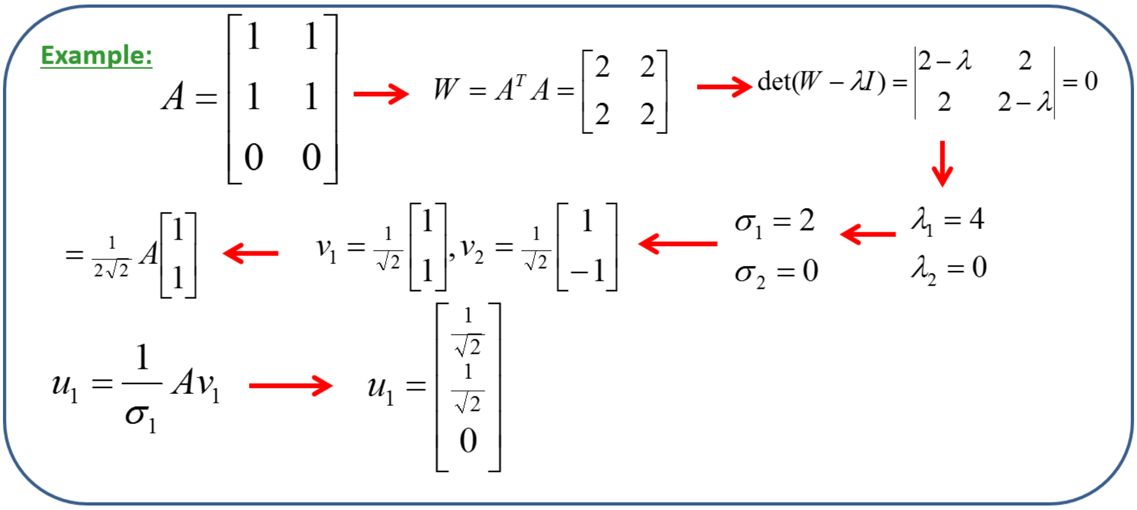
$\sigma(A) = \sqrt{\lambda(AA^T)}$

$A^T A = V\Sigma^T\Sigma V^T$

$A^T A$ is symmetric

$\sigma(A) = \sqrt{\lambda(A^T A)}$

(a)



Example:

$$A = \begin{bmatrix}1&1\\1&1\\0&0\end{bmatrix} \rightarrow W = A^T A = \begin{bmatrix}2&2\\2&2\end{bmatrix} \rightarrow \det(W-\lambda I) = \begin{vmatrix}2-\lambda & 2\\ 2 & 2-\lambda\end{vmatrix} = 0$$

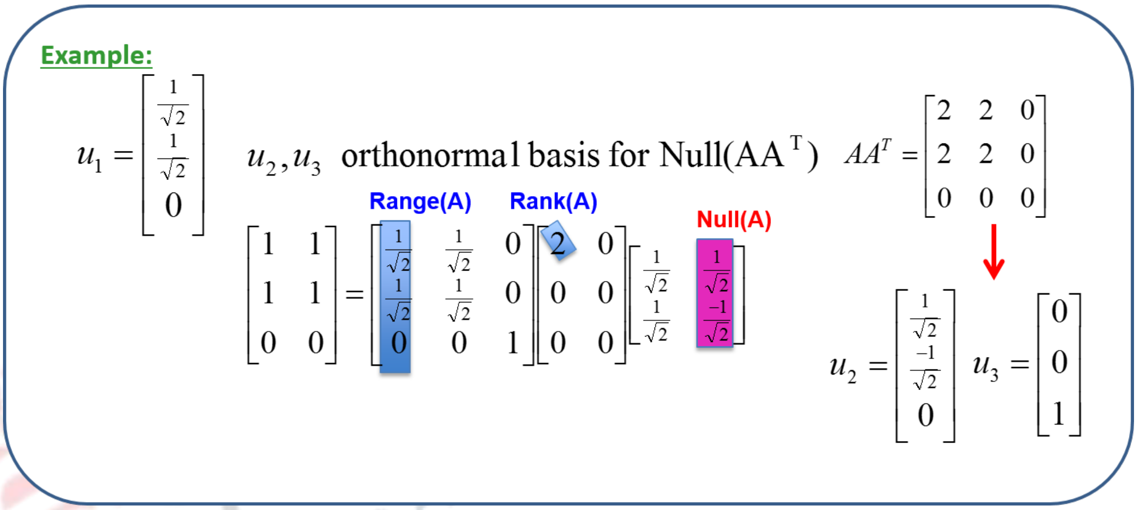
$$\downarrow$$

$$\begin{matrix}\lambda_1 = 4\\ \lambda_2 = 0\end{matrix} \leftarrow \begin{matrix}\sigma_1 = 2\\ \sigma_2 = 0\end{matrix}$$

$$= \frac{1}{2\sqrt{2}} A\begin{bmatrix}1\\1\end{bmatrix} \leftarrow v_1 = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix}, v_2 = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\-1\end{bmatrix} \leftarrow$$

$$u_1 = \frac{1}{\sigma_1}Av_1 \rightarrow u_1 = \begin{bmatrix}\frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}}\\ 0\end{bmatrix}$$

(b)



Example:

$$u_1 = \begin{bmatrix}\frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}}\\ 0\end{bmatrix} \qquad u_2, u_3 \text{ orthonormal basis for Null}(AA^T) \qquad AA^T = \begin{bmatrix}2&2&0\\2&2&0\\0&0&0\end{bmatrix}$$

$$\begin{bmatrix}1&1\\1&1\\0&0\end{bmatrix} = \begin{bmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0\\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}2&0\\0&0\\0&0\end{bmatrix}\begin{bmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}}\end{bmatrix}$$

$$\downarrow$$

$$u_2 = \begin{bmatrix}\frac{1}{\sqrt{2}}\\ \frac{-1}{\sqrt{2}}\\ 0\end{bmatrix} \quad u_3 = \begin{bmatrix}0\\0\\1\end{bmatrix}$$

(c)

Figure 28.10 Computation of SVD

Any real ***m* x *n*** matrix *A* can be decomposed uniquely:

$$A = UDV^T$$

Here U is ***m* x *n*** and column orthonormal ($U^TU=I$) and *D* is ***n* x *n*** and diagonal

$$D = diag(\sigma_1, \sigma_2, \ldots, \sigma_n)$$

σ_i are called *singular* values of A. It is assumed that $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n \geq 0$

V is ***n* x *n*** and orthonormal ($VV^T=V^TV=I$)

The columns of U are eigenvectors of AA^T

$$AA^T = UDV^T VDU^T = UD^2U^T$$

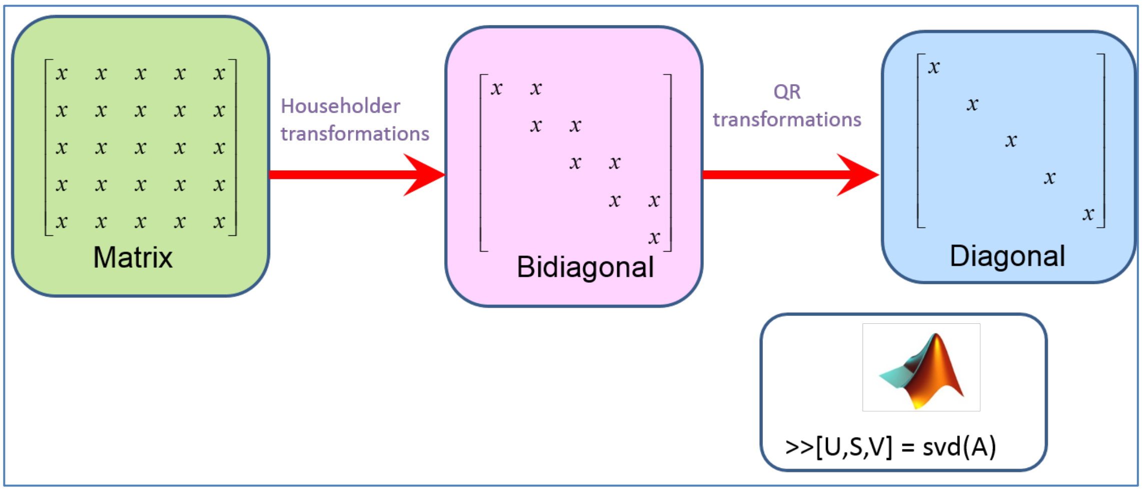
The columns of V are eigenvectors of A^TA

$$A^T A = VDU^T UDV^T = VD^2V^T$$

If λ_i is an eigenvalue of A^TA (or AA^T), then $\lambda_i = \sigma_i^2$

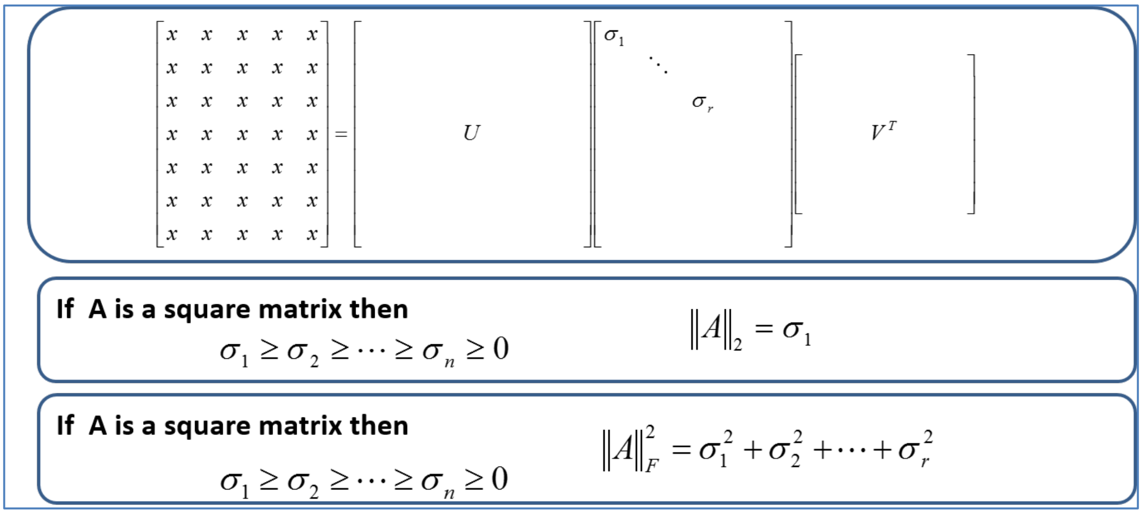
The steps in the computation of SVD is given in Figure 28.10. From Figure 28,10 (b) we see that given the matrix A, we first find W = A^TA and then make the determinant of (W-λI)=0 where I is the identity matrix to find the value of the eigen values λ_1 and λ_2 and hence the singular values σ_1 and σ_2. These eigenvectors become column vectors in a matrix ordered by the size of the corresponding eigenvalue. In other words, the eigenvector of the largest eigenvalue is column one, the eigenvector of the next largest eigenvalue is column two, and so forth and so on until we have the eigenvector of the smallest eigenvalue as the last column of our matrix. Finally after appropriate calculations shown in the figure we find the orthonormal basis u_1. Similarly we now we use AA^T to find u_1, u_2 and u_3, the orthonormal basis. Now we write the matrix A= range matrix formed by u_1 , u_2 and u_3 x rank matrix of A and the Null (A). Figure 28.11 shows how the matrix needs to be converted into bi-diagonal and then diagonal to get the SVD.

The properties of the standard deviation of a square matrix is shown in Figure 28.12. If ***m=n,*** then *U* is ***n* x *n*** and orthonormal ($U^TU=UU^T=I$), *D* is ***n* x *n*** and diagonal is as before and *V* is ***n* x *n*** and orthonormal ($VV^T=V^TV=I$). Similarly the properties of eigen values, singular values and determinants of square matrices is shown in Figure 28.13. Another example of the calculation of singular values is given in Figure 28.14.



>>[U,S,V] = svd(A)

Figure 28.11 Concept used in SVD calculation



$$\begin{bmatrix} x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \end{bmatrix} = \begin{bmatrix} \\ \\ U \\ \\ \\ \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \\ & & \\ & & \end{bmatrix} \begin{bmatrix} \\ V^T \\ \\ \end{bmatrix}$$

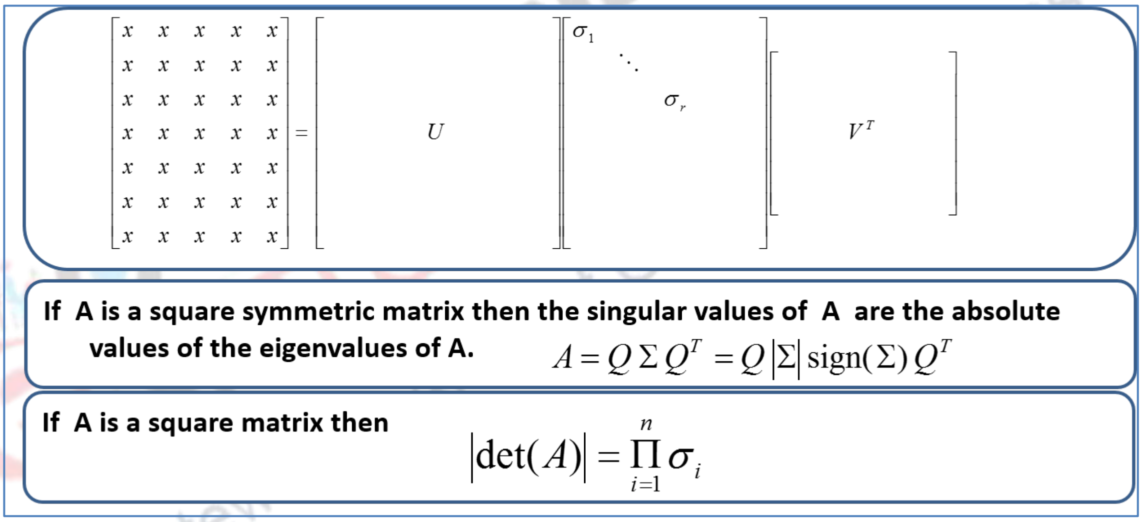
If A is a square matrix then

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0 \qquad \|A\|_2 = \sigma_1$$

If A is a square matrix then

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0 \qquad \|A\|_F^2 = \sigma_1^2 + \sigma_2^2 + \cdots + \sigma_r^2$$

Figure 28.12 Standard Deviation of a Square Matrix



$$\begin{bmatrix} x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \end{bmatrix} = \begin{bmatrix} \\ \\ U \\ \\ \\ \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \\ & & \\ & & \end{bmatrix} \begin{bmatrix} \\ V^T \\ \\ \end{bmatrix}$$

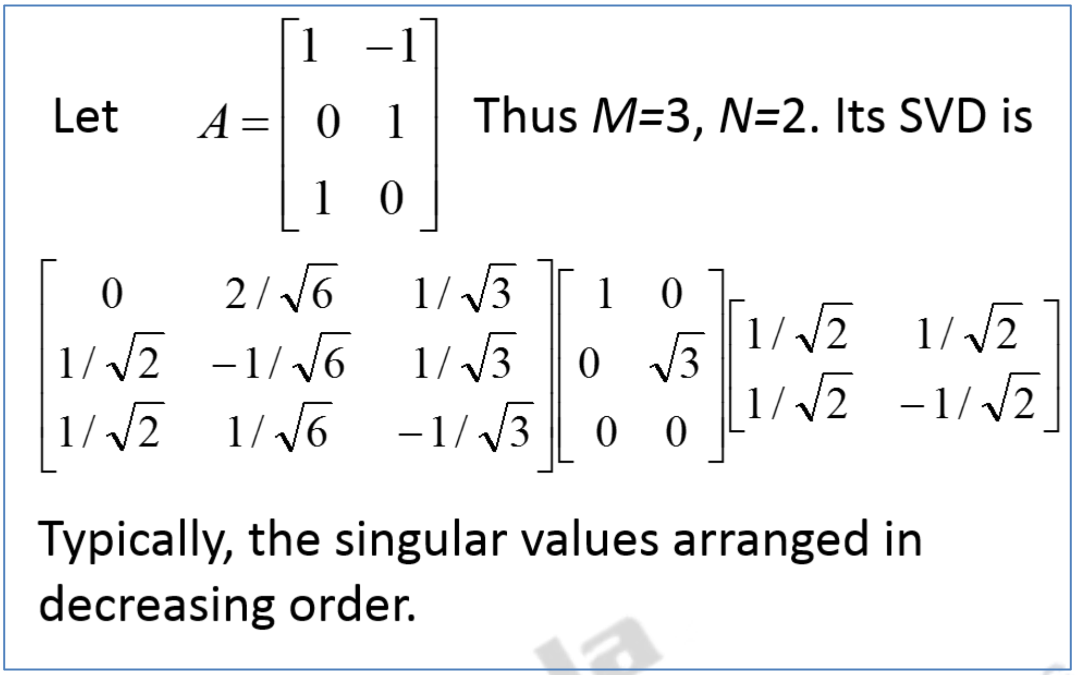
If A is a square symmetric matrix then the singular values of A are the absolute values of the eigenvalues of A.

$$A = Q\,\Sigma\,Q^T = Q\,|\Sigma|\,\mathrm{sign}(\Sigma)\,Q^T$$

If A is a square matrix then

$$|\det(A)| = \prod_{i=1}^{n} \sigma_i$$

Figure 28.13 Singular Values of a Square Symmetric Matrix



Let $A = \begin{bmatrix} 1 & -1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$ Thus *M*=3, *N*=2. Its SVD is

$$\begin{bmatrix} 0 & 2/\sqrt{6} & 1/\sqrt{3} \\ 1/\sqrt{2} & -1/\sqrt{6} & 1/\sqrt{3} \\ 1/\sqrt{2} & 1/\sqrt{6} & -1/\sqrt{3} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{3} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}$$

Typically, the singular values arranged in decreasing order.

Figure 28.14 Example for Calculation of Singular Values

Figure 28.15 gives a table showing the difference between SVD and eigen decomposition.

SVD	**Eigen Decomp**
Use two Different bases U & V	Uses just one (eigenvectors)
Uses orthonormal bases	Generally is not orthogonal
All matrices (even rectangular)	Not all matrices (even square) (only diagonalizable)

Figure 28.15 Difference between SVD and Eigen Decomposition

Summary

- Explained the Fisher Linear Discriminant approach
- Outlined the concept of Singular Value Decomposition
- Discussed the computation of Singular Value Decomposition