

# CS7.403 - SMAI: Assignment 2

Avneesh Mishra  
avneesh.mishra@research.iiit.ac.in

November 8, 2021

## Contents

<b>1</b>	<b>Q1: EVD and SVD</b>	<b>2</b>
1.1	A: Generalized to Matrices . . . . .	2
1.2	B: Find SVD of a matrix . . . . .	2
<b>2</b>	<b>Q2: LDA and PCA</b>	<b>5</b>
2.1	A: Points about PCA . . . . .	5
2.2	B: True / False . . . . .	5

# 1 Eigen Values and Eigen Vectors

## Eigen Value Decomposition

A matrix  $\mathbf{A}$  can be decomposed into

$$\mathbf{A} = \mathbf{Q} \mathbf{\Lambda} \mathbf{Q}^{-1} \quad (1)$$

Where the matrix  $\mathbf{Q}$  is formed by horizontally stacking eigenvectors of  $\mathbf{A}$  as columns and the matrix  $\mathbf{\Lambda} = \text{diag}(\lambda)$  where  $\lambda$  is the vector containing the corresponding eigen-values. The relation between  $\mathbf{A}$ , an eigenvector and corresponding eigen-value is given by

$$\mathbf{A} \mathbf{v}_i = \lambda_i \mathbf{v}_i$$

## Singular Value Decomposition

A matrix  $\mathbf{M}$  can be decomposed as

$$\mathbf{M} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^* \quad (2)$$

Where  $\mathbf{\Sigma}$  is a diagonal matrix consisting of singular values (usually in descending order). The columns of  $\mathbf{U}$  are formed by left-singular vectors of  $\mathbf{M}$ , which are the eigenvectors of  $\mathbf{M} \mathbf{M}^\top$ . The columns of  $\mathbf{V}$  are formed by right-singular vectors of  $\mathbf{M}$ , which are the eigenvectors of  $\mathbf{M}^\top \mathbf{M}$ . The matrix  $\mathbf{V}^*$  is the conjugate transpose of  $\mathbf{V}$ .

If  $\mathbf{M}$  is real,  $\mathbf{U}$  and  $\mathbf{V}$  can be guaranteed to be orthogonal matrices and the decomposition can be written as  $\mathbf{U} \mathbf{\Sigma} \mathbf{V}^\top$ .

### 1.1 A: Generalized to Matrices

Eigenvector decomposition, given by equation 1 can only be done for **square** matrices.

Singular Value Decomposition on the other hand, given by equation 2, can be done for matrices that are not square also.

Hence, *Singular Value Decomposition is more generalizable* to matrices as it can be applied to matrices of any shape (square or not square).

### 1.2 B: Find SVD of a matrix

Usually, numerical approaches are used to calculate the SVD. This is typically a two-step procedure.

In the first step, the matrix is reduced to a bidiagonal matrix (where the elements in the diagonal and either the diagonal above or the diagonal below are non-zero). This is done because calculating SVD of a bidiagonal matrix is faster.

In the second step, SVD of the resultant bidiagonal matrix is calculated. Here, the left and right eigenvectors can be calculated. This is done using a bounded iterative algorithm like QR Algorithm.

The above is a complex numeric procedure, usually abstracted and available on many platforms. When computing SVD of a given matrix, intuition based methods can be directly used (but they do not generalize or scale well).

#### SVD of $M$

The given matrix is

$$M = \begin{bmatrix} 4 & 8 \\ 11 & 7 \\ 14 & -2 \end{bmatrix}$$

We first calculate the left and right-singular vectors of  $M$ , then get the singular values. The left and right matrices are given by

$$M_L = M M^\top = \begin{bmatrix} 80 & 100 & 40 \\ 100 & 170 & 140 \\ 40 & 140 & 200 \end{bmatrix} \quad M_R = M^\top M = \begin{bmatrix} 333 & 81 \\ 81 & 117 \end{bmatrix} \quad (3)$$

Calculating the eigen-vectors of  $M_L$ . First calculate the eigen-values using  $\det(M_L - \lambda I) = 0$  equation

$$\begin{aligned}\det(M_L - \lambda I) = 0 &\Rightarrow \det \left( \begin{bmatrix} 80 & 100 & 40 \\ 100 & 170 & 140 \\ 40 & 140 & 200 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) = 0 \\ &\Rightarrow \det \left( \begin{bmatrix} 80 - \lambda & 100 & 40 \\ 100 & 170 - \lambda & 140 \\ 40 & 140 & 200 - \lambda \end{bmatrix} \right) = 0 \Rightarrow -\lambda^3 + 450\lambda - 32400\lambda = 0 \\ &\Rightarrow \lambda = [0, 90, 360]\end{aligned}$$

Solving for the eigen-vectors using the following equation

$$\begin{bmatrix} 80 & 100 & 40 \\ 100 & 170 & 140 \\ 40 & 140 & 200 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 80x + 100y + 40z \\ 100x + 170y + 140z \\ 40x + 140y + 200z \end{bmatrix} = \lambda_i \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (4)$$

For different  $\lambda$  values, we get

$$\begin{aligned}\lambda_1 = 0 &\rightarrow \begin{bmatrix} 80x + 100y + 40z \\ 100x + 170y + 140z \\ 40x + 140y + 200z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \Rightarrow \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 2z \\ -2z \\ z \end{bmatrix} \\ \lambda_2 = 90 &\rightarrow \begin{bmatrix} 80x + 100y + 40z \\ 100x + 170y + 140z \\ 40x + 140y + 200z \end{bmatrix} = \begin{bmatrix} 90x \\ 90y \\ 90z \end{bmatrix} \Rightarrow \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -z \\ -0.5z \\ z \end{bmatrix} \\ \lambda_3 = 360 &\rightarrow \begin{bmatrix} 80x + 100y + 40z \\ 100x + 170y + 140z \\ 40x + 140y + 200z \end{bmatrix} = \begin{bmatrix} 360x \\ 360y \\ 360z \end{bmatrix} \Rightarrow \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0.5z \\ z \\ z \end{bmatrix}\end{aligned}$$

This gives the potential candidates for  $U$ . For  $V$ , we get the eigen-vectors of  $M_R$ . First calculating eigen-values using

$$\begin{aligned}\det(M_R - \lambda I) = 0 &\Rightarrow \det \left( \begin{bmatrix} 333 - \lambda & 81 \\ 81 & 117 - \lambda \end{bmatrix} \right) = \lambda^2 - 450\lambda + 32400 = 0 \\ &\Rightarrow \lambda = [90, 360]\end{aligned}$$

Solving for eigen-vectors using the following equation

$$\begin{bmatrix} 333 & 81 \\ 81 & 117 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 333x + 81y \\ 81x + 117y \end{bmatrix} = \lambda_i \begin{bmatrix} x \\ y \end{bmatrix} \quad (5)$$

For different  $\lambda$  values, we get

$$\begin{aligned}\lambda_1 = 90 &\rightarrow \begin{bmatrix} 333x + 81y \\ 81x + 117y \end{bmatrix} = \begin{bmatrix} 90x \\ 90y \end{bmatrix} \Rightarrow \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -y/3 \\ y \end{bmatrix} \\ \lambda_2 = 360 &\rightarrow \begin{bmatrix} 333x + 81y \\ 81x + 117y \end{bmatrix} = \begin{bmatrix} 360x \\ 360y \end{bmatrix} \Rightarrow \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 3y \\ y \end{bmatrix}\end{aligned}$$

All the above eigen-vectors can be assumed to be unit vectors (so that the matrices  $U$  and  $V$  become orthogonal). The resultant matrix is given by

$$\begin{aligned}M = U\Sigma V^T &\Rightarrow \begin{bmatrix} 4 & 8 \\ 11 & 7 \\ 14 & -2 \end{bmatrix} = \begin{bmatrix} 0.5z_1 & -z_2 & 2z_3 \\ z_1 & -0.5z_2 & -2z_3 \\ z_1 & z_2 & z_3 \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 3y_1 & -y_2/3 \\ y_1 & y_2 \end{bmatrix}^T \\ &\Rightarrow M = \begin{bmatrix} 4 & 8 \\ 11 & 7 \\ 14 & -2 \end{bmatrix} = \begin{bmatrix} 1.5\sigma_1 y_1 z_1 + \frac{\sigma_2 y_2 z_2}{3} & 0.5\sigma_1 y_1 z_1 - \sigma_2 y_2 z_2 \\ 3\sigma_1 y_1 z_1 + \frac{5\sigma_2 y_2 z_2}{30} & \sigma_1 y_1 z_1 - 0.5\sigma_2 y_2 z_2 \\ 3\sigma_1 y_1 z_1 - \frac{\sigma_2 y_2 z_2}{3} & \sigma_1 y_1 z_1 + \sigma_2 y_2 z_2 \end{bmatrix}\end{aligned}$$

Since vectors in  $U$  are unit vectors:  $z_1 = \pm 2/3$ ,  $z_2 = \pm 2/3$ ,  $z_3 = \pm 1/3$ . Since vectors in  $V$  are also unit vectors:  $y_1 = \pm 1/\sqrt{10}$ ,  $y_2 = \pm 3/\sqrt{10}$ . Using only the +ve values, we get the following

$$M = \begin{bmatrix} 4 & 8 \\ 11 & 7 \\ 14 & -2 \end{bmatrix} = \begin{bmatrix} 0.316227\sigma_1 + 0.210818\sigma_2 & 0.105409\sigma_1 - 0.632455\sigma_2 \\ 0.632455\sigma_1 + 0.105409\sigma_2 & 0.210818\sigma_1 - 0.316227\sigma_2 \\ 0.632455\sigma_1 - 0.210818\sigma_2 & 0.210818\sigma_1 + 0.632455\sigma_2 \end{bmatrix} \Rightarrow \begin{bmatrix} \sigma_1 \\ \sigma_2 \end{bmatrix} = \begin{bmatrix} 18.973665 \\ -9.486832 \end{bmatrix}$$

But since singular values **have** to be positive, we can change the sign of  $z_2$  and thereby consider  $z_2 = -2/3$ . This changes the above equation as

$$M = \begin{bmatrix} 4 & 8 \\ 11 & 7 \\ 14 & -2 \end{bmatrix} = \begin{bmatrix} 0.316227\sigma_1 - 0.210818\sigma_2 & 0.105409\sigma_1 + 0.632455\sigma_2 \\ 0.632455\sigma_1 - 0.105409\sigma_2 & 0.210818\sigma_1 + 0.316227\sigma_2 \\ 0.632455\sigma_1 + 0.210818\sigma_2 & 0.210818\sigma_1 - 0.632455\sigma_2 \end{bmatrix} \Rightarrow \begin{bmatrix} \sigma_1 \\ \sigma_2 \end{bmatrix} = \begin{bmatrix} 18.973665 \\ 9.486832 \end{bmatrix}$$

The above equations give

$$U = \begin{bmatrix} 0.\bar{3} & 0.\bar{6} & 0.\bar{6} \\ 0.\bar{6} & 0.\bar{3} & -0.\bar{6} \\ 0.\bar{6} & -0.\bar{6} & 0.\bar{3} \end{bmatrix} \quad \Sigma = \begin{bmatrix} 18.973665 & 0 \\ 0 & 9.486832 \\ 0 & 0 \end{bmatrix} \quad V = \begin{bmatrix} 0.948683 & -0.316227 \\ 0.316227 & 0.948683 \end{bmatrix} \quad (6)$$

Hence, the singular value decomposition of  $M$  is given by

$$M = U\Sigma V^\top = \begin{bmatrix} 0.\bar{3} & 0.\bar{6} & 0.\bar{6} \\ 0.\bar{6} & 0.\bar{3} & -0.\bar{6} \\ 0.\bar{6} & -0.\bar{6} & 0.\bar{3} \end{bmatrix} \begin{bmatrix} 18.973665 & 0 \\ 0 & 9.486832 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0.948683 & 0.316227 \\ -0.316227 & 0.948683 \end{bmatrix} = \begin{bmatrix} 4 & 8 \\ 11 & 7 \\ 14 & -2 \end{bmatrix}$$

The equation 6 gives the SVD of  $M$ .

## 2 LDA and PCA

### 2.1 A: Points about PCA

Suppose we have our 2D data as  $\mathbf{X}$  and we decompose (the covariance matrix of)  $\mathbf{X}$  as  $\mathbf{UDV}^\top$ , then which of the following are correct

(a) **PCA can be useful if all elements of  $\mathbf{D}$  are equal** This is **false**. If all elements in  $\mathbf{D}$  are equal, then the data is uniformly spread across all principal components. Removing any will remove an equal fraction of useful statistical information from the data. PCA (projection onto a subset of principal components) won't be useful then.

(b) **PCA can be useful if all elements of  $\mathbf{D}$  are not equal** This is **true**, depending on the particular values (how spread they are). If the values are different, then the different principal components have different contribution towards the data, hence a sub-space of components with highest values in  $\mathbf{D}$  can be created without too much loss in statistical information.

(c)  **$\mathbf{D}$  is not full-rank if all points in  $\mathbf{X}$  lie on a straight line** This is **true**. If all points in  $\mathbf{X}$  are on a straight line, then there is no statistical information along any axis other than the one on which the data is (the line). Therefore, the primary principal axis contains a value but the second principal axis would contain 0 value. Hence,  $\mathbf{D}$  is not a full-rank matrix (it's a diagonal matrix, with the second - or the last - element being 0).

(d)  **$\mathbf{V}$  is not full-rank if all points in  $\mathbf{X}$  lie on a straight line** This is **false**. Since the matrix whose SVD is being done is *real*, the matrices  $\mathbf{U}$  and  $\mathbf{V}$  can be guaranteed to be real orthogonal matrices (which are full rank). Hence,  $\mathbf{V}$  is a full-rank matrix.

(e)  **$\mathbf{D}$  is not full-rank if all points in  $\mathbf{X}$  lie on a circle** This is **false**. If all points in  $\mathbf{X}$  lie on a circle in 2D, then they're distributed equally about any two orthogonal principal axis. Therefore, the values of the diagonal elements are going to be non-zero. Hence, the matrix  $\mathbf{D}$  is going to be full rank.

### 2.2 B: True / False

**Statement:** PCA will project the data points (multi-class) on a line which preserve information useful for data classification.

#### TL;DR

This is **false**. PCA (Principal Component Analysis) projects into a subspace of principal components, whereas LDA (Linear Discriminant Analysis) projects points such that inter-class variance (from mean) is maximized whereas intra-class variance is minimized.

#### PCA

Principal Component Analysis (PCA) projects the data points onto the principal components (eigenvectors of the covariance matrix of data) that preserve the most amount of statistical information (have the highest eigen-values / singular values). This is an *unsupervised* dimensionality reduction algorithm.

#### LDA

Linear Discriminant Analysis (LDA) projects the *labelled* data such that the resulting projection has a large distance between the mean of the classes while the classes have a small intra-class variance. That is, multi-class data is divided into small clusters that are physically apart in the smaller dimensional subspace. This is a *supervised* dimensionality reduction algorithm (the class labels are required).