Assignment-2

Laksh Nanwani

2021701002

Question 1: Eigen Values and Eigen Vectors

A. Among Eigen Value Decomposition and Singular Value Decomposition, which one is more generalizable to matrices and why?

$A = U \Sigma V^T$ $\qquad\qquad$ $A = V \lambda V^{-1}$

(A) Singular Value Decomposition is more generalisable than Eigen Value Decomposition.

EVD can only be applied to square matrices and even then does not always have a solution.

Whereas, SVD can be applied to any rectangular or square matrices, and always has a solution.

SVD is also faster in time complexity than EVD for larger matrices.

B. Show the method and find the Singular Value Decomposition of the following matrix::

$$M = \begin{bmatrix} 4 & 8 \\ 11 & 7 \\ 14 & -2 \end{bmatrix} \tag{1}$$

(B) $m = \begin{bmatrix} 4 & 8 \\ 11 & 7 \\ 14 & -2 \end{bmatrix} \equiv A$ $\qquad$ Show the method and find the Singular Value Decompositi

$\rightarrow$ $V^T = \text{eigenvectors}(A^TA)^T = \begin{bmatrix} v_1^T \\ v_2^T \end{bmatrix} = [v_1 \ \ v_2]^T$

$U = \left[\frac{1}{\sigma_1} Av_1 \quad \frac{1}{\sigma_2} Av_2 \quad \frac{NS(A^T)}{|NS(A^T)|} \right]$ $\qquad$ $\Sigma = \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \\ 0 & 0 \end{bmatrix}$, $\sigma_n = \sqrt{\lambda_n}$

NS = null space

$A = U \Sigma V^T$

$$A^TA = \begin{bmatrix} 4 & 11 & 14 \\ 8 & 7 & -2 \end{bmatrix} \begin{bmatrix} 4 & 8 \\ 11 & 7 \\ 14 & -2 \end{bmatrix} = \begin{bmatrix} 333 & 81 \\ 81 & 117 \end{bmatrix} = N$$

Finding λ, $\det(N-\lambda I)=0 \Rightarrow \begin{vmatrix} 333-\lambda & 81 \\ 81 & 117-\lambda \end{vmatrix} = 0$

$$\Rightarrow (333-\lambda)(117-\lambda)-81^2=0 \Rightarrow \lambda^2-450\lambda+32400=0$$

$$\therefore \lambda = \frac{450 \pm \sqrt{450\times450-4\times32400}}{2\times1} \Rightarrow \boxed{\lambda_1=360 \ , \ \lambda_2=90}$$

$$\Rightarrow \boxed{\sigma_1 = 6\sqrt{10} \ , \ \sigma_2 = 3\sqrt{10}} \Rightarrow \boxed{\Sigma = \begin{bmatrix} 6\sqrt{10} & 0 \\ 0 & 3\sqrt{10} \\ 0 & 0 \end{bmatrix}}$$

Calculating v_1 and $v_2 \rightarrow$ $Av_n = \lambda_n v_n$

$$\begin{bmatrix} 333 & 81 \\ 81 & 117 \end{bmatrix}\begin{bmatrix} a \\ b \end{bmatrix} = 360\begin{bmatrix} a \\ b \end{bmatrix} \Rightarrow \left.\begin{matrix} 81b = 27a \\ \text{and } 81a = 243b \end{matrix}\right\} \Rightarrow a = 3b$$

$$\therefore v_1 = b\begin{bmatrix} 3 \\ 1 \end{bmatrix} \longrightarrow \text{orthonormal} \Rightarrow \boxed{v_1 = \begin{bmatrix} 3/\sqrt{10} \\ 1/\sqrt{10} \end{bmatrix}}$$

$$\begin{bmatrix} 333 & 81 \\ 81 & 117 \end{bmatrix}\begin{bmatrix} c \\ d \end{bmatrix} = 90\begin{bmatrix} c \\ d \end{bmatrix} \Rightarrow \left.\begin{matrix} 243c = -81d \\ 81c = -27d \end{matrix}\right\} \Rightarrow d = -3c$$

$$\therefore v_2 = c\begin{bmatrix} 1 \\ -3 \end{bmatrix} \longrightarrow \text{orthonormal} \Rightarrow \boxed{v_2 = \begin{bmatrix} 1/\sqrt{10} \\ -3/\sqrt{10} \end{bmatrix}}$$

$$\therefore V^T = \begin{bmatrix} 3/\sqrt{10} & 1/\sqrt{10} \\ 1/\sqrt{10} & -3/\sqrt{10} \end{bmatrix}^T \Rightarrow \boxed{V^T = \begin{bmatrix} 3/\sqrt{10} & 1/\sqrt{10} \\ 1/\sqrt{10} & -3/\sqrt{10} \end{bmatrix}}$$

$$\frac{Av_1}{\sigma_1} = \frac{1}{6\sqrt{10}}\begin{bmatrix} 4 & 8 \\ 11 & 7 \\ 14 & -2 \end{bmatrix}\begin{bmatrix} 3/\sqrt{10} \\ 1/\sqrt{10} \end{bmatrix} = \frac{1}{60}\begin{bmatrix} 20 \\ 40 \\ 40 \end{bmatrix} = \boxed{\begin{bmatrix} 1/3 \\ 2/3 \\ 2/3 \end{bmatrix} = \frac{Av_1}{\sigma_1}}$$

$$\frac{Av_2}{\sigma_2} = \frac{1}{3\sqrt{10}}\begin{bmatrix} 4 & 8 \\ 11 & 7 \\ 14 & -2 \end{bmatrix}\begin{bmatrix} 1/\sqrt{10} \\ -3/\sqrt{10} \end{bmatrix} = \frac{1}{30}\begin{bmatrix} -20 \\ -10 \\ 20 \end{bmatrix} = \boxed{\begin{bmatrix} -2/3 \\ -1/3 \\ 2/3 \end{bmatrix} = \frac{Av_2}{\sigma_2}}$$

Let $NS(A^T) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = X \Rightarrow A^T X = 0 \Rightarrow \begin{bmatrix} 4 & 11 & 14 \\ 8 & 7 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = 0$

$\therefore \; 4x_1 + 11x_2 + 14x_3 = 0 \; - (1)$ and $8x_1 + 7x_2 - 2x_3 = 0 \; - (2)$

$(1) + 7 \times (2) \rightarrow \; 60x_1 + 60x_2 = 0 \Rightarrow x_1 = -x_2$ → putting in (2)

$-8x_2 + 7x_2 - 2x_3 = 0 \Rightarrow 2x_3 = -x_2$

$\therefore \; x_1 = -x_2 = 2x_3 \Rightarrow X = \begin{bmatrix} -x_2 \\ x_2 \\ -x_2/2 \end{bmatrix} = x_2 \begin{bmatrix} -1 \\ 1 \\ -0.5 \end{bmatrix}$

$|X| = |NS(A^T)| = \sqrt{x_2^2(1+1+1/4)} = \frac{3}{2} x_2 \Rightarrow \boxed{\frac{NS(A^T)}{|NS(A^T)|} = \begin{bmatrix} -2/3 \\ 2/3 \\ -1/3 \end{bmatrix}}$

$$\Rightarrow \boxed{U = \begin{bmatrix} 1/3 & -2/3 & -2/3 \\ 2/3 & -1/3 & 2/3 \\ 2/3 & 2/3 & -1/3 \end{bmatrix}}$$

$$\therefore \boxed{\underbrace{\begin{bmatrix} 4 & 8 \\ 11 & 7 \\ 14 & -2 \end{bmatrix}}_{M} = \underbrace{\begin{bmatrix} 1/3 & -2/3 & -2/3 \\ 2/3 & -1/3 & 2/3 \\ 2/3 & 2/3 & -1/3 \end{bmatrix}}_{U} \underbrace{\begin{bmatrix} 6\sqrt{10} & 0 \\ 0 & 3\sqrt{10} \\ 0 & 0 \end{bmatrix}}_{\Sigma} \underbrace{\begin{bmatrix} 3/\sqrt{10} & 1/\sqrt{10} \\ 1/\sqrt{10} & -3/\sqrt{10} \end{bmatrix}}_{V^T}}$$

Question 2: LDA and PCA

A. Suppose you want to apply PCA to your data X which is in 2D and you decompose X as UDV^T. Then, which of the following are correct:

(*a*) PCA can be useful if all elements of D are equal

(*b*) PCA can be useful if all elements of D are not equal

(*c*) D is not full-rank if all points in X lie on a straight line

(*d*) V is not full-rank if all points in X lie on a straight line

(*e*) D is not full-rank if all points in X lie on a circle

A Correct options –

b) PCA can be useful if all elements of D are not equal

c) D is not full-rank if all points in X lie on a straight line

B. True/False

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

B False

Question 3: Bayes Theorem

A. What is the difference between prior and posterior probabilities?

A Prior probability is the probability of an event occuring before considering any evidence into account. Whereas, posterior probability is the probability of the event when an evidence /hypothesis has been presented and taken into account.

$$P(A|B) = \frac{P(A) \times P(B|A)}{P(B)}$$

$P(A)$ = prior probability

$P(A|B)$ = posterior probability

B. Let's say that you are at work one day and have just finished lunch. You suddenly feel horrible and find yourself lying down. Maybe it is because one of your friend was recently sick with flu.

You have a headache and sore throat, and you know that people with the flu have the same symptoms roughly 90% of the time. In other words, 90% of people with the flu have the same symptoms you currently have.

Wanting to gain a little more information you roll over, grab your phone and search Google. You find a reputable article that says that only 5% of the population will get the flu in a given year. Or, the probability of having the flu, in general, is only 5%.

You then spot one more statistic that says 20% of the population in a given year will have a headache and sore throat at any given time.

What is the probability of you having a flu given you have a sore throat and headache?

(B) Let, probability that population has flu = $P(F)$ ~~= 5%~~ $\therefore P(F) = 5\% = 0.05$

probability that people have headache and sore throat $= P(H) \Rightarrow P(H) = 20\% = 0.2$

probability that people having flu also have headache and sore throat $= P(H|F) \Rightarrow p(H|F) = 90\% = 0.9$

and probability that a person has flu when they have headache and sore throat $= P(F|H)$

Using Baye's theorem →

$$P(F|H) = \frac{p(F) \times p(H|F)}{p(H)} = \frac{0.05 \times 0.9}{0.2} = 0.225 = 22.5\%$$

$$\Rightarrow \boxed{p(F|H) = 22.5\% = 0.225}$$

$\therefore$ Probability that a person has flu given they have a sore throat and headache is 22.5% or 0.225.

Question 4: K-Nearest Neighbors

Write a code to perform KNN classification on Iris dataset provided. Use the statrter code for loading the train, test dataset. Report the accuracy obtained on test dataset. Do not use direct inbuilt functions. Numpy or other math libraries are allowed.

Refer python notebook – KNN classifier.ipynb

Question 5: Logistic Regression

For the sample dataset provided, write a code to perform logistic regression. Plot a decision boundary between the two classes. Sample result image is provided. Do not use direct inbuilt functions. Numpy or other math libraries are allowed.

Refer python notebook – Logistic Regression.ipynb

------ x ------