Department of Computer Science
Cochin University of Science and Technology
Machine Learning Algorithms - Assignment 02

Algorithms for Dimensionality Reduction in Machine Learning

Aneetta Sara Shany
M.Tech Computer Science & Engineering - Software Engineering

January 12, 2023

Introduction

What is Dimensionality Reduction ?
Why is Dimension Reduction is important in Machine Learning predictions?

We often encounter high dimensional data (more redundant features) when we work with real world problems. Dimension reduction is a process of converting a set of data to visualize and work on the training dataset. Dimensionality reduction algorithms are used to solve ML problems like classification and regression with improved accuracy by reducing the features of the dataset. There are various methods used for dimensionality reduction which include linear dimensionality reduction techniques such as Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Singular Value Decomposition (SVD). We will be focusing more on LDA and SVD in this assignment.

LDA is a linear combination, where we mathematically apply functions to that set of various data items to separately analyze multiple classes of objects or items. LDA as a supervised linear transformation technique aims to find the feature subspace that optimizes class separability.

SVD as an unsupervised linear transformation technique helps us identify patterns in data based of the correlation between the features. It doesn't center the data before computing the singular value decomposition. Therefore it can work with sparse matrices efficiently.

Linear Discriminant Analysis

LDA is effective for multi-class classification tasks as the linear discriminants create decision boundaries and maximize the separation between multiple classes.

Example: Illustrate LDA computation to separate two input matrices each representing class C1 and C2 respectively.

$$X_1 = \begin{bmatrix} 4 & 2 \\ 2 & 4 \\ 2 & 3 \\ 3 & 6 \\ 4 & 4 \end{bmatrix} \qquad X_2 = \begin{bmatrix} 9 & 10 \\ 6 & 8 \\ 9 & 5 \\ 8 & 7 \\ 10 & 8 \end{bmatrix}$$

Solution:
LDA transforms the input matrix into a plot using projection vector to project the data in order to maximize classes separability.
Following are the computations to find the projection vector:

$$S_w = \sum S_i$$
$$S_i = \sum (X_i - \mu_i)(X_i - \mu_i)^T$$
$$S_b = \sum_i n_i(\mu_i - \mu)(\mu_i - \mu)^T$$

where
S_w - **Scatter matrix within class**
S_b - **Scatter matrix between class**
S_i - **Covariance matrix of each classes**
i - Number of output classes
μ_i - Average of features X_i
μ - Global mean of features X_i of the whole dataset

Step 1: Compute the scatter matrices and covariance matrices for given two classes:

$$\mu_1 = 1/n \sum x_i$$
$$\mu_2 = 1/n \sum x_i$$
$$S_b = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$
$$S_1 = \sum (x_1 - \mu_1)(X_1 - \mu_1)^T$$
$$S_2 = \sum (x_2 - \mu_2)(X_2 - \mu_2)^T$$
$$S_w = S_1 + S_2$$

Step 2: Compute eigenvalues using the equation $|S_w^{-1} S_b - \lambda I| = 0$

Step 3: Compute eigenvectors using ordered eigenvalues from Step 2.

Step 4: Sort the eigenvectors from highest to lowest depending on their corresponding eigenvalue and then choose the top k eigenvectors, where k is the number of dimensions we want to keep.

Step 5: Compute the projection vector/Discriminant function $Z = W^T X$ where W is the eigenvector matrix with k features and X is the adjusted data.

$$\mu_1 = \begin{bmatrix} 3 & 3.8 \end{bmatrix} \qquad \mu_2 = \begin{bmatrix} 8.4 & 7.6 \end{bmatrix} \qquad (\mu_1 - \mu_2) = \begin{bmatrix} -5.4 & -3.8 \end{bmatrix} \qquad (\mu_1 - \mu_2)^T = \begin{bmatrix} -5.4 \\ -3.8 \end{bmatrix}$$

$$S_b = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T = \begin{bmatrix} 29.16 & 20.52 \\ 20.52 & 14.44 \end{bmatrix}$$

To compute S_1 and S_2 which are the covariance matrices of X_1 and X_2 respectively:

$$Covariance \quad Matrix \quad for \quad the \quad sample \quad X \quad = \begin{bmatrix} var(x1) & cov(x1, x2) \\ cov(x1, x2) & var(x2) \end{bmatrix}$$

- Subtract the mean from all (n) elements from x1 of X_1.
  $(4-3), (2-3), (2-3), (3-3), (4-3)$
- Take the sum of the squares of the differences obtained in the previous step.
  $(4-3)^2, (2-3)^2, (2-3)^2, (3-3)^2, (4-3)^2$

- Divide above value by (n - 1) to get the sample variance of x1.
  var(x1) = $[(4-3)^2 + (2-3)^2 + (2-3)^2 + (3-3)^2 + (4-3)^2]$ / (5 - 1) = 1

- Similarly find the sample variance of x2 using above steps.
  var(x2) = $[(2-3.8)^2 + (4-3.8)^2 + (3-3.8)^2 + (6-3.8)^2 + (4-3.8)^2]$ / (5 - 1) = 2.2

- Choose a pair of variables x1 and x2.

- Multiply all pairs of x1 and x2 such that (x1 - mean(x1)) * (x2 - mean(x2))
  $(4-3)(2-3.8), (2-3)(4-3.8), (2-3)(3-3.8), (3-3)(6-3.8), (4-3)(4-3.8)$

- Add these values and divide them by (n - 1) to get the covariance.
  cov(x1, x2) = **cov(x2, x1)** = [(4-3)(2-3.8) + (2-3)(4-3.8) + (2-3)(3-3.8) + (3-3)(6-3.8) + (4-3)(4-3.8)] / (5 - 1) = -0.25

$$S_1 = \begin{bmatrix} 1 & -0.25 \\ -0.25 & 2.2 \end{bmatrix} \qquad Similarly, \quad S_2 = \begin{bmatrix} 2.3 & -0.05 \\ -0.05 & 3.3 \end{bmatrix}$$

$$S_w = S_1 + S_2 = \begin{bmatrix} 3.3 & -0.3 \\ -0.3 & 5.5 \end{bmatrix} \qquad S_w^{-1} = \begin{bmatrix} 0.3045 & 0.0166 \\ 0.0166 & 0.1827 \end{bmatrix} \qquad S_w^{-1}S_b = \begin{bmatrix} 9.2213 & 6.489 \\ 4.2339 & 2.9794 \end{bmatrix}$$

$$|S_w^{-1}S_b - \lambda I| = 0$$
$$(9.2213 - \lambda)(2.9794 - \lambda) - (6.489)(4.2339) = 0$$
$$\lambda^2 - 12.2007\lambda = 0$$
$$\lambda_1 = 0 \quad , \quad \lambda_2 = 12.2007$$

Refer SVD to find how to compute the eigenvectors from eigenvalues.

$$Eigenvector for \lambda_1 = 0 : \quad e_1 = \begin{bmatrix} -0.5755 \\ 0.8178 \end{bmatrix} \qquad Eigenvector for \lambda_2 = 12.2007 : \quad e_2 = \begin{bmatrix} 0.9088 \\ 0.4173 \end{bmatrix}$$

$$W = \begin{bmatrix} e_1 & e_2 \end{bmatrix} = \begin{bmatrix} -0.5755 & 0.9088 \\ 0.8178 & 0.4173 \end{bmatrix}$$

$$\text{Projection} \quad \text{vector} \quad : Z = W^T X$$

Hence, Projection vector to separate input matrices X1 and X2 is computed.

Singular Value Decomposition

SVD retains the basic structure of the data as the orthogonal matrices capture the structure of the input matrix so that their properties do not change when multiplied by other numbers. This can help us in the task of reducing the number of features in the input data set.

Example: Illustrate SVD computation to decompose given input matrix.

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Solution:
SVD of a matrix is a factorization of input matrix into three feature matrices.
The formula for SVD is as follows:

$$A = USV^T$$

where
A - **input matrix**
U - **decomposition matrix** with left singular vectors
S - **diagonal matrix**
V - **feature reduced matrix** with right singular vectors

Step 1: Converting A, a rectangular matrix of shape 3 x 2, to a square matrix by multiplying A with its transpose for easier computation. Compute A^T, A^TA.

Step 2: Compute eigenvalues and singular values from the input matrix. Determine the eigenvalues of A^TA using the equation $|A^TA - \lambda I| = 0$. Sort the values in descending order. Obtain singular values by finding the square roots of eigenvalues.

Step 3: Construct a diagonal matrix S with singular values. Also, find S^{-1}.

Step 4: Compute eigenvectors using ordered eigenvalues from Step 2. Place eigenvectors along each column of matrix V and Compute V^T.

Step 5: Compute $U = AVS^{-1}$ and Verify the complete SVD process using the formula $A = USV^T$

Following are the computations :

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad A^T = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \qquad A^TA = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \qquad A^TA - \lambda I = \begin{bmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{bmatrix}$$

$$|A^TA - \lambda I| = 0$$
$$(2-\lambda)^2 - 1 = 0$$
$$\lambda^2 - 4\lambda + 3 = 0$$
$$\lambda_1 = 3 \quad , \quad \lambda_2 = 1 \quad , \quad \sqrt{\lambda_1} = \sqrt{3} \quad , \quad \sqrt{\lambda_2} = \sqrt{1}$$

Eigenvalues :$\lambda_1 = 3 \quad , \quad \lambda_2 = 1$ *and* **Singular values :**$s_1 = \sqrt{3} \quad , \quad s_2 = \sqrt{1}$

$$S = \begin{bmatrix} 1.732 & 0 \\ 0 & 1 \end{bmatrix} \qquad S^{-1} = \begin{bmatrix} 0.5773 & 0 \\ 0 & 1 \end{bmatrix}$$

For λ_1 = 3, $|A^TA - 3I| = 0$

$$A^TA - \lambda_1 = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \qquad X_1 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \qquad (A^TA - \lambda_1)X_1 = 0 \Rightarrow \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$-x_1 + x_2 = 0 \quad and \quad x_1 - x_2 = 0 \quad \Rightarrow \quad x_1 = 1 \quad , \quad x_2 = 1 \quad \Rightarrow \quad L = \sqrt{x_1^2 + x_2^2} = \sqrt{2}$$

$$\textbf{Eigen vector} \quad e_1 = \begin{bmatrix} x_1/L \\ x_2/L \end{bmatrix} = \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix} = \begin{bmatrix} 0.7071 \\ 0.7071 \end{bmatrix}$$

For $\lambda_2 = 2$, $|A^TA - I| = 0$

$$A^TA - \lambda_2 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \qquad X_1 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \qquad (A^TA - \lambda_2)X_2 = 0 \Rightarrow \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$x_1 + x_2 = 0 \quad and \quad x_1 + x_2 = 0 \quad \Rightarrow \quad x_1 = 1 \quad , \quad x_2 = -1 \quad \Rightarrow \quad L = \sqrt{x_1^2 + x_2^2} = \sqrt{2}$$

$$\textbf{Eigen vector} \quad e_2 = \begin{bmatrix} x_1/L \\ x_2/L \end{bmatrix} = \begin{bmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{bmatrix} = \begin{bmatrix} 0.7071 \\ -0.7071 \end{bmatrix}$$

Computing the orthogonal matrices V^T and U.

$$V = \begin{bmatrix} e_1 & e_2 \end{bmatrix} = \begin{bmatrix} 0.7071 & 0.7071 \\ 0.7071 & -0.7071 \end{bmatrix} \qquad V^T = \begin{bmatrix} 0.7071 & 0.7071 \\ 0.7071 & -0.7071 \end{bmatrix}$$

$$U = AVS^{-1} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0.7071 & 0.7071 \\ 0.7071 & -0.7071 \end{bmatrix} \begin{bmatrix} 0.5773 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.8164 & 0 \\ 0.4084 & 0.7071 \\ 0.4084 & -0.7071 \end{bmatrix}$$

Calculated matrices U, S and V and verified the decomposition of the input matrix A into three matrices by computing the left hand side and right hand side of the equation given below.

$$A = USV^T$$

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.8164 & 0 \\ 0.4084 & 0.7071 \\ 0.4084 & -0.7071 \end{bmatrix} \begin{bmatrix} 1.732 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0.7071 & 0.7071 \\ 0.7071 & -0.7071 \end{bmatrix}$$

Hence, Singular value decomposition of input matrix A is computed.

. . . .