Homework week 5

Nguyễn Minh Đức
Student code: 11204838

December 2022

1 Question 1

We have

$$X = \begin{bmatrix} —x_1^T— \\ —x_2^T— \\ ... \\ —x_n^T— \end{bmatrix} \in R^{n*D}$$

If we want to reduce the dimension of X by choosing M most important features in X (M < D), doing that by create B matrix

$$B = \begin{bmatrix} | & | & & | \\ b_1 & b_2 & ... & b_m \\ | & | & & | \end{bmatrix} \in R^{D*M}$$

$$R^D \to R^M : \begin{cases} x_1 \to z_1 \\ ... \\ x_n \to z_n \end{cases}$$

$$Z = X*B = \begin{bmatrix} —x_1^T— \\ —x_2^T— \\ ... \\ —x_n^T— \end{bmatrix} \begin{bmatrix} | & | & & | \\ b_1 & b_2 & ... & b_m \\ | & | & & | \end{bmatrix} = \begin{bmatrix} x_1^T b_1 & x_1^T b_2 & & x_1^T b_m \\ x_2^T b_1 & x_2^T b_2 & & x_2^T b_m \\ ... & ... & ... & ... \\ x_n^T b_1 & x_n^T b_2 & & x_n^T b_m \end{bmatrix} = \begin{bmatrix} —z_1^T— \\ —z_2^T— \\ ... \\ —x_n^T— \end{bmatrix}$$

We want the new coordinate have the highest variance and the mean = 0. We consider b_1

$$\mu_x = \frac{x_1 + x_2 + ... + x_n}{N}$$

$$x^{'} = x - \mu_x \to \mu_{x'} = 0$$

$$\mu_z = \frac{x_1^{'T} * b_1 + x_2^{'T} * b_1 + ... + x_n^{'T} * b_1}{N} = \frac{b_1^T \sum_{i=1}^{N} x_i^{'}}{N} = b_1^T \mu_x^{'} = 0$$

$$
\begin{aligned}
Var_z &= \frac{\sum_{i=1}^{N}(x_i^{'T}b_1-\mu_z)^2}{N} = \frac{\sum_{i=1}^{N}(x_i^{'T}b_1)^2}{N} = \frac{\sum_{i=1}^{N}b_1^T x_i^{'} x_i^{'T} b_1}{N} \\
&= b_1^T \frac{\sum_{i=1}^{N}x_i^{'} x_i^{'T}}{N} b_1 = b_1^T \frac{\sum_{i=1}^{N}(x-\mu_x)(x-\mu_x)}{N} b_1 \\
&= b_1^T cov(X,X) b_1 = b_1^T S b_1 \text{ where S is coveriance matrix}
\end{aligned}
$$

If we increase the size of the vector b_1, Var will increase so we create a constraint $||b_1||_2^2 = 1$. As a result we have the constrained optimization problem to find out which direction projected data will varies most.

$$
\begin{aligned}
\max_{b_1} \quad & b_1^T S b_1 \\
\text{s.t.} \quad & ||b_1||_2^2 = 1
\end{aligned}
$$

Lagrangia function

$$
\begin{aligned}
\mathcal{L}(\lambda) &= b_1^T S b_1 + \lambda(1 - ||b_1||_2^2) \\
\frac{\partial L(\lambda)}{\partial b_1} &= 2b_1^T S - \lambda 2 b_1 = 0 \\
\rightarrow S b_1 &= \lambda b_1 \rightarrow \begin{cases} b_1 \text{ is eigenvector} \\ \lambda \text{ is eigenvalue} \end{cases}
\end{aligned}
$$

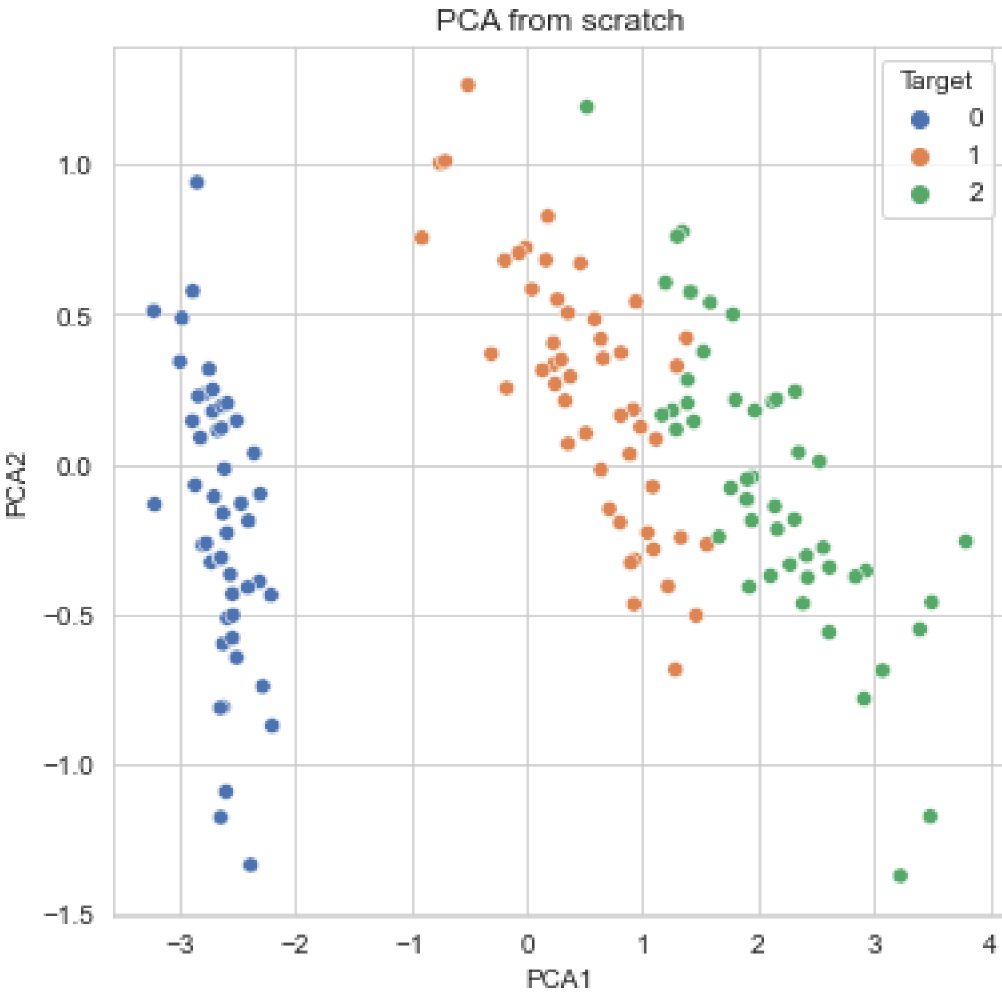
$$
\begin{aligned}
\frac{\partial L(\lambda)}{\partial \lambda} &= 1 - ||b_1||_2^2 = 0 \\
b_1^T b_1 = 1 &
\end{aligned}
$$

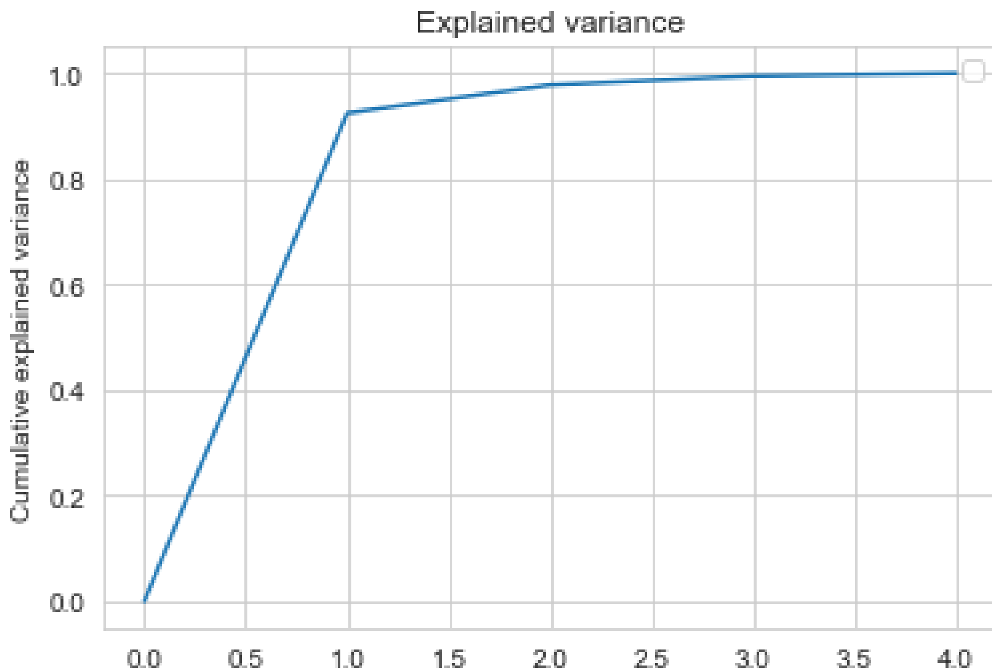
$$
Var = b_1^T S b_1 = b_1^T \lambda b_1 = \lambda b_1^T b_1 = \lambda
$$

We sort the eigenvalue in decending order and choose D fisrt largest eigenvector to create B matrix $\rightarrow Z = Bx^{'}$

2 Question 2

3 Question 3

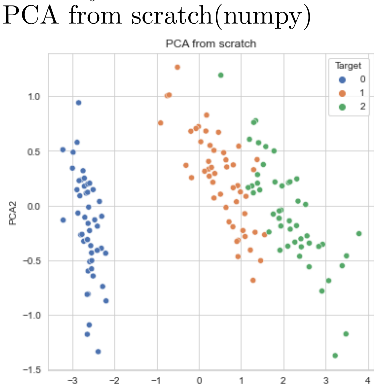




4 Question 4

The output of PCA algorithm in Sklearn package is little bit different from the PCA algorithm implemented from scratch. PCA algorithm after calculated eigenvalues and eigenvectors will make some changes. It will calculate the absolute values of each eigen vectors. If the greatest absolute value is negative, it will be multiphy by -1, otherwise eigenvector will remain the same. This different will lead to the different in eigenvectors and the final outputs. This problem really does not affect the classification model.

PCA from scratch(numpy)



PCA from sklearn