BITS F464 : Machine Learning
Assignment - 1
Fisher's Linear Discriminant

Omkar Pitale 2019A7PS0083H

Aditya Chopra 2019A7PS0178H

Anuradha Pandey 2019A7PS0265H

March 2021

1 Introduction

In Fisher's linear discriminant model for binary classification, we maximise a function that will give large separation between the projected class means while also giving minimum variance between each class. We try to analyse the performance of Fisher's linear discriminant on the given dataset by implementing it using Jupyter Notebooks, NumPy, Pandas and Matplotlib.

2 Model Description

2.1 Dataset and Pre-processing

Only one dataset was provided as described in Table 1. The dataset was sampled as PyData Dataframe, and then converted to two NumPy Arrays: Feature vectors and Targets. The Feature vectors were then divided into further two NumPy Arrays: positive class and negative class based on Target vector. The data was not split into training and testing sets as asked in the problem.

Name	dataset_FLD.csv
Type	csv
Number of features	3
Positive label	1
Negative label	0
Positive sample size	500
Negative sample size	500
Total sample size	1000

Table 1: Description of the Dataset

2.2 Approach

1. First we split the data into 2 classes (positive and negative) according to the target variable.

2. We calculate means of both classes i.e. $M_0 = \frac{1}{N_0} \sum_{n \in \mathcal{C}_0} x_n$ and $M_1 = \frac{1}{N_1} \sum_{n \in \mathcal{C}_1} x_n$, negative and positive respectively.

3. We calculate the total within class covariance matrix S_w which can be obtained as the sum of the covariance matrices of both classes, i.e. $S_w = S_0 + S_1$.

4. The unit vector onto which the points are projected can be found out by $\overrightarrow{w} = {S_w}^{-1}(M_1 - M_0)$ and $\hat{w} = \frac{\overrightarrow{w}}{\|\overrightarrow{w}\|}$.

5. Now we project the data points in the direction of the unit vector. Positive points have red colors and Negative points have blue color as shown in the below figures.

6. With the help of the reduced 1D features as inputs , we plot the Gaussian distributions for both the classes and find the intersection point (α) by solving the following quadratic equation -

$$\left(\frac{1}{\sigma_1^2} - \frac{1}{\sigma_0^2}\right) x^2 + 2\left(\frac{\mu_1}{\sigma_1^2} - \frac{\mu_0}{\sigma_0^2}\right) x + \frac{\mu_0^2}{\sigma_0^2} - \frac{\mu_1^2}{\sigma_1^2} + \log\left(\frac{\mu_0}{\mu_1}\right)$$

   where μ_0, μ_1 are means of negative and positive classes & σ_0^2, σ_1^2 are variances of negative and positve classes respectively.

7. The point α is the discriminating point (Threshold) in 1D space. The equation for decision boundary is found in the original feature space using $w^T x = \alpha$. Therefore if $w^T x > \alpha$ then x will be classified as positive point, else it will be a negative point.

8. Measures of Performance for this model:

$$\text{Accuracy} = \frac{\text{tp + tn}}{\text{tp + tn + fp + fn}}$$

$$\text{Precision} = \frac{\text{tp}}{\text{tp + fp}}$$

$$\text{Recall} = \frac{\text{tp}}{\text{tp + fn}}$$

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision + Recall}}$$

Results

Class statistics

3D weight vector : $\hat{w}$	$\begin{bmatrix}-0.00655686 & -0.01823739 & 0.99981218\end{bmatrix}$
1D threshold : α	-0.30381959407288783

Table 2: Weight Vector

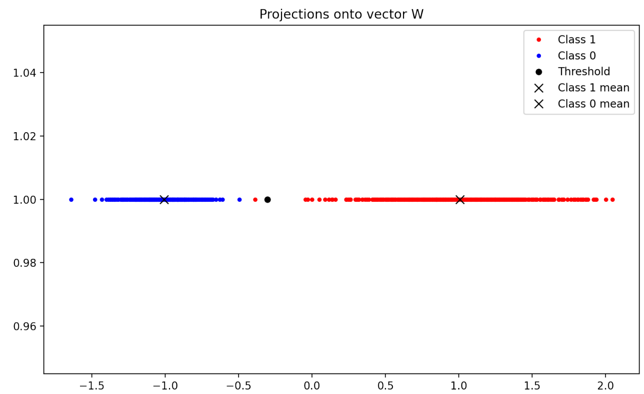
Performance :
Four Measures of Performance: Accuracy, Precision, Recall and F1-Score were considered.

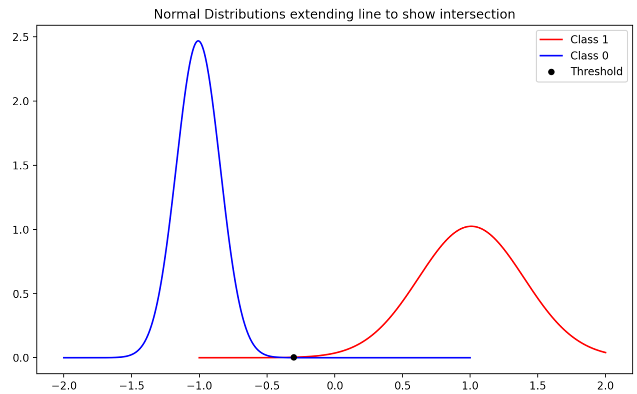
Class statistics		
	Negative Class	Positive Class
Mean	-1.0074971	1.0085714
S.D.	0.1616813	0.389815
Variance	0.026140	0.151956

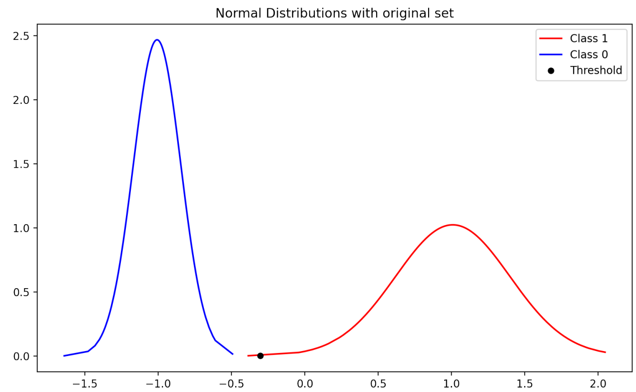
Performance Measure	Value
Accuracy	0.999
Precision	1.0
Recall	0.998
F1-Score	0.99899899

Table 3: Performance and statistics of Dataset

1D space







Original Feature space