# Introduction to Pattern Recognition Homework 2 Report

Andrés Ponce(彭思安) 0616110

May 1, 2020

# 1 Coding

## 1.1 Compute the mean vectors $m_i, (i = 1, 2)$ of each 2 classes on training data

To calculate the **mean vectors** for each class, we need to know which of the input vectors belong to $C_1$ and $C_2$, respectively. This is done by just checking which rows in `y_test` are 0 and 1, since those tell us which class the test input vectors belong to. After separating them and getting the mean for each column separately, we end with the result

$$m_1 = [2.47107265\ 1.97913899] \quad m_2 = [1.82380675\ 3.03051876]$$

## 1.2 Compute the within-class scatter matrix $S_W$ on training data.

The within class matrix $S_W$ measures how much the points in ever class differ from the mean. The within class variance for the dataset is the sum of the within-class variance for all the classes in our dataset. The result is a $2x2$ matrix with the values:

$$S_W = \begin{bmatrix} 140.40035447 & -5.30881553 \\ -5.30881553 & 138.14297637 \end{bmatrix}$$

## 1.3 Compute the between-class scatter matrix $S_B$ on training data.

The between-class covariance matrix $S_B$ measures how much the two class means vary from each other. The result is another $2x2$ matrix:

$$S_B = \begin{bmatrix} 0.41895314 & -0.68052227 \\ -0.68052227 & 1.10539942 \end{bmatrix}$$

## 1.4 Compute the Fisher's linear discriminant $W$ on training data.

The Fihser's Linear Discriminant can be calculated by the ratio of the between-class variance to the within-class variance. The linear discriminant comes out to be

$$w = \begin{bmatrix} -0.00432865 \\ 0.00744446 \end{bmatrix}$$

## 1.5 Project the testing data by linear discriminant to get the class prediction by nearest-neighbor rule and calculate your accuracy score on testing data.

The projection of the test data can be done once we have calculated the $w$ on the training data. For this assighment, we can call `accuracy_score(...)` from the `sklearn` module. To keep the code simple, but maybe at a loss of accuracy, only the nearest neighbor was calculated. For the testing data, we find the closest point in the training data and assign it the same class. The final result for accuracy from `sklearn.accuracy_score` was $0.88$.

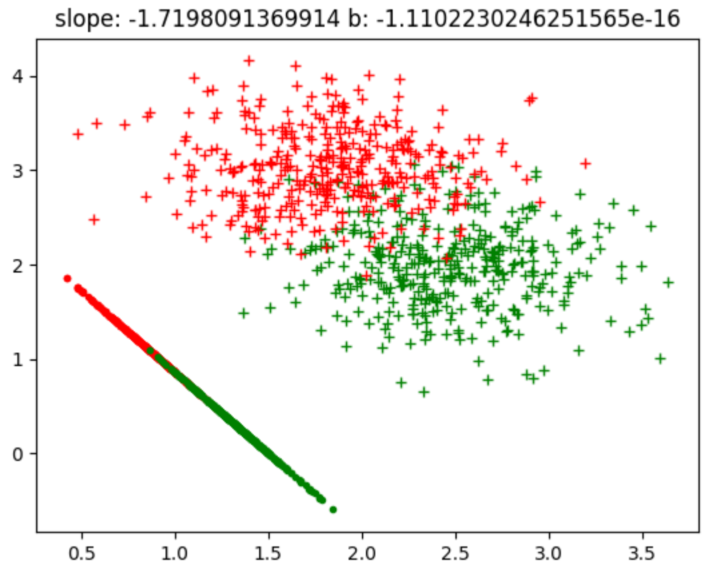


Figure 1: The class data is plotted along the line that maximizes the separation between the classes.

### 1.6 Plot the 1) best projection line on the training data and slow the slope and intercept on the title 2)colorize the data with each class 3) project all data points on your projection line.

Below is the plot for $C_1$ and $C_2$ on the testing data. The + symbols indicate the original class data received from x_train and y_train. The *overlapping* red and green circles are the class data projected onto the line that best separates the two classes.

## 2 Questions

### 2.1 Show that maximization of the class separation criterion given by $L(\lambda, w) = w^T(m_2 - m_1) + \lambda(w^T w - 1)$ with respect to w, using a Lagrange multiplier to enforce the constraint $w^T w = 1$, leads to the result that $w \propto (m_2 - m_1)$.

From the definition of the between-class covariance matrix, we know $S_B$ is always a scalar of $m_2 - m_1$. When we derive w.r.t. w, we get:

$$\frac{\partial w}{\partial L} = (m_2 - m_1) - \lambda w$$

In order to maximize this value, we can set the partial derivative of $L$ equal to 0, and achieve the following result

$$(m_2 - m_1) - \lambda w = 0$$

$$(m_2 - m_1) = \lambda w$$

And thus this implies that $w \propto (m_2 - m_1)$.

### 2.2 Using eq 1 and eq 2, derive the result eq 3 for the posterior class probability in the two-class generative model with Gaussian densities, and verify the results eq 4 and eq 5 for the parameters w and $w_0$.

$$\frac{1}{1 + exp(-\alpha) = \sigma(\alpha)} \tag{1}$$

$$a = ln\frac{p(x|C_1)p(C_1)}{p(x|C_2)p(C_2)} \tag{2}$$

$$p(C_1|x) = \sigma(w^T x + w_0) \tag{3}$$

$$w = \sum^{-1}(\mu_1 - \mu_2) \tag{4}$$

$$w_0 = -\frac{1}{2}\mu_1^T \sum{}^{-1}\mu_1 + \frac{1}{2}\mu_2^T \sum{}^{-1}\mu_2 + ln\frac{p(C_1)}{p(C_2)} \tag{5}$$

The posterior class probability is directly related to the **sigmoid function** becasue since all the real numbers live in (0,1), then a value closer to 1 will mean a higher chance p is in $C_1$. Thus, our input $w^T x + w_0$ will directly determine how confident we are in x belonging in $C1$, which shows that these two are equal.

For eq 4 and eq 5, assuming an equal covariance matrix for both classes, the square values will cancel out once the vector multiplications are done, and thus eq. 4 verifies $w_0$.