A Report for Machine Learning Lab 3

Ziyi Guo

zg2u21@soton.ac.uk

1 Class Boundaries and Posterior Probabilities

In this part, the problem of two-class classification in two dimensions was raised, in which features of the two classes were of Gaussian distribution. Three distribution cases were used to figure out the problem, from each of which 200 samples were generated and used to draw the scatters. Also, the contours on the likelihood of the two classes in each case and the posterior probability of one of the classes were plotted. The figures are as below:

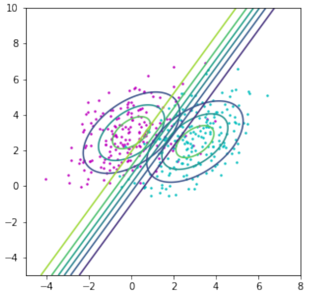


Figure 1: Scatter and Contours of the First Example

In the first case, samples came from two classes with distinct means $m_1 = \begin{pmatrix} 0 \\ 3 \end{pmatrix}$ and $m_2 = \begin{pmatrix} 3 \\ 2.5 \end{pmatrix}$, identical covariance matrices $C = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$ and equivalent prior probability of 0.5.

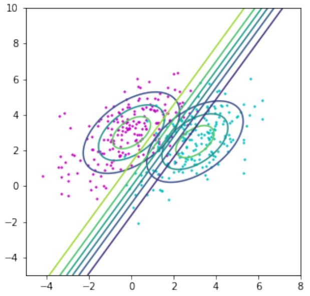


Figure 2: Scatter and Contours of the Second Example

In the second case, samples came from two classes with the same distinct means and identical covariance matrices as the first case, but the different prior probabilities($P_1 = 0.7$, $P_2 = 0.3$). It can be observed that the shape of the class boundaries did not change, but translated right along the x axis.

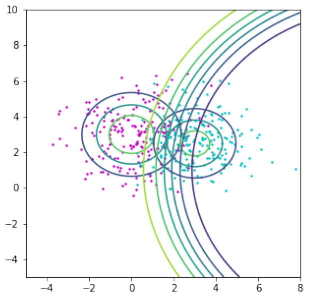


Figure 3: Scatter and Contours of the Third Example

In the third case, samples came from two classes with the same distinct means but the different covariance matrices as before ($C_1 = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$ and $C_2 = \begin{pmatrix} 1.5 & 0 \\ 0 & 1.5 \end{pmatrix}$). It can be observed that the shape of the class boundaries changed to curves. This result is as the expectation. To make a formal explaination, the function of class boundary $P(C_1|x)$ should be worked out. For Bayes Classifier:

$$P(C_1|x) = \frac{P(x|C_1)P(C_1)}{P(x|C_1)P(C_1) + P(x|C_2)P(C_2)} = \frac{1}{1 + \frac{P(x|C_2)P(C_2)}{P(x|C_1)P(C_1)}} \tag{1}$$

Suppose samples from the two classes are of Gaussian Distributions, there are

$$P(x|C_1) \sim \mathcal{N}(\mu_1, \Sigma_1), P(x|C_2) \sim \mathcal{N}(\mu_2, \Sigma_2) \tag{2}$$

Then,

$$P(C_1|x) = \frac{1}{1 + \frac{P(x|C_2)P(C_2)}{P(x|C_1)P(C_1)}} = \frac{1}{1 + \frac{P(C_2)}{P(C_1)} * \frac{P(x|C_2)}{P(x|C_1)}} \tag{3}$$

For $P_1 = P_2$, $\Sigma_1 = \Sigma_2 = \Sigma$:

$$P(C_1|x) = \frac{1}{1 + \frac{P(C_2)}{P(C_1)} * \frac{P(x|C_2)}{P(x|C_1)}} = \frac{1}{1 + e^{(x^T \Sigma^{-1} x - 2\mu_2^T \Sigma^{-1} x + \mu_2^T \Sigma^{-1} \mu_2) - (x^T \Sigma^{-1} x - 2\mu_1^T \Sigma^{-1} x + \mu_1^T \Sigma^{-1} \mu_1)}} \tag{4}$$

$$P(C_1|x) = \frac{1}{1 + e^{-w^T x + b}} \tag{5}$$

For $P_1 \neq P_2$, $\Sigma_1 = \Sigma_2 = \Sigma$:

$$P(C_1|x) = \frac{1}{1 + \frac{P(C_2)}{P(C_1)} * \frac{P(x|C_2)}{P(x|C_1)}} = \frac{1}{1 + \alpha e^{-w^T x + b}} \tag{6}$$

For $P_1 \neq P_2$, $\Sigma_1 \neq \Sigma_2$:

$$P(C_1|x) = \frac{1}{1 + \frac{P(C_2)}{P(C_1)} * \frac{P(x|C_2)}{P(x|C_1)}} = \frac{1}{1 + e^{(x^T \Sigma_2^{-1} x - 2\mu_2^T \Sigma_2^{-1} x + \mu_2^T \Sigma_2^{-1} \mu_2) - (x^T \Sigma_1^{-1} x - 2\mu_1^T \Sigma_1^{-1} x + \mu_1^T \Sigma_1^{-1} \mu_1)}} \tag{7}$$

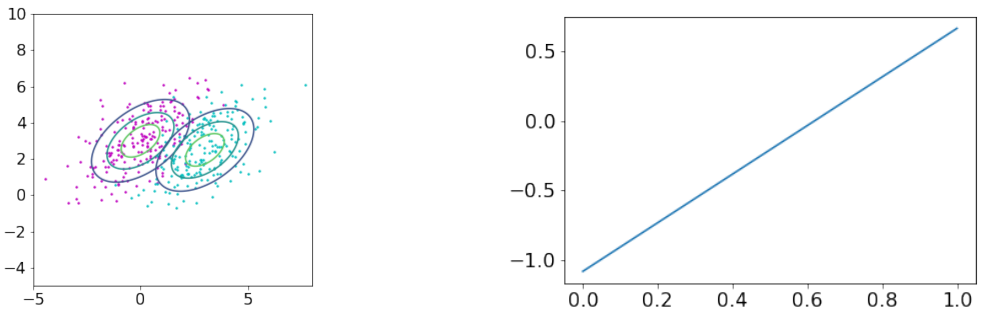
$$P(C_1|x) = \frac{1}{1 + \alpha e^{x^T A x + b x + c}} \tag{8}$$

From the perspective of mathematical inference, it is indicated that for the circumstance of classifying two classes with identical covariance matrices, the index of e is linear and the class boundaries are lines; once changing the prior probability of the two classes and make them different, the boundaries are still lines but will translate along the axis according to the value of bias. For the circumstance of classifying two classes with different covariance, the index of e contains quadratic term of x matrices and the boundaries of the classes become curves.

In conclusion, the result of the experiment is consistent with the analytical mathematics derivation.

2 Fisher Linear Discriminant Algorithm and ROC Curve

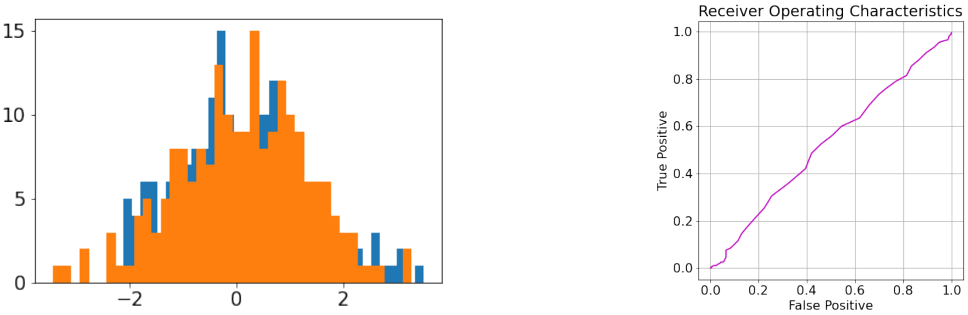
In this part, Fisher Linear Discriminant Algorithm was used in the same two-class classification problem. The scatter of the samples, the contours of the distributions and discriminant direction, and the histogram of the projections' distribution on the direction were drawn. Also, the true positive rates and false positive rates corresponding to every thresholds set to the classification algorithm were worked out and plotted to the Receiver Operating Characteristics Curve.The figures are as below:



(a) Samples of the Classification Example

(b) Discriminant Direction

Figure 4: Scatter and Contours of the Classification Example and Discriminant Direction



(a) Projections' Distribution

(b) ROC Curve

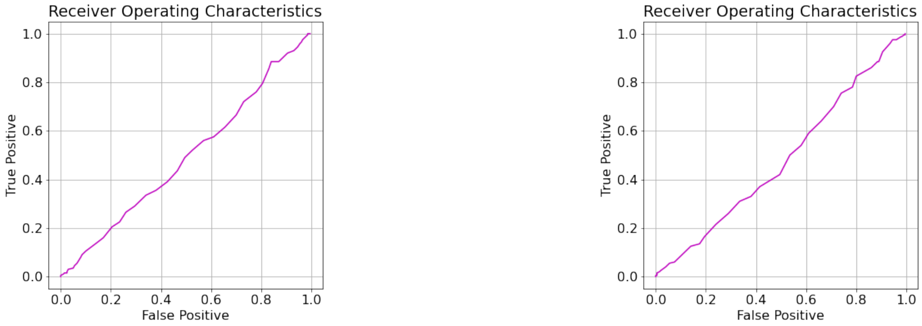
Figure 5: Histogram of Projections' Distribution and ROC Curve for Fisher LDA

In this case, the discriminant direction was defined by the means and the covariance matrices: $w_F = (C_1 + C_2)^{-1}(m_1 - m_2)$. It is observed that the True Positive Rate (TPR) is positively correlated to the False Positive Rate (FPR) as the threshold changes. Therefore, there need to be a suitable decision threshold to make the algorithm performance best.

Here, the theory of Youden Index was illustrated to measure the performance of the algorithm. Youden Index is the sum of Sensitivity and Specificity minus 1[1]. Sensitivity represents the TPR and the Specificity represents the opposite value of the FPR. According to the Youden Index, which indicates the algorithm capability of detecting both True Samples and False Samples, all the thresholds were iterated through again to seek the best Youden Index. As a result, the best threshold of the case above is -1.39, corresponding to a TPR of 0.875, an FPR of 0.825 and a best accuracy of 0.563.

Comparatively, two other cases in which the same samples were projected on different directions were proposed. In the first case, samples were projected to a random direction of vector $u = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$; in the second case, samples were projected onto the direction connecting the means of the two classes above and the direction could be expressed as $uF = m_2$ - m_1. The ROC curves of the two cases are plotted as below:

In the three cases above, the Area Under ROC Curve (AUC) was computed through function *numpy.trapz*(ROC[:,1], ROC[:,0]) and the corresponding value of them were 0.525, 0.486 and 0.478.



(a) Random Direction

(b) Means-connecting Direction

Figure 6: ROC Curves for Random Direction and Means-connecting Direction

The AUC value is the area between the ROC Curve and the X axis, representing the classification performance of the algorithm. Specifically, suppose picking up a positive sample and a negative sample from all and if the AUC value of the classifier is 0.5, the probabilities of the classifier judging both the two samples positive is identically 0.5. Similarly, if the AUC value is bigger or smaller, the classifier is more or less likely to judge positive samples positive than negative samples. To conclude, the AUC reflects the difference between probabilities.

Among the cases above, to find the best projecting direction to classify given samples, the Fisher Linear Discriminant Direction got an AUC value of 0.525 and performed better than random direction and the means-connecting direction, indicating the effectiveness of the Fisher Linear Discriminant Algorithm.

3 Mahalanobis Distance

In this part, two-class classification problems were purposed to identify the difference between a distance-to-mean classifier and a Mahalanobis distance-to-mean classifier. To accomplish the classification algorithm, functions from *scipy* Library was imported to calculate the two distance as $from$ *scipy.spatial.distance import pdist*. With the functions in the Library, the distances of the samples to the means were worked out and used to make a classification.

In the first example, 200 samples in each class were generated from two Gaussian Distributions with means $m_1 = \begin{pmatrix} 0 \\ 3 \end{pmatrix}$ and $m_2 = \begin{pmatrix} 3 \\ 2.5 \end{pmatrix}$, identical covariance matrices $C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$. Using the distances from samples to means to classify the samples into the nearest class, the Euclidean Distance-based classifier got an accuracy of 0.92 and the Mahalanobis Distance-based classifier also got an accuracy of 0.92, indicating that the Euclidean Distance equals the Mahalanobis Distance under this circumstance.

In the second example, samples were generated from two Gaussian Distributions with means $m_1 = \begin{pmatrix} 0 \\ 3 \end{pmatrix}$ and $m_2 = \begin{pmatrix} 3 \\ 2.5 \end{pmatrix}$, identical covariance matrices $C = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$. Using the same classifiers, the Euclidean Distance-based classifier got an accuracy of 0.875 and the Mahalanobis Distance-based classifier got an accuracy of 0.89, indicating that the Mahalanobis Distance is more accurate here.

From the perspective of mathematical analysis, the difference between the Euclidean Distance and the Mahalanobis Distance is that the latter considers the correlations of variables and multiply the distance with the inverse of the covariance matrix. Therefore, the Euclidean Distance-based classifier and the Mahalanobis Distance-based classifier showed no visible difference when the covariance matrix is unit matrix. However, when the covariance matrix becomes flexible, the Mahalanobis Distance-based classifier is more likely to be accurate to identify the samples from two classes as it reduces the influence of the large data and the correlations of variables.

References

[1] Fluss, R., Faraggi, D. and Reiser, B. (2005), Estimation of the Youden Index and its Associated Cutoff Point. Biom. J., 47: 458-472. https://doi.org/10.1002/bimj.200410135