ML Assignment1 Q1: Fisher's LDA

2016A7PS0049H Nikhil Nair
2016B3A70652H Ramnath Kumar
2016B4A70649H Shreyas Kulkarni

This document serves to summarise the findings from the linear discriminant analysis that was conducted on the two given datasets (a1_d1: dataset of 1000 points in a 2D space, a1_d2: dataset of 1000 points in a 3D space; both labelled 0/1).

Data points were split 0.8/0.2 to serve as training and testing sets.

While the Fisher's linear discriminant seeks to only establish an implementation of an optimal projection vector, this exercise has gone a further step to build a classifier using the priors and gaussian distributions corresponding to the training sets. This exercise was carried out to appreciate the simplicity of being able to design probabilistic classification models given priors and probability distributions, which comes as a result of Fisher's assumption - that the distributions on the projection vector are to be considered as normal distributions corresponding to the individual labels.

2-Dimensional Points Dataset

Throughout both implementations (2D, 3D), owing to the binary nature of target labelling, the $\mathbf{w} \propto \mathbf{S_W}^{-1}(\mathbf{m2} - \mathbf{m1})$ method was adopted as opposed to the multiclass generalisation.

Implementing Fisher's LDA for 2-dimensional points

Raw data points are projected onto the w_hat vector. Each labelled projection falls into either of two labels, 0/1, which according to Fisher's hypothesis can be generalised as individual gaussian distributions, and their intersection the discriminant of the two classes.

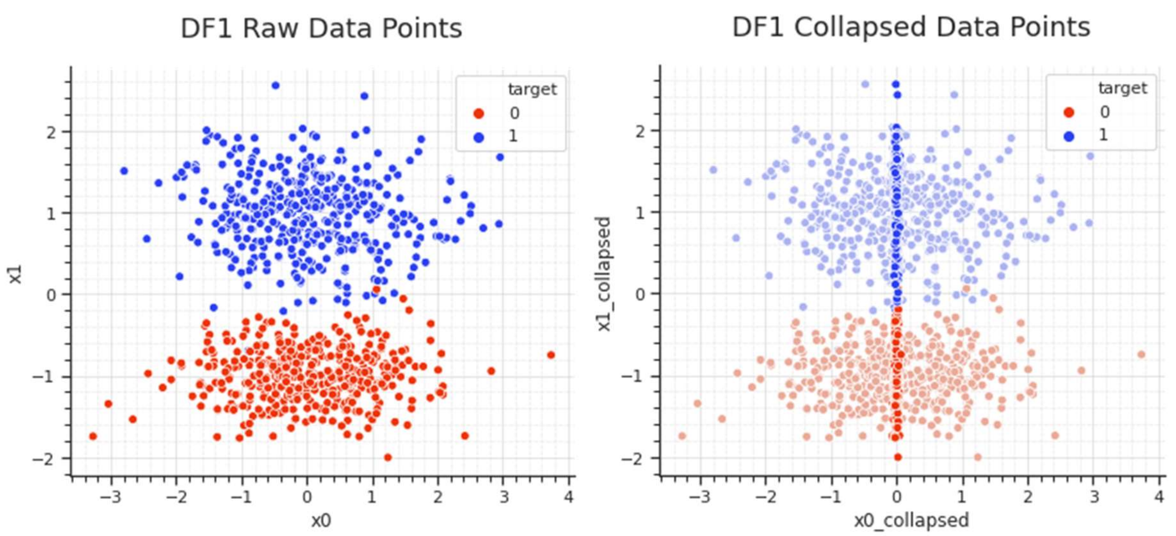

Fig 1: 2D Training Points – before/after projecting onto w_hat

Instead of working with joint gaussian distributions, this implementation went on to plot them component-wise, i.e. a plot each for x_0 and x_1, and then consider their individual intersections in the interclass plots as **candidates** for the discriminant point.

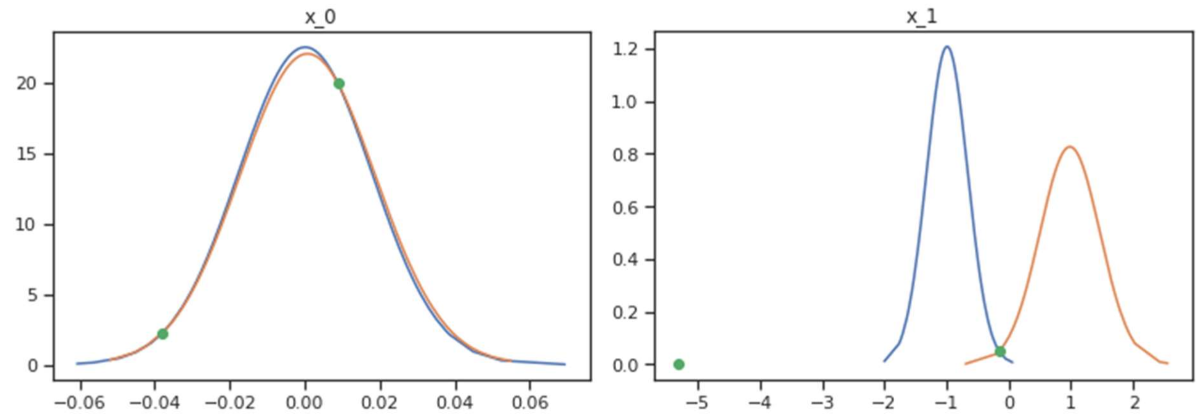

Fig 2: Component-wise (x0, x1) gaussian intersections to yield discriminant point on w_hat

Typically, any pair of gaussian distributions will can two intersections, like depicted in the x_0 plot in Fig2. And if the means were to be sufficiently separated from each other, like depicted in the x_1 plot in Fig2, then the next point of intersection could be as far away as wherever the asymptotes of the two plots meet. This is what essentially leads to possible candidate solutions.

To eliminate the inapplicable candidates, each of the candidate sets were marked up against the w_hat vector. Since only one of such candidates will lie on the w_hat vector, that must be the implementation's discriminant point. For a specific 0.8/0.2 (not deterministic due to randomized splitting/mixing) with the w_hat vector **[0.01852249 0.99982844]^T**, the discriminant came out to be **(0.00016308, -0.15865644)** *(ordered x_0, x_1)*.

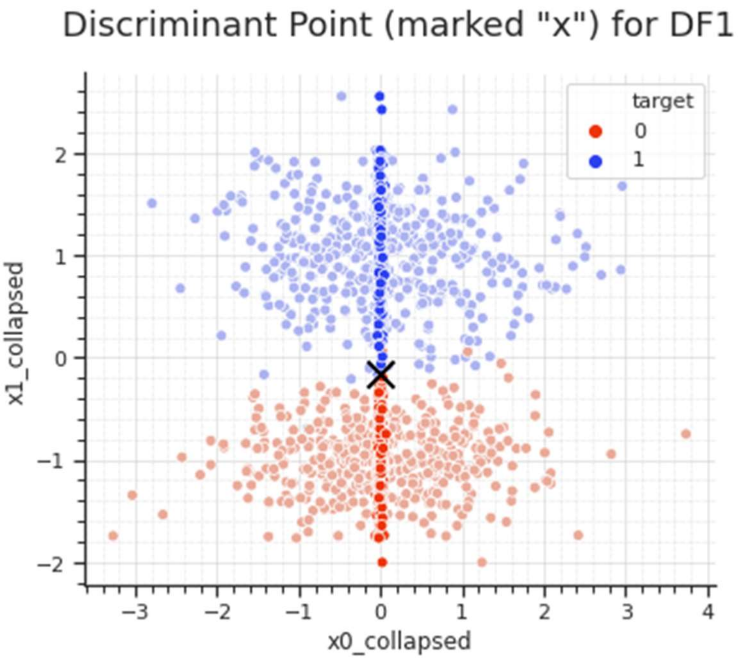

Fig 3: Plotting discriminant point on w_hat

Training against Implementation

After plotting the discriminant, sample set used to determine the w_hat vector and discriminant point was evaluated against the implementation.

True Negatives: 396	False Positives: 2
False Negatives: 3	True Positives: 399

Table 1: Confusion matrix for the 2D points dataset (training)

Using the above values, the following performance metrics were concluded upon. This particular implementation seemingly performs extraordinarily owing to the nature of points in the dataset, i.e. a discernible inter-class segregation obvious simply on immediate observation, exactly the kind of data Fisher's LDA favours.

Performance Metric	Value
Accuracy	99.38
Precision	99.50
Recall	99.25
F-Score	99.38

Table 2: Performance – 2D training points against implementation

Testing against LDA Classifier

As was discussed earlier, 0.2 split from the original data was set aside to evaluate against a probabilistic classifier. This classifier is based on the training sample's prior probabilities and the gaussian distributions discovered from Fisher's LDA.

True Negatives: 102	False Positives: 0
False Negatives: 1	True Positives: 97

Table 3: Confusion matrix for the 2D points dataset (testing)

Performance Metric*	Value
Accuracy	99.50
Precision	100.00
Recall	98.98
F-Score	99.49

Table 4: Performance - 2D testing points against the LDA classifier (see Appendix)*

3-Dimensional Points Dataset

Parallel to the 2-dimensional binary classification implementation, this report also encapsulates the findings from a 3-dimensional binary labelled dataset as well.

Implementing Fisher's LDA for 3-dimensional points

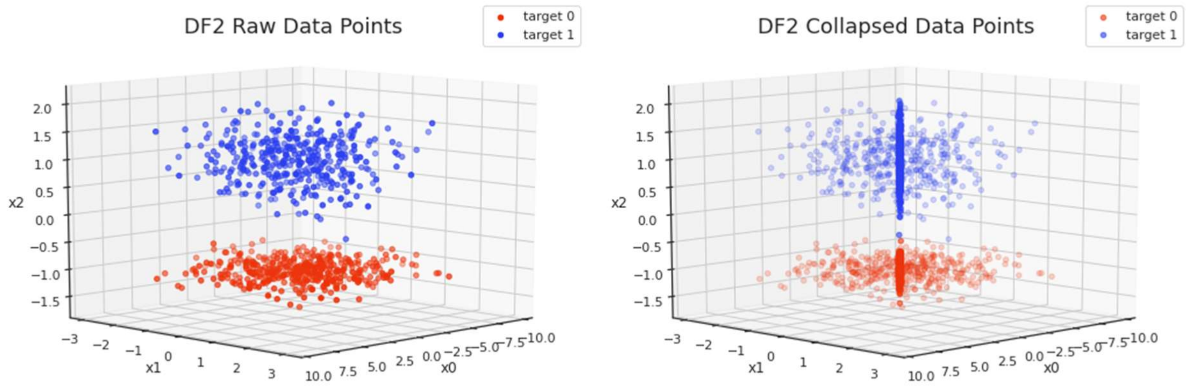

Fig 4: 3D Training Points – before/after projecting onto w_hat

As was carried out with the 2D dataset, the discriminant is calculated by considering the possible candidate solutions from the following component-wise gaussian intersections.

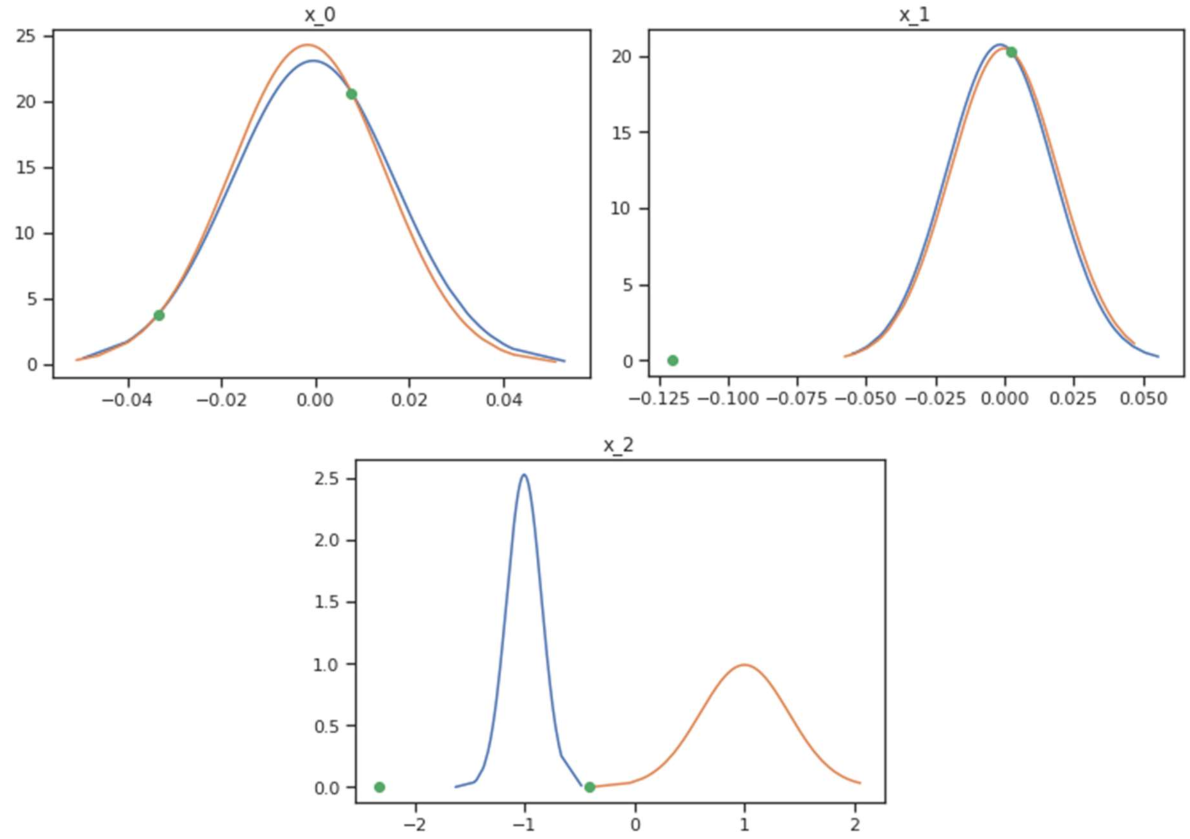

Fig 5: Component-wise (x0, x1, x2) gaussian intersections to yield discriminant point on w_hat

For a specific 0.8/0.2 (not deterministic due to randomized splitting/mixing) with the w_hat vector **[-0.00560076 -0.01886984 0.99980626]^T**, the discriminant came out to be **(0.00018757, 0.00226881, -0.40793212)** *(ordered x_0, x_1, x_2)*.

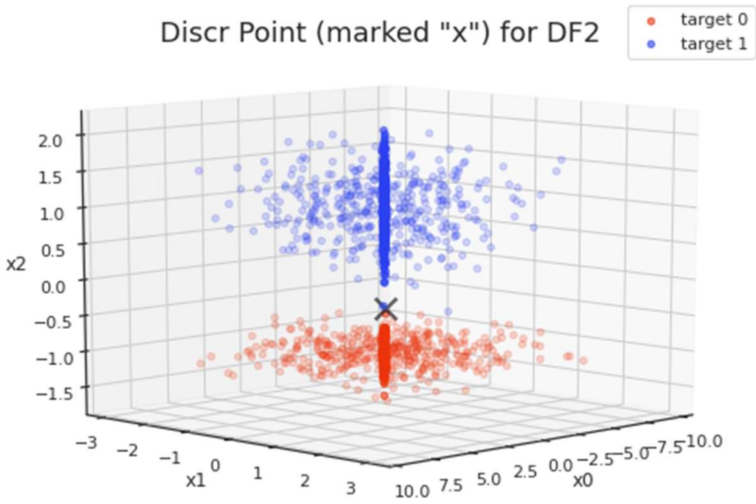

Fig 6: Plotting discriminant point on w_hat

Training against Implementation

Analogous to the 2D implementation, the 3D dataset was also split into a training sample and a test set. The current and following sections tally up the confusion matrices and performance metrics for the training sample against the LDA implementation and the test set with the LDA inspired probabilistic classifier.

True Negatives: 400	False Positives: 0
False Negatives: 0	True Positives: 400

Table 5: Confusion matrix for the 3D points dataset (training)

Again, as was discussed with the 2D models, the apparent brilliant performance discernible in these two sections is purely attributed to the nature of the separability of the data points.

Performance Metric	**Value**
Accuracy	100.00
Precision	100.00
Recall	100.00
F-Score	100.00

Table 6: Performance - 3D training points against implementation

Testing against LDA Classifier

True Negatives: 100	False Positives: 0
False Negatives: 0	True Positives: 100

Table 7: Confusion matrix for the 3D points dataset (testing)

Performance Metric	Value
Accuracy	100.00
Precision	100.00
Recall	100.00
F-Score	100.00

Table 8: Performance - 3D testing points against the LDA classifier

Appendix

The accuracy of any exercise or observation is simply the number of correct observations (in this case classifications) vs the total number of observations. In addition to accuracy, the other performance metrics discussed throughout the report have been formulated briefly as follows.

F-score, or F1-score as it is popularly known, follows

$$F1 = 2 \times \frac{Precision * Recall}{Precision + Recall}$$

where precision and recall follow the following formulae.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$= \frac{True\ Positive}{Total\ Predicted\ Positive}$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

$$= \frac{True\ Positive}{Total\ Actual\ Positive}$$