EE 565 - Machine Learning I
Project 2 Report

Jorge A. Garcia

***Abstract*—Classification of four different data sets is done by implementing various linear models. The performance of Least Squares Classification, Maximum Mean Projection and Fisher's Discriminant on the data sets are compared.**

***Index Terms*—Machine Learning, Data Science**

I. Introduction

Looking at the problem of classification, one common approach in dividing data distributions of D dimensions is by separation through D-1 dimensional hyperplanes. This is the task that Least Squares Classification performs, by finding the plane that minimizes the square error of the data to their respective classes. Another approach is to think of linear classification as a dimensionality-reduction task, performed by algorithms such as Maximum Mean Projection and Fisher's Discriminant. It is possible to, for example, map a 2-D data set into 1-D and find an optimal plane that separates the distribution groups within a data set, albeit with a loss in information from the reduced dimension.

In this project, we implement three different linear models to determine the optimal planes that divide four different data sets into their respective classes. The results obtained from Least Squares Classification, Maximum Mean Projection and Fisher's Discriminant are compared and discussed.

II. Problem 1: Least Squares Classification (Two Classes)

For this section, a class *LeastSquares_Classifier* is implemented and used on all of the data sets. A separate function *oneHotEncode* is used to map the classes in a one-hot format.

A. Classifying "Dataset A"

The resulting decision boundary and regions after training for this data set can be seen in Figure 1. For this particular data set, the training data is fitted perfectly and achieves 100% accuracy. Both groups are distributed similarly, with no noticeable outliers; this allows the Least Squares classifier to converge properly. The algorithm perfectly captured the distribution in this data set.

B. Classifying "Dataset A2"

The resulting decision boundary and regions for this data set are shown in Figure 2. In this data set, the algorithm has trouble effectively capturing the distribution due to the outliers. Minimization of the error from these outliers is weighted heavier than the error due to misclassified points. The resulting accuracy for the model on this data set is of 96.25%. It is not a bad fit, but perfect classification of the training set could be obtained with other algorithms.

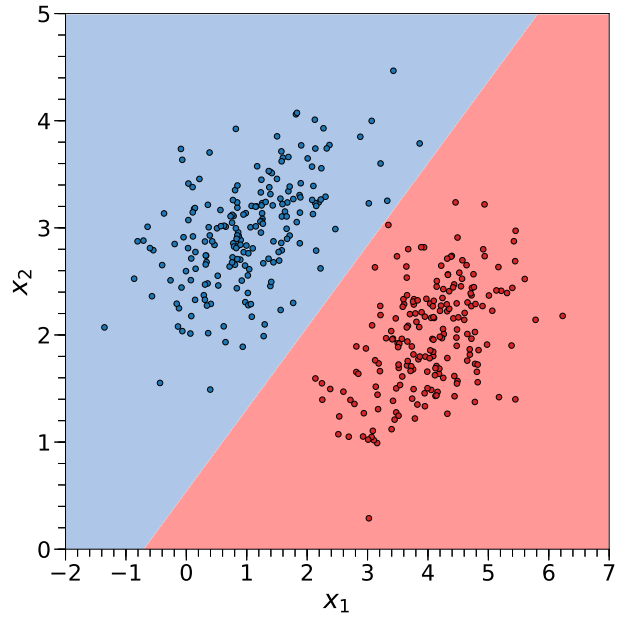

Fig. 1. Decision region for the classes in "DatasetA" using a Least Squares Classifier.

C. Classifying double moon distribution

The resulting decision boundary and regions for this data set are shown in Figure 3. Since this is not a linearly separable problem, it is impossible to obtain a perfect model with linear Least Squares Classification. With an accuracy of 95.82%, it is fairly accurate but other algorithms or different paramaterization of the data could result in better models.

III. Problem 2: Least Squares Classification (Three Classes)

For this section, the previously used *Least-Squares_Classifier* was used alongside the *oneHotEncode* function to map the outputs.

A. Classifying "Dataset B"

The resulting decision boundary and regions for this data set are shown in Figure 4. The model converges well and is able

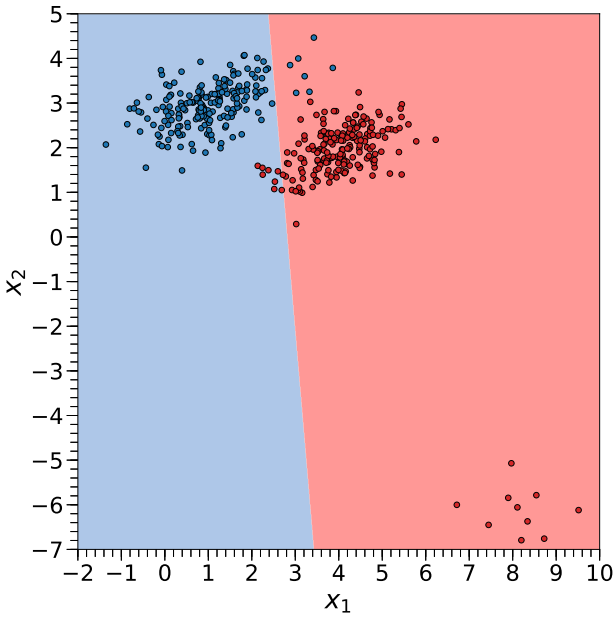

Fig. 2. Decision region for the classes in "DatasetA2" using a Least Squares Classifier.

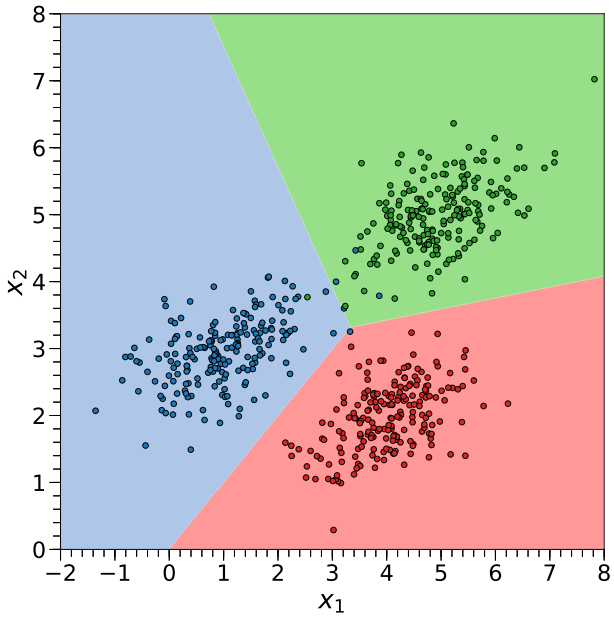

Fig. 4. Decision region for three classes in "DatasetB" using a Least Squares Classifier.

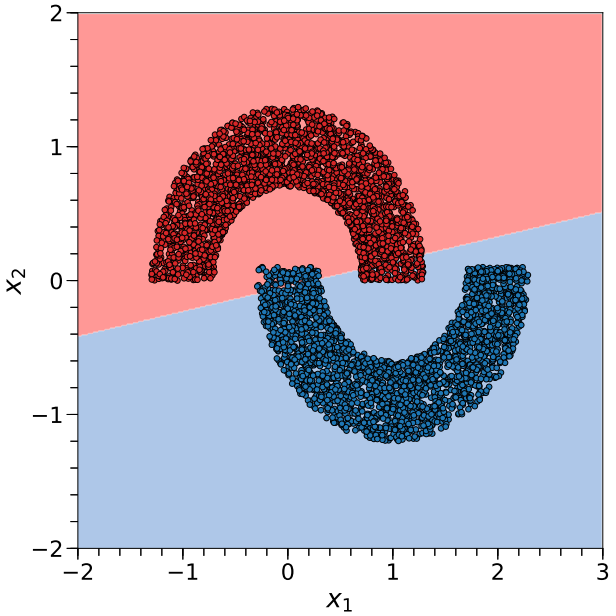

Fig. 3. Decision region for the classes in the Double Moon data set using a Least Squares Classifier.

to effectively separate the three distributions, with an accuracy of 99% for predicting the training data. It effectively captures the regions of the three classes, and its near-perfect accuracy make it a very good model.

IV. Maximum Mean Projection Classification

For this section, a class *MaxMeanProjection* is implemented and used on three of the data sets. Confusion arose from determining the decision threshold, as the function given in the prompt

$$y_0 = \text{mean}(\boldsymbol{X}\boldsymbol{w}) - w_1 \tag{1}$$

did not give the best possible results for the algorithm. The added term $-w_1$ produces an arbitrary shift in the decision region that biases the models towards a certain class. A more optimal decision boundary would be

$$y_0 = \text{mean}(\boldsymbol{X}\boldsymbol{w}) \tag{2}$$

which is approximately the mid-point between the two distributions. This decision threshold results in better predictions, as seen in the following results.

A. Classifying "Dataset A"

The resulting decision region when using Equation 1 as a decision threshold for this data set is shown in Figure 5. The added term biases the model to predict more instances of the red class, resulting in an accuracy of 91% for the training set. Also seen in Figure 5 is the histogram due to the 1-D projection of the data set. Notice how the decision threshold lies well within the blue class.

Figure 6 shows the decision region when using Equation 2 as the decision threshold, and the histogram of the 1-D projection. It is a more fair split, resulting in an accuracy of 98.25% percent for the training data. It performs worse compared to the Least Squares classifier in Section II-A. The loss of information due to the 1-D projection results in data points between both classes overlapping, giving a lower accuracy in what is a data set that is linearly separable in a perfect manner.

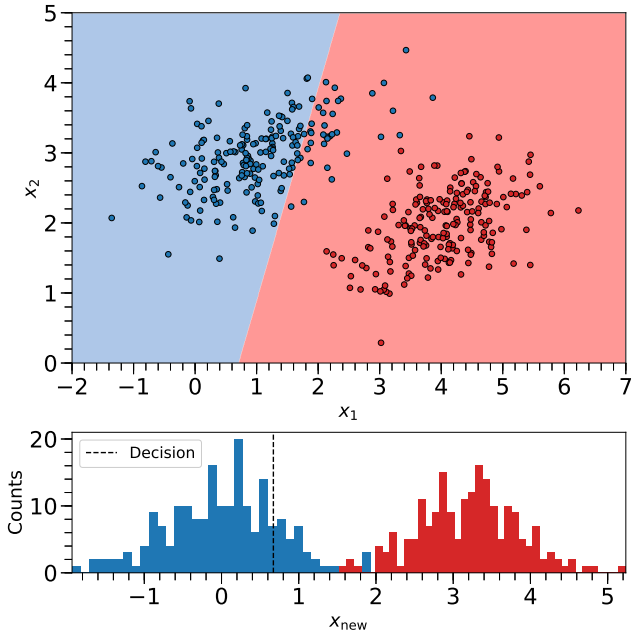

Fig. 5. Decision region for the classes in "DatasetA" using Maximum Mean Projection and a decision boundary $y_0 = \text{mean}(\boldsymbol{X}\boldsymbol{w}) - w_1$.

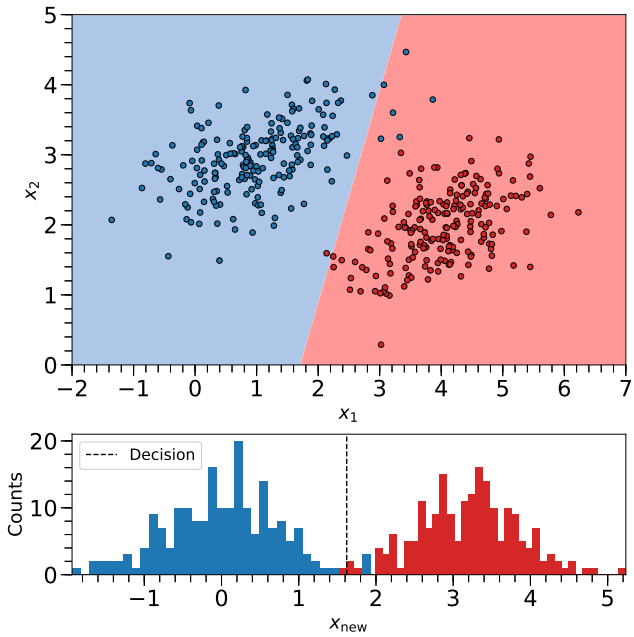

Fig. 6. Decision region for the classes in "DatasetA" using Maximum Mean Projection and a decision boundary $y_0 = \text{mean}(\boldsymbol{X}\boldsymbol{w})$.

B. Classifying "Dataset A2"

Shown in Figure 7 are the decision region map using Equation 1 as the decision threshold, and the 1-D projection histogram for this data set. As before, the decision region is biased towards predicting for the red class due to the added term. Using this decision threshold, the accuracy of the model is of 94.5% for the training data.

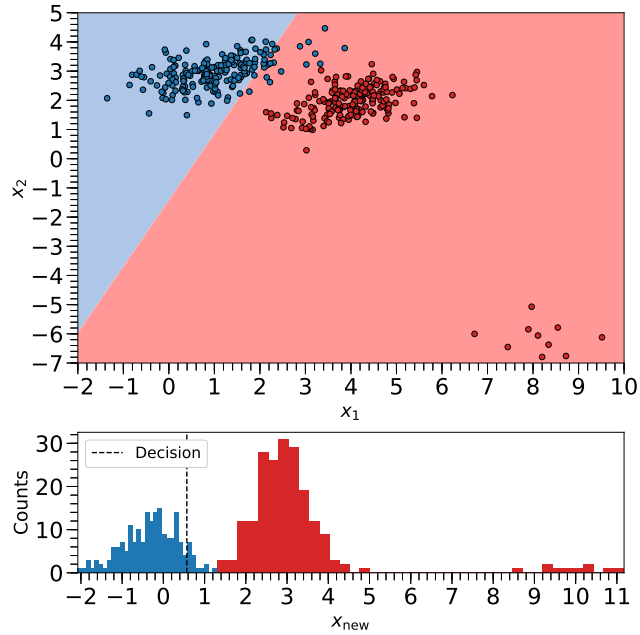

Fig. 7. Decision region for the classes in "DatasetA2" using Maximum Mean Projection and a decision boundary $y_0 = \text{mean}(\boldsymbol{X}\boldsymbol{w}) - w_1$.

Figure 8 shows the decision region map using Equation 2 as the decision threshold, and the corresponding 1-D projection histogram. The accuracy using this threshold is of 98.75% for the training data, which outperforms the Least Squares classifier on this data set, seen in Section II-B. Since both calculation of the weights to project the data and the decision threshold rely on the in-class average, the impact on the model of the outliers in the red class distribution is reduced compared to that in the Least Squares classifier.

C. Classifying double moon distribution

Figure 9 shows the decision region map using Equation 1 as the decision threshold, and the corresponding 1-D projection histogram for this data set. As in the previous two subsections, the decision threshold is shifted into the blue class distribution as seen in the histogram. This decision threshold results in a training accuracy of 82.46%.

Shown in Figure 10 is the decision region map using Equation 2 as the decision threshold, and the corresponding 1-D projection histogram. This decision threshold results in a symmetric decision region, which is what is expected due to the symmetric nature of the data. With a training accuracy of 86.52%, it is a worse model compared to Least Squares classification on the same data set, as seen in Section II-C. It is not a good algorithm for this data set.

V. Problem 4: Fisher's Discriminant

For this section, a class *FishersDiscriminant* is implemented and used on three of the data sets. As with the Maximum Mean Projection section, the issue regarding choice of decision threshold is present for this algorithm as well. Both Equations 1 and 2 will be considered for all data sets. Particularly, for

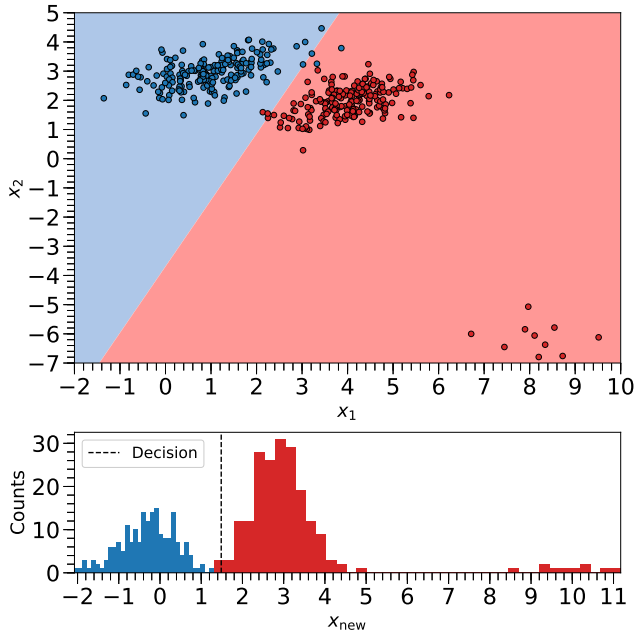

Fig. 8. Decision region for the classes in "DatasetA2" using Maximum Mean Projection and a decision boundary $y_0 = \text{mean}(\boldsymbol{X}\boldsymbol{w})$

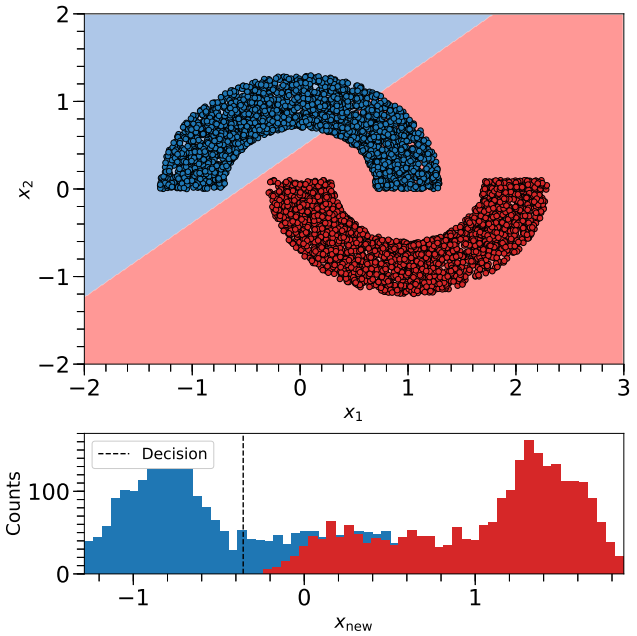

Fig. 9. Decision region for the classes in the Double Moon data set using Maximum Mean Projection and a decision boundary $y_0 = \text{mean}(\boldsymbol{X}\boldsymbol{w}) - w_1$.

the case of using Equation 2 as the decision threshold, it is shown that Fisher's Discriminant is a special case of the Least Squares classifier when considering a two-class problem [1].

A. Classifying "Dataset A"

Figure 11 shows the decision region using Equation 1 as the decision threshold, and the 1-D projection histogram for the data set. As in the previous section, the added term biases

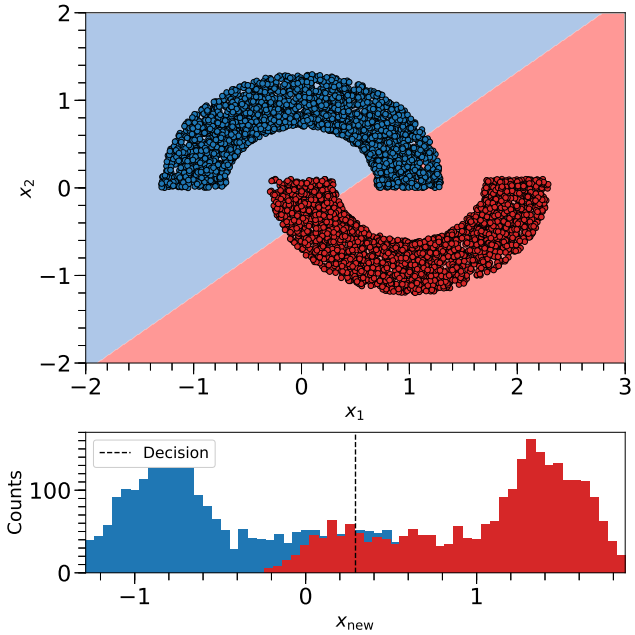

Fig. 10. Decision region for the classes in the Double Moon data set using Maximum Mean Projection and a decision boundary $y_0 = \text{mean}(\boldsymbol{X}\boldsymbol{w})$

the model towards predicting for the red class. The prediction accuracy using this threshold is 96.75%

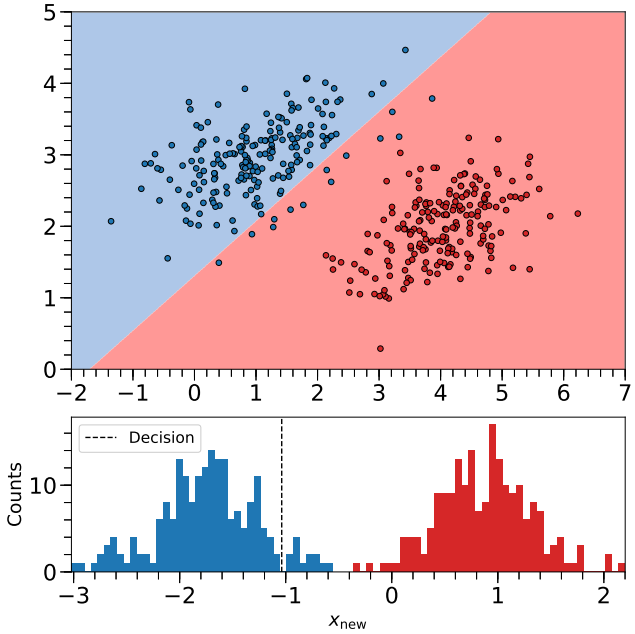

Fig. 11. Decision region for the classes in "DatasetA" using Fisher's Discriminant and a decision boundary $y_0 = \text{mean}(\boldsymbol{X}\boldsymbol{w}) - w_1$

The decision region when considering Equation 2 as the decision threshold can be seen in Figure 12, alongside the 1-D projection histogram. The achieved accuracy was 100%, an improvement from Maximum Mean Projection and performing as the Least Squares classifier did due to the chosen decision

threshold.

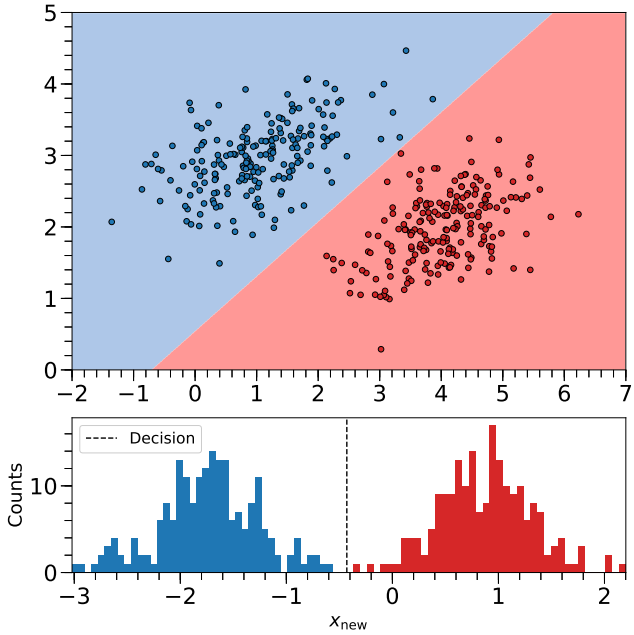

Fig. 12. Decision region for the classes in "DatasetA" using Fisher's Discriminant and a decision boundary $y_0 = \text{mean}(\boldsymbol{X}\boldsymbol{w})$

B. Classifying "Dataset A2"

Figure 13 shows the decision region using Equation 1 as the decision threshold, and the corresponding 1-D projection histogram for this data set. Due to the extra bias in the decision threshold, the prediction accuracy is of 86.25% for the training data.

The resultant decision region when using Equation 2 as a decision threshold is shown in Figure 14. This threshold results in an accuracy of 96.25% with the train data. Taking into account the in-class variance to determine the projection would create sensitivity to outliers such as in this data set, causing it to perform worse than Maximum Mean Projection. As mentioned before, this decision threshold choice results in the same performance as Least Squares.

C. Classifying double moon distribution

The decision region resulting from Equation 1 being used as a decision threshold is shown in Figure 15 alongisde the 1-D projection histogram. The added bias results in a training accuracy of 94.62%.

Figure 16 shows the decision region when using Equation 2 as the decision threshold. As stated before, it is not a linearly separable problem and thus cannot be perfectly fitted with a linear model. The resulting training accuracy is of 95.82%, which outperforms Maximum Mean Projection and is identical to to the Least Squares classifier.

VI. Conclusion

In this project, we implemented Least Squares Classification, Maximum Mean Projection and Fisher's Discriminant and tested their performance on four different data sets. The results obtained show how they may excel in generating useful boundaries to model the data, but also their potential pitfalls.

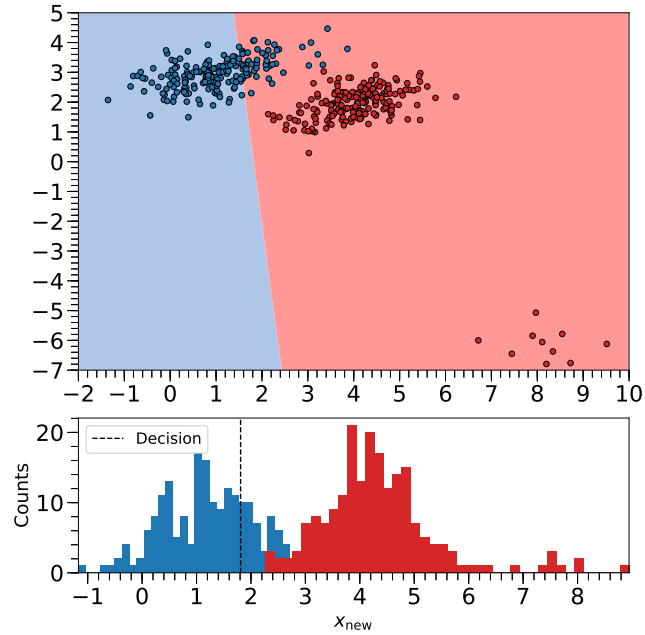

Fig. 13. Decision region for the classes in "DatasetA2" using Fisher's Discriminant and a decision boundary $y_0 = \text{mean}(\boldsymbol{X}\boldsymbol{w}) - w_1$

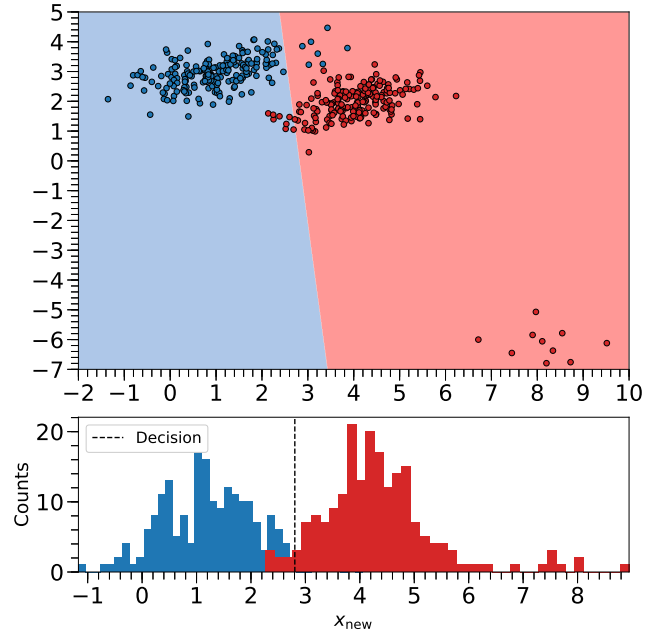

Fig. 14. Decision region for the classes in "DatasetA2" using Fisher's Discriminant and a decision boundary $y_0 = \text{mean}(\boldsymbol{X}\boldsymbol{w})$

References

[1] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

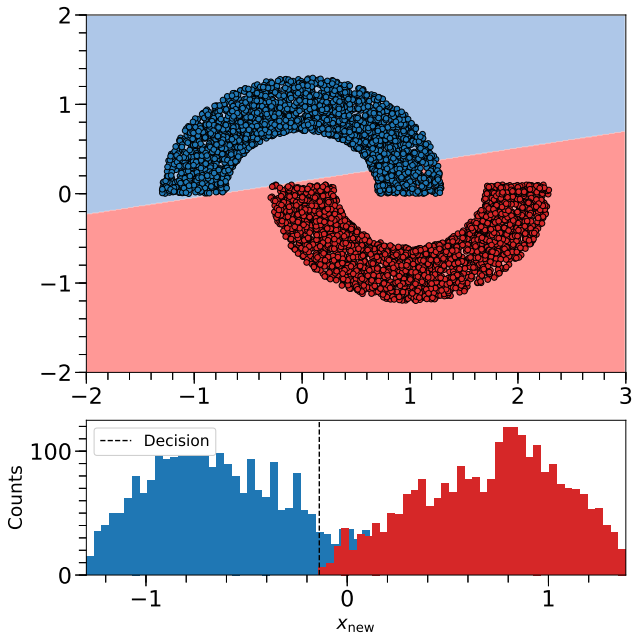

Fig. 15. Decision region for the classes in the Double Moon data set using Fisher's Discriminant and a decision boundary $y_0 = \text{mean}(\boldsymbol{X}\boldsymbol{w}) - w_1$

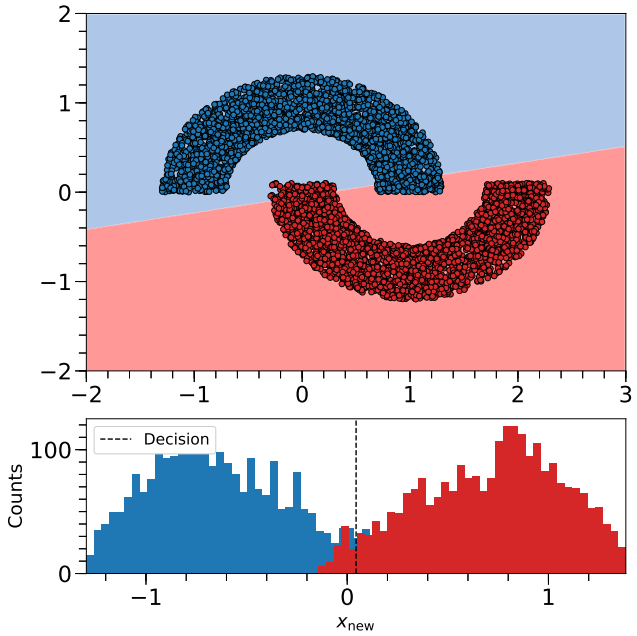

Fig. 16. Decision region for the classes in Double Moon data set using Fisher's Discriminant and a decision boundary $y_0 = \text{mean}(\boldsymbol{X}\boldsymbol{w})$