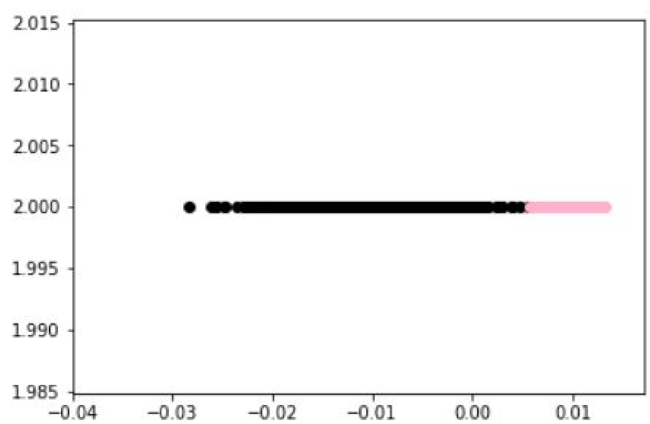
Machine Learning [BITS F464]
Assignment – 1 Report

Group Members :
R Shivani Reddy [2015B3A70531H]
Srujana Chegireddy [2015AAPS0273H]
Vijitha Gunta [2015B3A70491H]

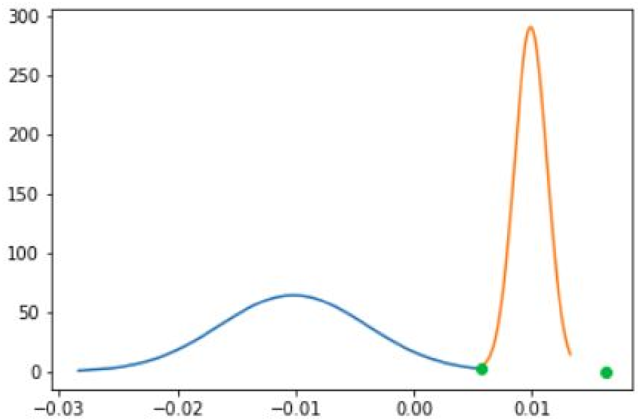
Question 1 – Fischers Linear Discriminant

As per Fischers linear discriminant algorithm, we project the points on weights vector which is proportional to Sw-1(m1-m2).

1. Observations for dataset_1

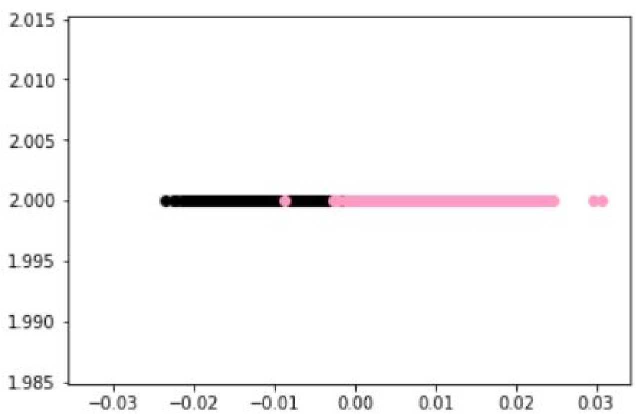


Above figure shows the scatter plot of the projected points in 1-dimensional plane for dataset1. Black points correspond to class 0 and pink points correspond to class1.

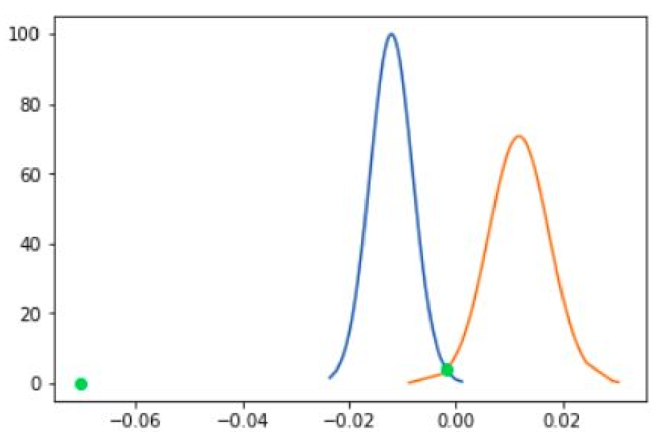


Above graph shows the normal curves for two classes and the intersection point. Blue curve belongs to class 0 and orange curve belongs to class 1 points. The intersection point of two curves gives the threshold to classify a given testing point into two classes. The threshold obtained for this dataset was 0.016199.

2. Observations for dataset_2

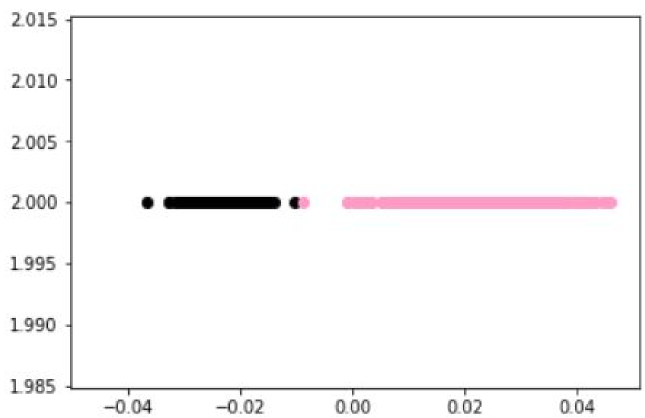


Above figure shows the scatter plot of the projected points in 1-dimensional plane for dataset1. Black points correspond to class 0 and pink points correspond to class1.

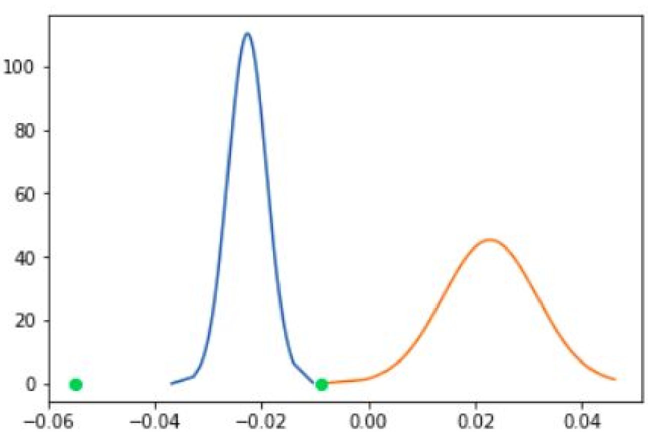


Above graph shows the normal curves for two classes and the intersection point. Blue curve belongs to class 0 and orange curve belongs to class 1 points. The intersection point of two curves gives the threshold to classify a given testing point into two classes. The threshold obtained for this dataset was -0.00183085.

3. Observations for dataset_3



Above figure shows the scatter plot of the projected points in 1-dimensional plane for dataset1. Black points correspond to class 0 and pink points correspond to class1.

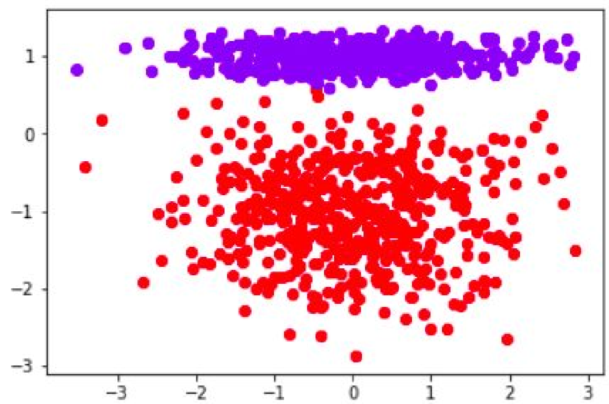


Above graph shows the normal curves for two classes and the intersection point. Blue curve belongs to class 0 and orange curve belongs to class 1 points. The intersection point of two curves gives the threshold to classify a given testing point into two classes. The threshold obtained for this dataset was -0.008855.

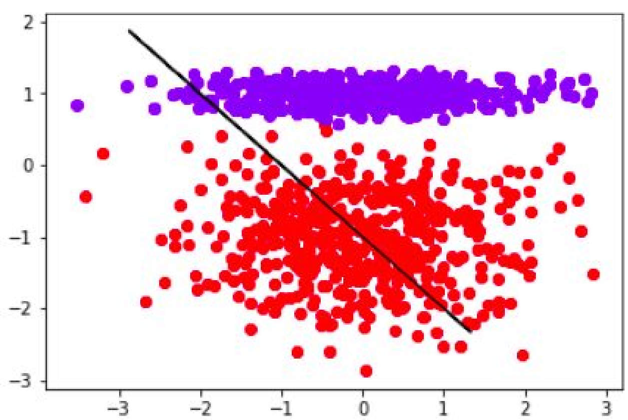
Question 2 – Perceptron

According to perceptron, we start with a particular set of weights and change it in small amounts in the opposite direction of first derivative of the error function. For gradient descent, we take the first misclassified point and alter the value of weights accordingly.

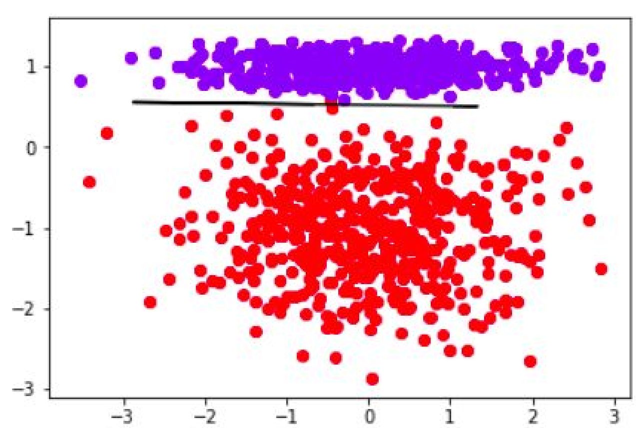
1. Observations for dataset_1



Above scatter plot shows the distribution of the dataset. Blue corresponds to class 1 and red points belong to class 0.

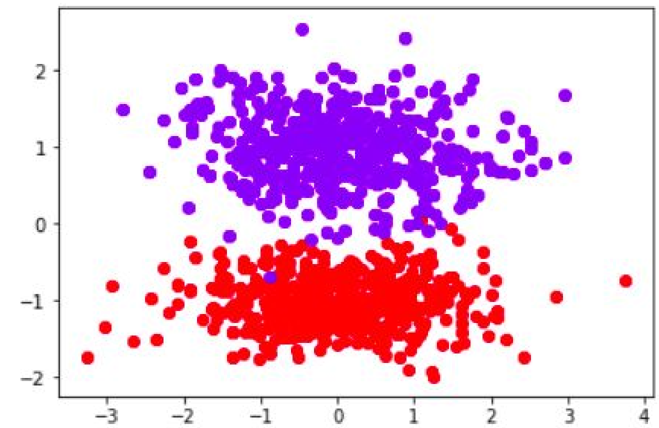


Initially we start with equal weights and apply gradient descent. The following figure shows the final plot after 8 epochs.

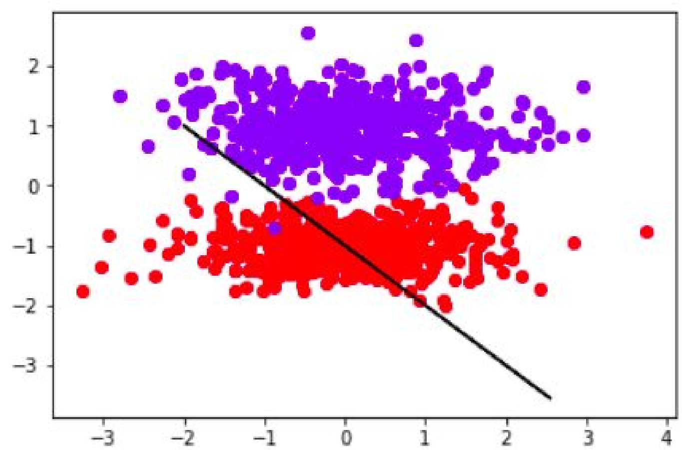


As the points are linearly separable, a perfect convergence is achieved for this dataset. The above black line is the best possible line of classification that can be achieved using gradient descent and perceptron.

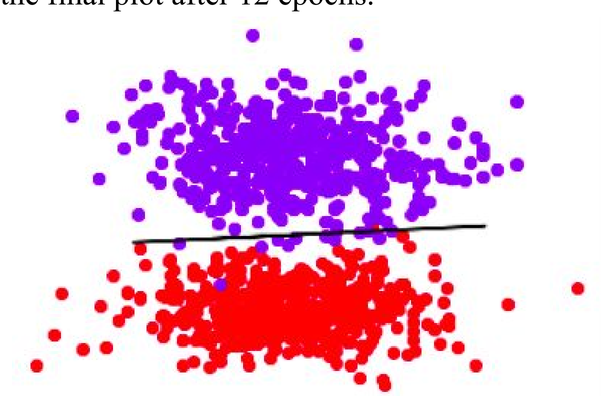
2. Observations for dataset_2



Above scatter plot shows the distribution of the dataset. Blue corresponds to class 1 and red points belong to class 0.

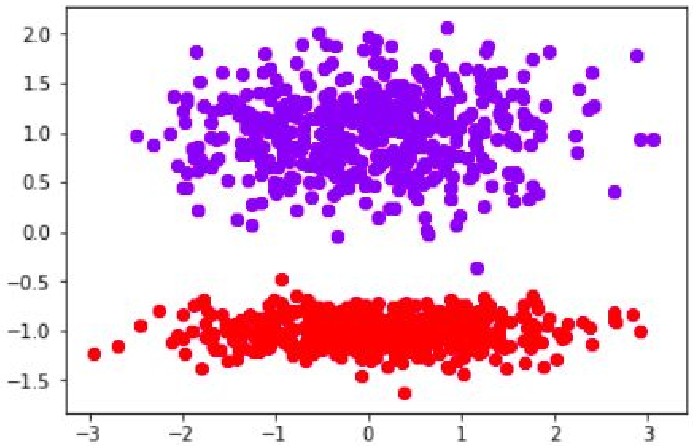


Initially we start with equal weights and apply gradient descent. The following figure shows the final plot after 12 epochs.

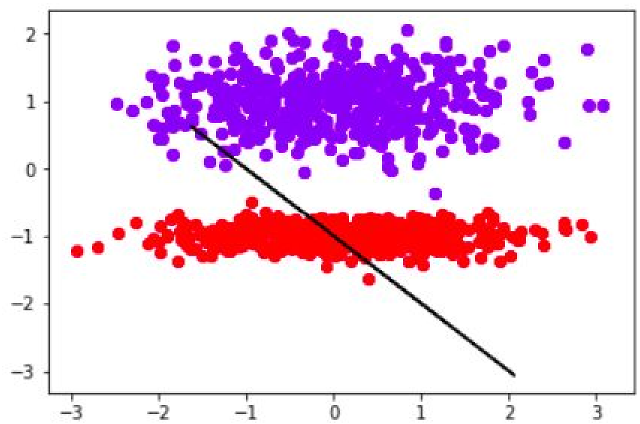


As the points are not linearly separable, a perfect convergence is not achieved for this dataset. The above black line is the best possible line of classification that can be achieved using gradient descent and perceptron.

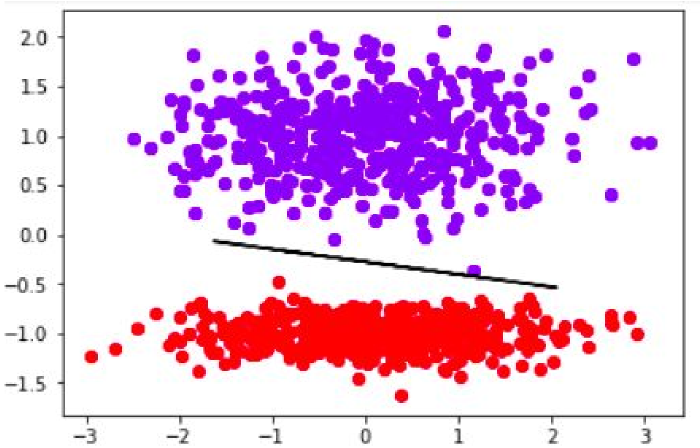
3. Observations for dataset_3



Above scatter plot shows the distribution of the dataset. Blue corresponds to class 1 and red points belong to class 0.



Initially we start with equal weights and apply gradient descent. The following figure shows the final plot after 7 epochs.



As the points are linearly separable, a perfect convergence is achieved for this dataset. The above black line is the best possible line of classification that can be achieved using gradient descent and perceptron.

Difference between datasets and the impact of parameter initialization and the training data ordering on the models:

1. Difference between datasets: From the perceptron results, it can be observed that the datasets 1 and 3 achieved perfect convergence and hence are linearly separable while the 2nd dataset is not. Simple deduction from the scatter plot also suggests the same.
2. Impact of parameter initialization: In perceptron modelling, initialization of the weights and biases has a drastic impact on the final results. Weights and biases are generally not preferred to be close to zero since the effect of learning rate can be observed only if the weights have significant values.
   Learning rate is also an important factor to achieve convergence. A learning parameter of 1 changes the weights at a higher rate for every modification in the gradient descent and might tend to miss reaching the convergence point based on the data. A better learning rate is 0.1 which slowly changes the weights and comes up with a better classification line.
3. Impact of training data ordering: If datasets are linearly separable, the data ordering does not matter. In other words, if they are linearly separable, upto a constant factor, the weights will be same regardless of the training data order. Otherwise, different training orders will lead to different weights.