Key Differences Between the Perceptron and MLP Architectures

A) <u>Architecture</u>

The perceptron has a single unit of a neuron that results in a single decision boundary. What we mean is that it maps the inputs to outputs by making use of linear transformations by a binary activation, like the heaviside activation function we used.

The multi layer perceptron consists of multiple units of neurons, 9 to be exact, into 1 hidden layer of 8 neurons and 1 output neuron. This enables the MLP to create non-linear transformations of the input, of course given a non-linear activation function, and find spaces that make them linearly separable which isn't really possible for the perceptron.

B) <u>Learning</u>

The perceptron uses a simple update rule based on misclassified values to adjust its weights and learn. The reason it is simple is because we only are concerned with a single neuron's weight and bias.

The MLP optimizes a continuous loss function, in our case a cross-entropy loss function, using backpropagation and gradient descent. This allows more efficient learning and better generalization.

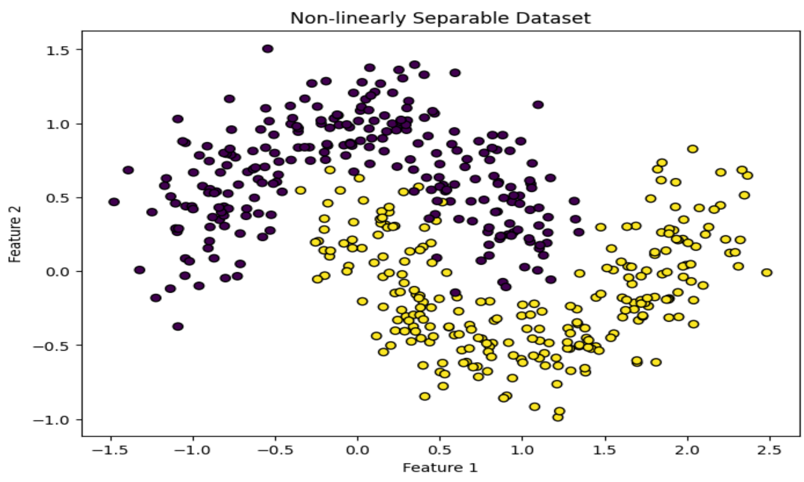
C) <u>Performance</u>

The performance of the perceptron is on par with MLP when the data is linearly separable. Since it is smaller it learns faster and hence it can be used instead of the MLP, because simpler models are better than complex ones if they have similar performance with regards to accuracy.

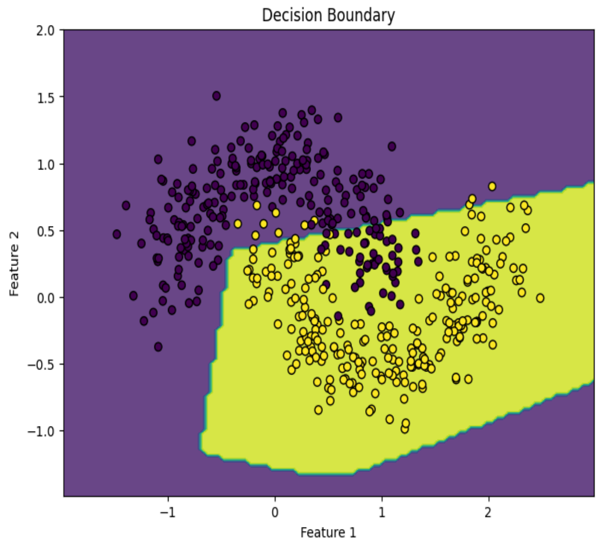
When it comes to non-linear data, the perceptron's shortcomings are exposed. Since it can't model complex decision boundaries it fails to beat the performance of the MLP. We will show evidence on the next topic.

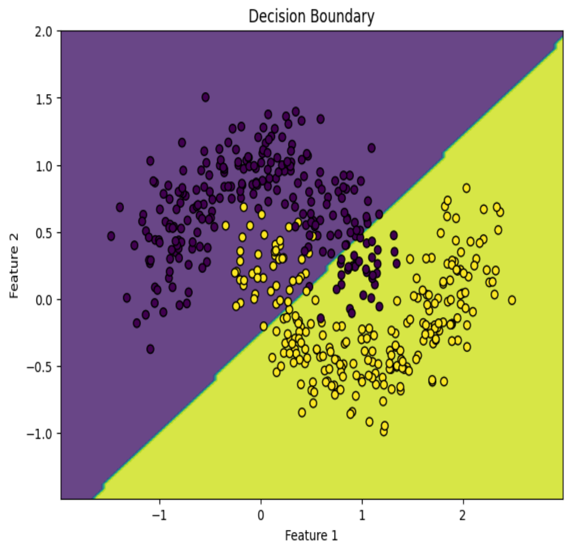
Visualizations

We used sklearn to create a non-linearly separable data set. This is the visualization of the data:

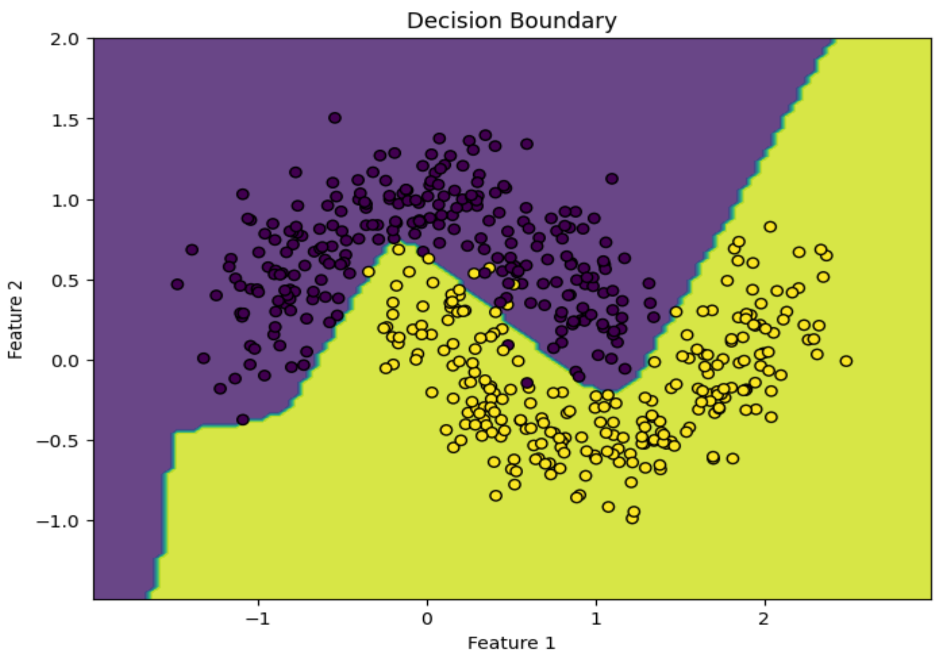


We then trained both the perceptron and MLP on the same training set and evaluated them on the same testing set. The results show the difference, the MLP had an accuracy of 91% while the perceptron had an accuracy of 86%. The decision boundaries of both are visualized below.





As you can see the decision boundary on the left, which is MLP's boundary, is more representative of the dataset while the perceptrons is linear and doesn't bisect the data training properly. Below there is even a better example of the MLP's boundary found from one of our experiments.



Here the boundary is even more stricter and creates a better decision line for the data set.