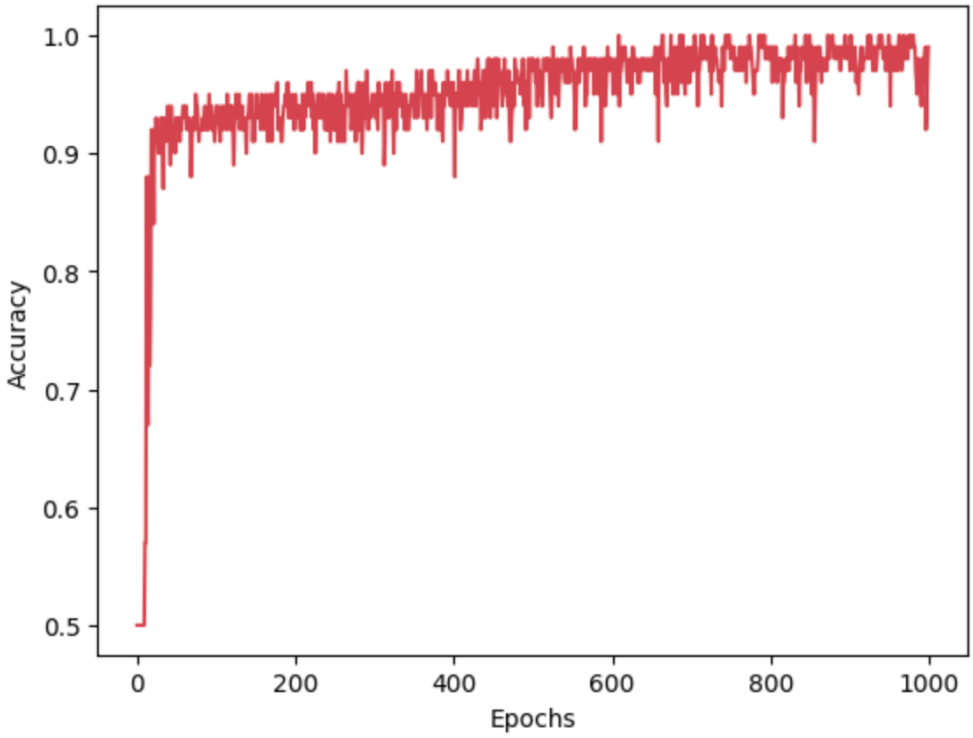
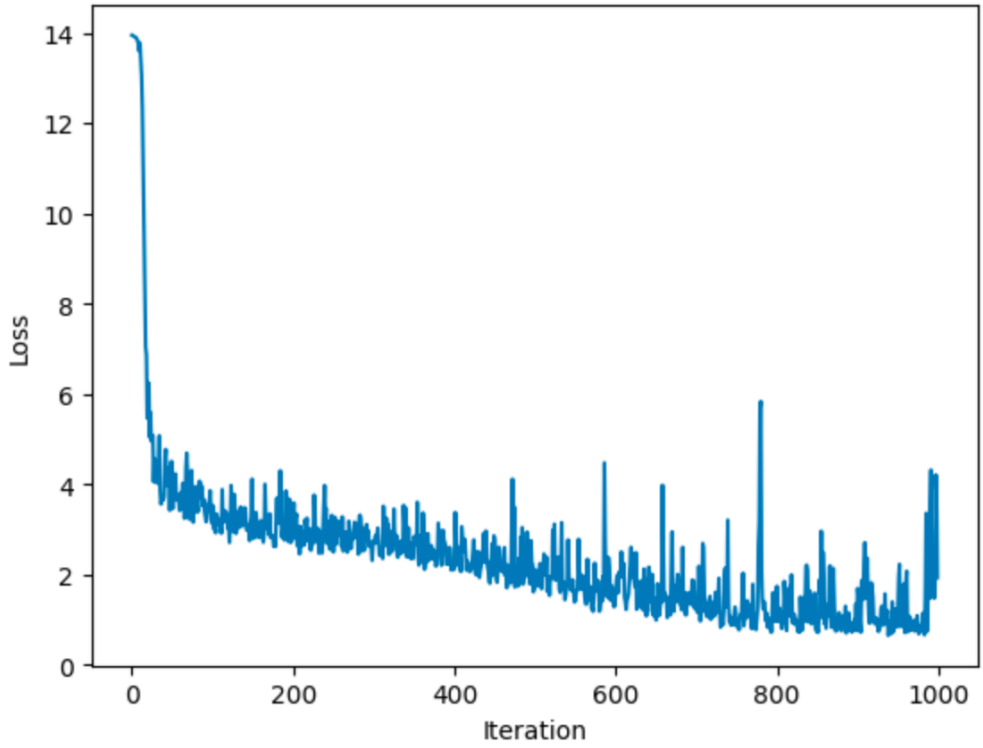
(a) For the training dataset with ⩾ 98% accuracy:

```
train accuracy: 99.0000 %
```



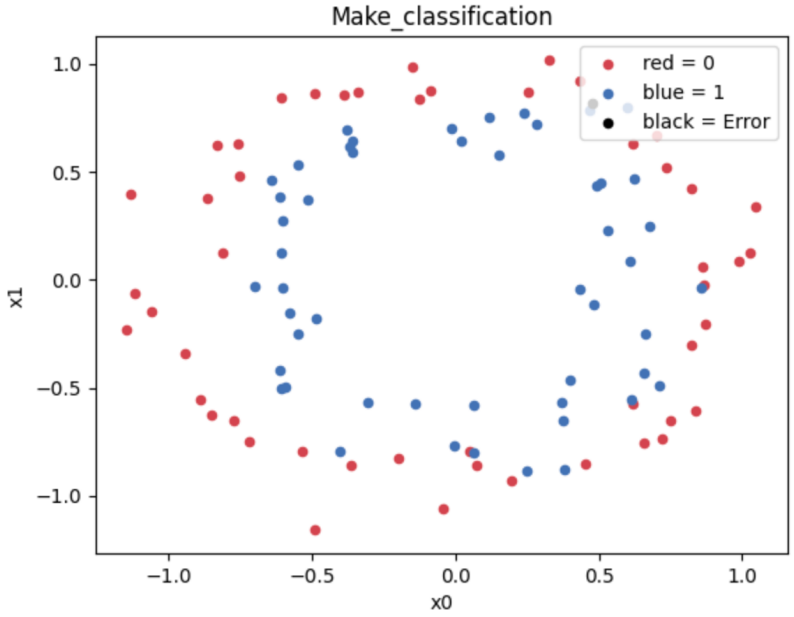



For the test set, we get the following accuracy rate:

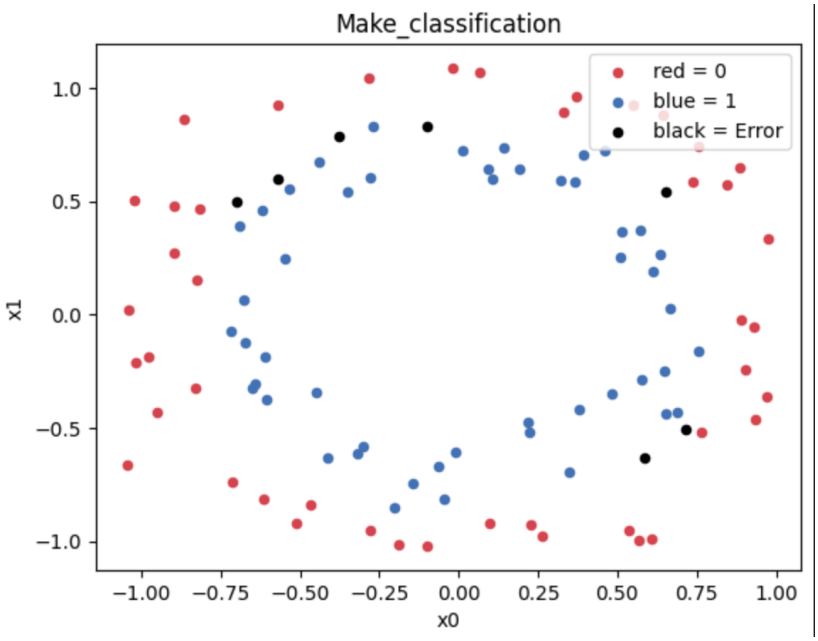
```
#test data analysis
Xtest, Ytest= hw2.test(network, x_test, y_test, batch_size = 10)

✓ 0.5s
Test accuracy: 0.9200000000000002
```

(b) The following plot shows which points in the training and test sets that are correctly classified using a scatter plot.

Train set:



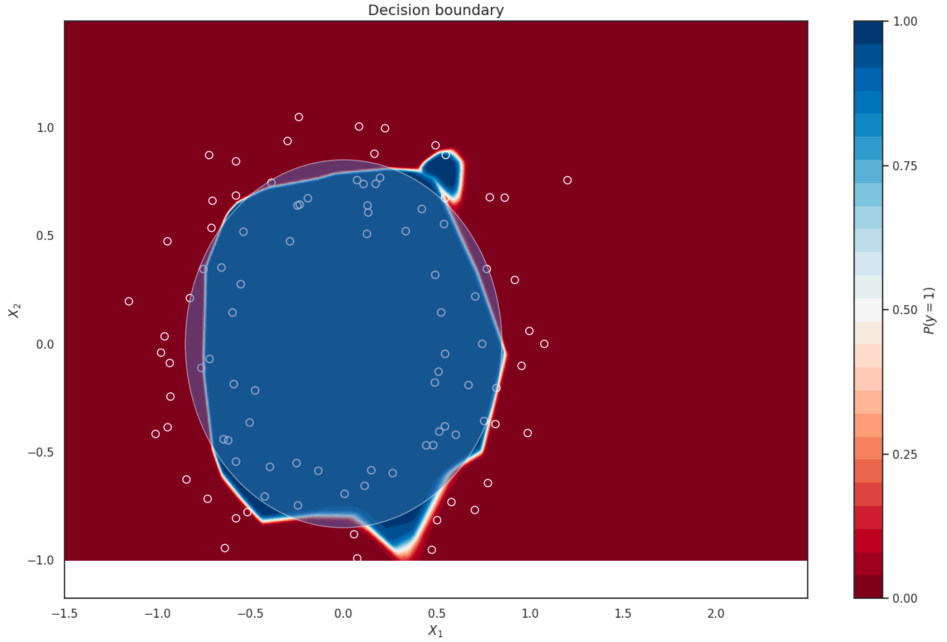

Test set:



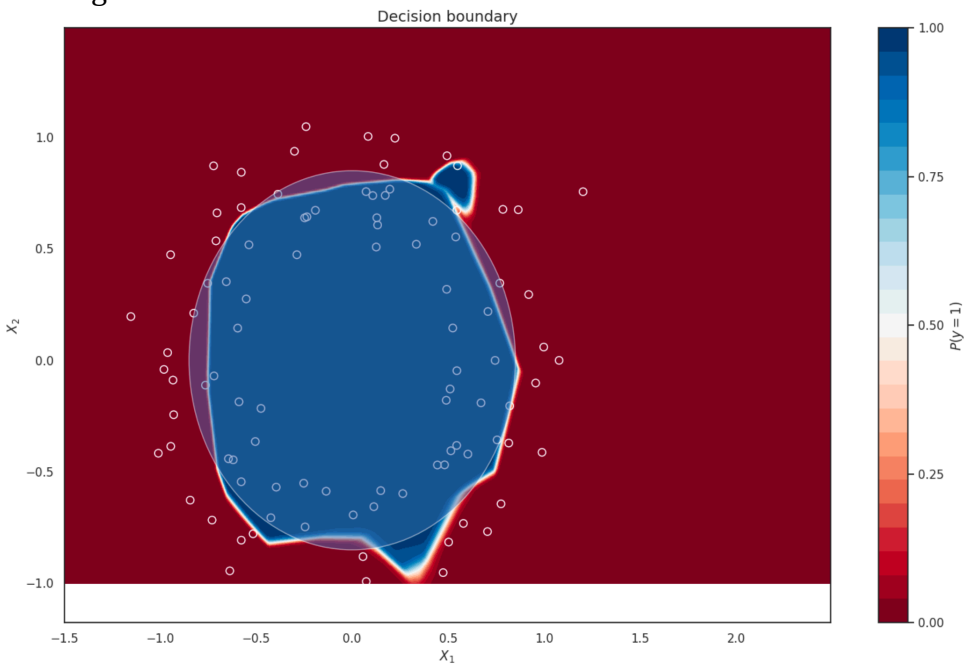
(c) For the training dataset:

*(The circle is the manual function for problem d)*

For the training dataset:




For the testing dataset:

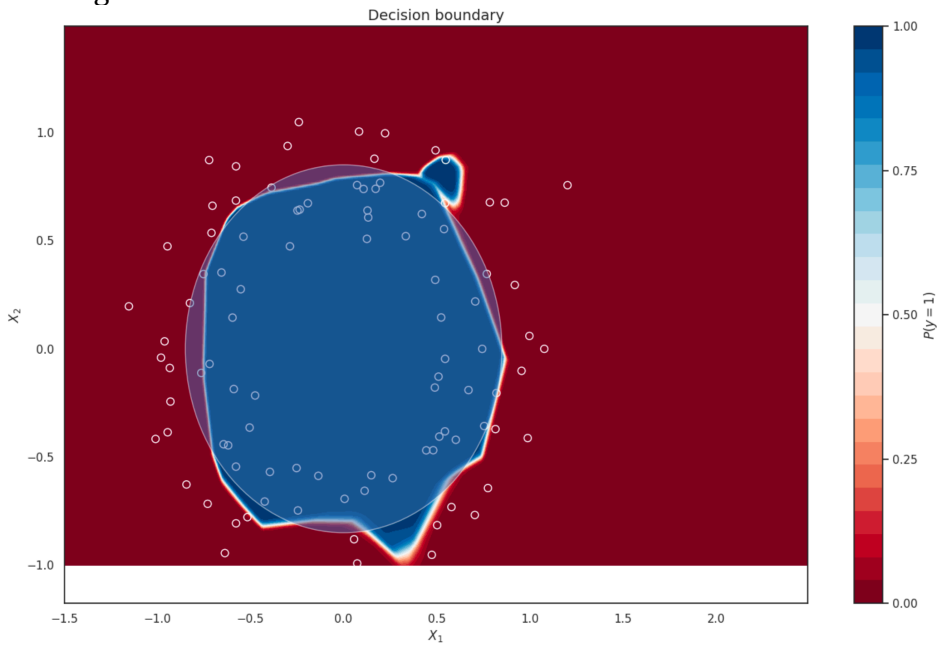



(d) Make a classifier manually:

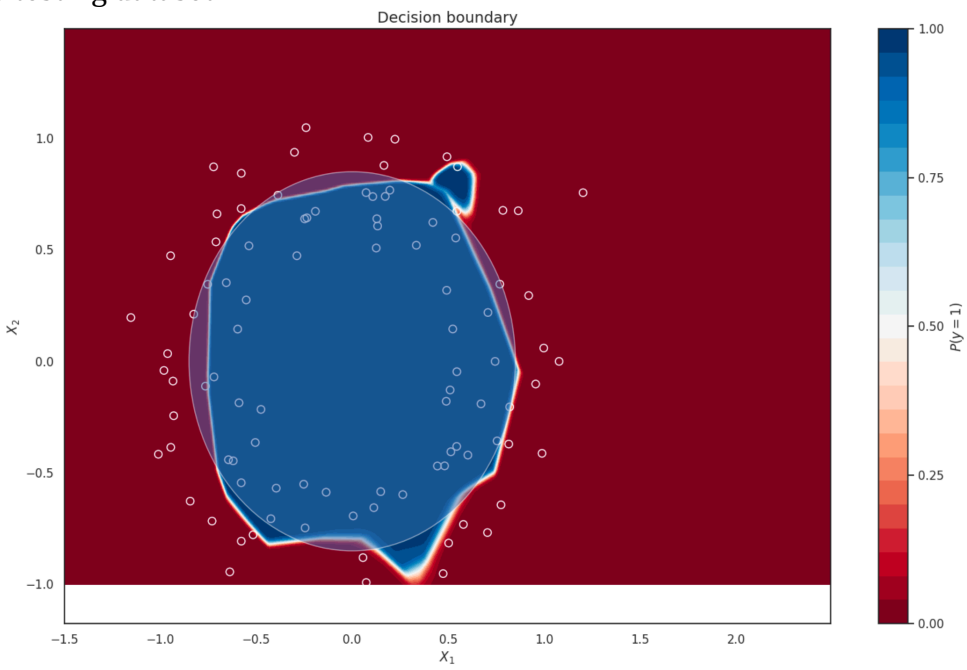
For the dataset, we draw a circle with the function x**2 + y**2 = 0.85**2

Then we have the plot on training set and the testing set.

For the training dataset:




For the testing dataset:




We calculated the accuracy:

For the training dataset

```
accuracy_manual_train = sum_train/len(y_train)
accuracy_manual_train
```

✓ 0.8s

```
0.89
```

For the testing dataset:

```
accuracy_manual_test = sum_test/len(y_test)
accuracy_manual_test
```

✓ 0.6s

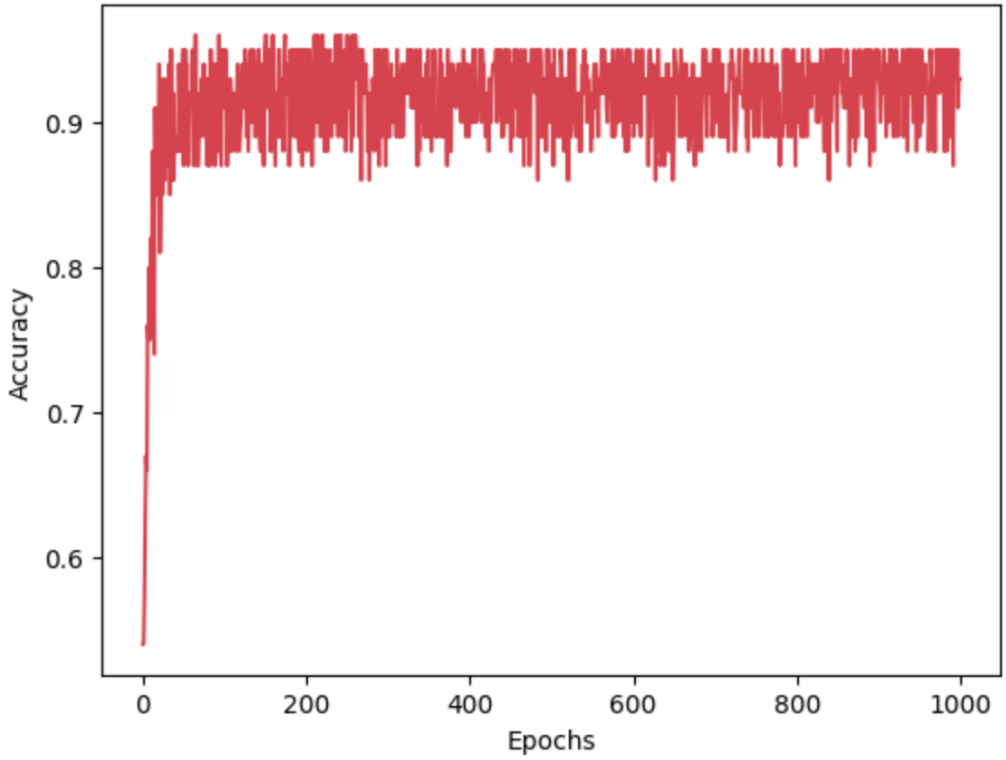
```
0.99
```

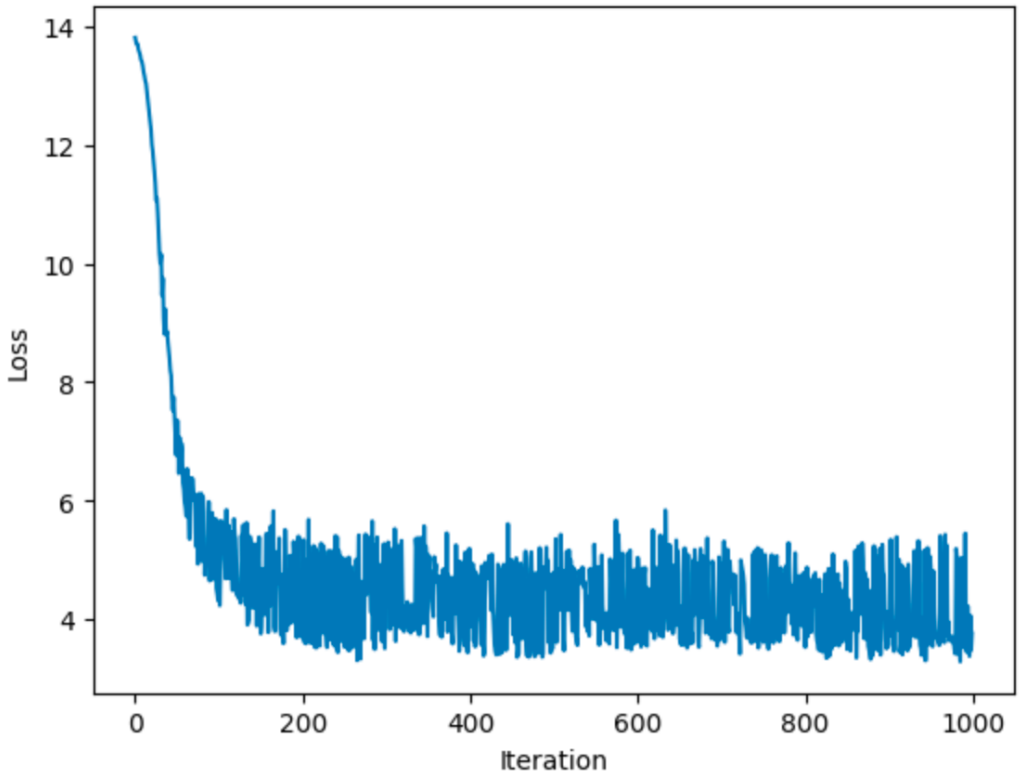
(Extra try: not overfit)

During each epoch, we use different random state parameter – different datasets for training.

For the training dataset:

```
train accuracy: 93.0000 %
```






For the testing dataset:

```
Test accuracy: 0.9500000000000002
```