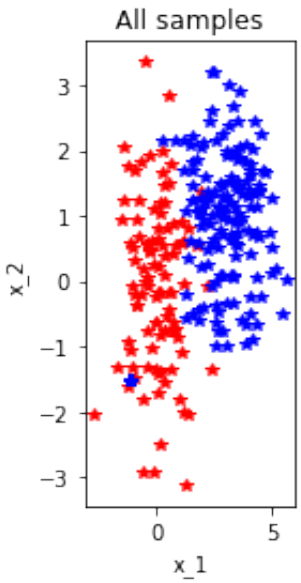
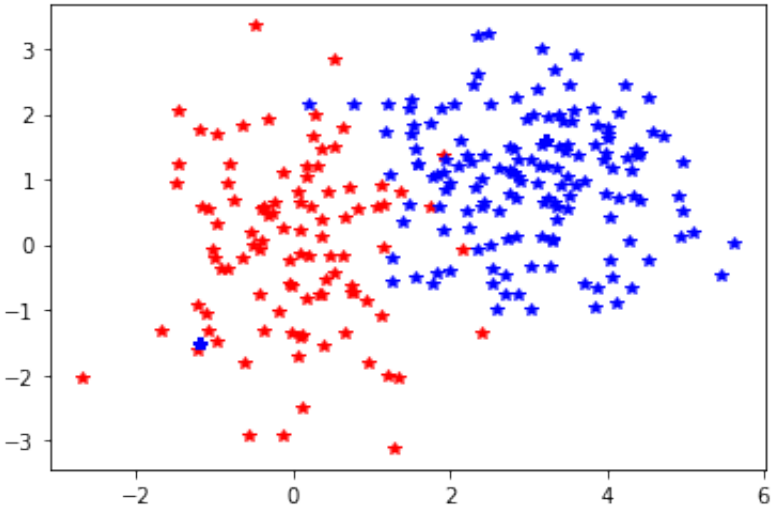
ECON 425 Homework 3

Madelyn Caufield

Part 1

Visualization of data:





```
0.9538461538461539
theta  [array([0.09110389]), array([0.03099945])]
cost is  [0.6086607]
theta  [array([1.02138757]), array([0.31402565])]
cost is  [0.34764366]
theta  [array([1.02783572]), array([0.31145209])]
cost is  [0.34763913]
theta  [array([1.02801296]), array([0.3112651])]
cost is  [0.34763912]
theta  [array([1.02801962]), array([0.31125591])]
cost is  [0.34763912]
```

Out[77]: To toggle on/off output_stderr, click here.

```
Scikit won.. :(
Your score:  0.8076923076923077
Scikits score:  0.9538461538461539
```

Based on the above methods (Scikit and gradient descent), Scikit appears to be performing better.

Applying logistic model over testing samples

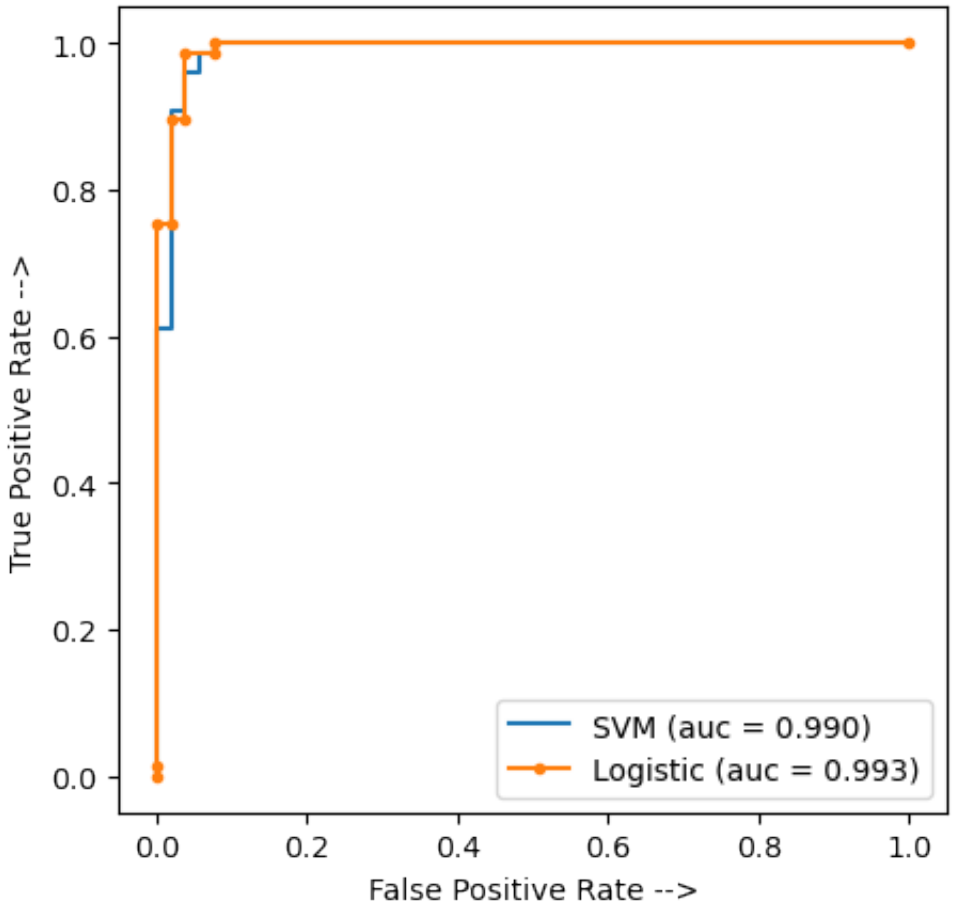
```
Optimization terminated successfully.
         Current function value: 0.329382
         Iterations 7
```

Out[79]:
```
array([[28., 25.],
       [ 0., 77.]])
```

Evaluation

```
average error: 0.4076923076923077 (0.4914054231902904)
```

ROC & AUC



Out[82]: To toggle on/off output_stderr, click here.

Part II

Accuracy:

Out[37]: 0.35

Precision (dog):

Out[38]: 0.25

Precision (cat):

Out[39]: 0.2

Precision (monkey):

Out[40]: 0.42857142857142855

Recall rate (dog):

Out[41]: 0.25

Recall rate (cat):

Out[42]: 0.16666666666666666

Recall rate (monkey):

Out[43]: 0.5

Confusion matrix for self developed implementation

Out[83]: To toggle on/off output_stderr, click here.

Out[84]: (59.23076923076923, 59.23076923076923, nan, 100.0, 0.0)

Part III

For Part 3, I split up the training data into k = 5 equal folds, and obtained five validation accuracies for each learning rate. I then calculated the mean validation accuracy for each learning rate. Once I had all ten mean validation accuracies for the ten learning rates, I selected the learning rate that provided the highest (best) accuracy. The code in the .py file provided the following scores.

The average accuracy for alpha = 0.001 is 0.78

The average accuracy for alpha = 0.002 is 0.77

The average accuracy for alpha = 0.005 is 0.77

The average accuracy for alpha = 0.01 is 0.77

The average accuracy for alpha = 0.02 is 0.77

The average accuracy for alpha = 0.05 is 0.75

The average accuracy for alpha = 0.07 is 0.75

The average accuracy for alpha = 0.1 is 0.75

The average accuracy for alpha = 0.5 is 0.75

The average accuracy for alpha = 1 is 0.75

The learning rate with the highest average accuracy was alpha = 0.001.

After training the logistic model with all the training samples, applying the training model over the testing samples, and calculating the models accuracy over the testing samples, we obtained the following scores.

Accuracy:

Alpha = 0.001: 0.83076923

Alpha = 0.002: 0.83076923

Alpha = 0.003: 0.83076923

Alpha = 0.01: 0.85384615

Alpha = 0.02: 0.85384615

Alpha = 0.05: 0.85384615

Alpha = 0.07: 0.85384615

Alpha = 0.1: 0.85384615

Alpha = 0.5: 0.85384615

Alpha = 1: 0.85384615

The alpha that gave us the highest score was not the alpha = 0.001 we expected previously. Here, alpha = 0.01,0.02,0.05,0.07,0.1,0.5,1 all gave us the same accuracye of 0.85384615.