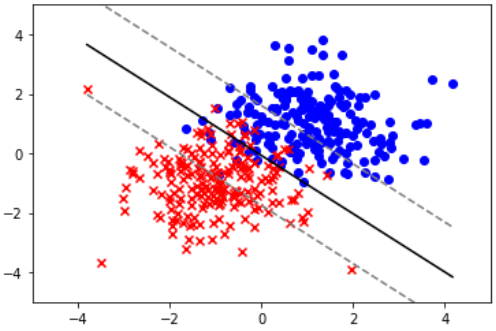
Machine Learning Programming Assignment 1

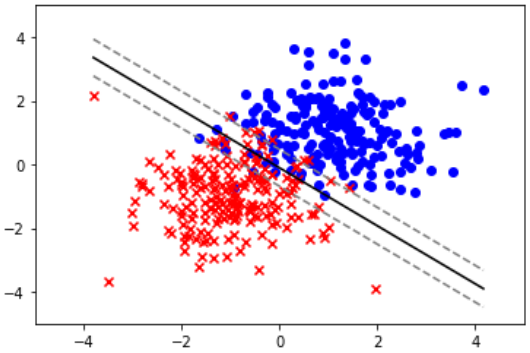
Name : Kuan-Chieh Peng || FSUid : kp19i

1. SVM

```
C= 0.01
w vector= [0.58333183 0.59592716]
margin is 1/||w||= 1.1991707650192347
```



```
C= 100000
w vector= [1.57402195 1.73024129]
margin is 1/||w||= 0.4275191071105674
```



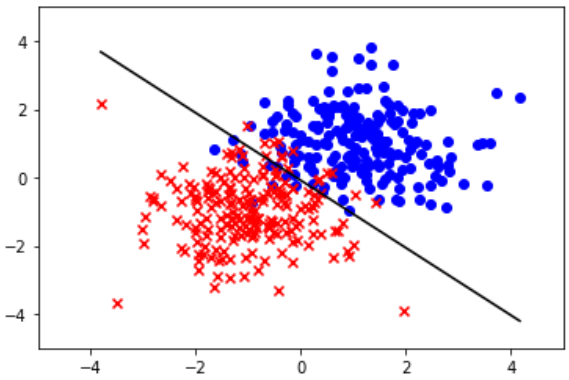
```
SVM Cross Validation Error: 7.000 %
```

About the method to get the support vectors, I use the following code:
(constraints.dual_value !=0) and (wx + b > 1 - zi)
This means that the dual value is not zero and also meet the requirement of support vectors, and because it is slack margin SVM, there are so many points to affect the margin, so I only put it in the notebook.
The penalty rate C, vector w and the margin are as the above. We can observe that the higher the C is, the thinner the margin is. This is exactly the definition of the C, the penalty.

```
SVM MNIST error 0.095 %
Time elapsed: 97.580  s
```

2. Linear Regression

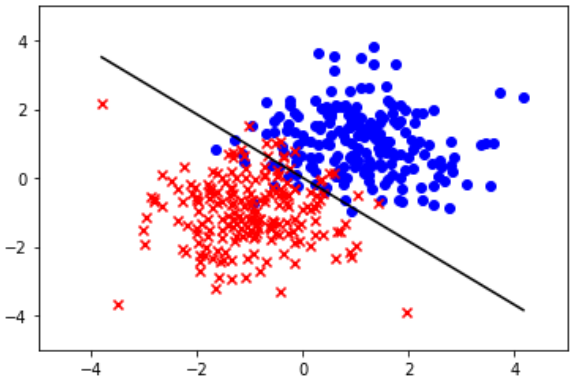
```
w= [0.34128436 0.34541478] b= -0.028602784386588715
Linear regression empirical error= 28.0 / 400 =7.000 %
```




```
Linear regression cross validation error: 28 / 400 = 7.000 %
Linear Regression MNIST error 23.404 %
Time elapsed: 0.192  s
```

3. Logistic Regression

```
beta= [2.47162046 2.67992781]
Logistic regression empirical error =7.000 %
```




```
Logistic regression cross validation error: = 7.000 %
optimization finished
MNIST error 0.236 %
Time elapsed: 486.148  s
```

Conclusion:

Wow, you can explicitly observe that the comparison of SVM, linear regression and logistic regression, SVM(`0.095 % error rate`)>logistic regression(`0.236 % error rate`)>linear regression(`23.404 % error rate`). This result is as we expected.