SUPPORT VECTOR MACHINES

Here, we are asked to implement the Pegasos algorithm and use LIBSVM library for the MNIST Handwritten digit classification. We are provided with MNIST data set, which contains the feature vectors and digit (class) assigned to each feature vector.

Q(a)

In this section, we have to implement the **Pegasos** algorithm.

$$\frac{1}{2}w^T w + C \sum_{i=m}^{m} \max{(0,1 - t_i)}$$

where, $t_i = y^{(i)}\left(w^T x^{(i)} + b\right)$

we have to make a solver which gives the values of w,b. In Pegasos algorithm, we have implemented Stochastic Gradient Descent (SGD) with a batch size of 100.

Q(b)

In this section, we implemented binary classification problem. We have used one-vs-one classification and feed each binary combination of classes to the formula derived in Q(a).

We have scaled the data between 0 and 1.

We have taken C = 1.0.

Train set accuracy: 88.70%

Test set accuracy: 89.92%

Q(c)

In this section, we have trained a multi class SVM on the MNIST data set using LIBSVM library.
We have scaled the data between 0 and 1.

We have taken C = 1.0 in both kernels.
We have used two kernels in the setting of LIBSVM parameters:

1. **Linear Kernel:**
   We will again compute test set accuracy for one-vs-one classification using LIBSVM.

   Parameter to libsvm: '-s 0 -t 0 -c 1’

   Test set accuracy: 92.76%

2. **Gaussian Kernel:**
   Here we will compute test set accuracy for one-vs-one classification using LIBSVM. ($\gamma = 0.05$)

   Parameter to libsvm: '-s 0 -t 2 -c 1 -g 0.05’

   Test set accuracy: 97.22%

Observations:
Linear kernel accuracy: 92.76%

one-vs-one classification accuracy: 89.92%. There is a marginal difference (2%) between the accuracies of both the implementations. Linear kernel method is slightly better.

Gaussian kernel has better accuracy compared to both.

Q(d)

In this section, we have implemented 10 fold Cross-Validation using LIBSVM.

Here γ is fixed at 0.05 and we will vary C in the set $\{10^{-5}, 10^{-3}, 1, 5, 10\}$.

We will compute the 10-fold cross validation accuracy for each value of C and also compute the corresponding test set accuracy.

1. **C = 10^{-5}**

   Parameter to libsvm: '-s 0 -t 2 -c 0.00001 -g 0.05 -v 10'
   Cross Validation accuracy: 71.25%

   Parameter to libsvm: '-s 0 -t 2 -c 0.00001 -g 0.05'
   Test set accuracy: 71.92%

2. **C = 10^{-3}**

   Parameter to libsvm: '-s 0 -t 2 -c 0.001 -g 0.05 -v 10'
   Cross Validation accuracy: 71.52%

   Parameter to libsvm: '-s 0 -t 2 -c 0.001 -g 0.05'
   Test set accuracy: 71.92%

3. **C = 1**

   Parameter to libsvm: '-s 0 -t 2 -c 1 -g 0.05 -v 10'
   Cross Validation accuracy: 97.38%

Parameter to libsvm: '-s 0 -t 2 -c 1 -g 0.05'
Test set accuracy: 97.22%

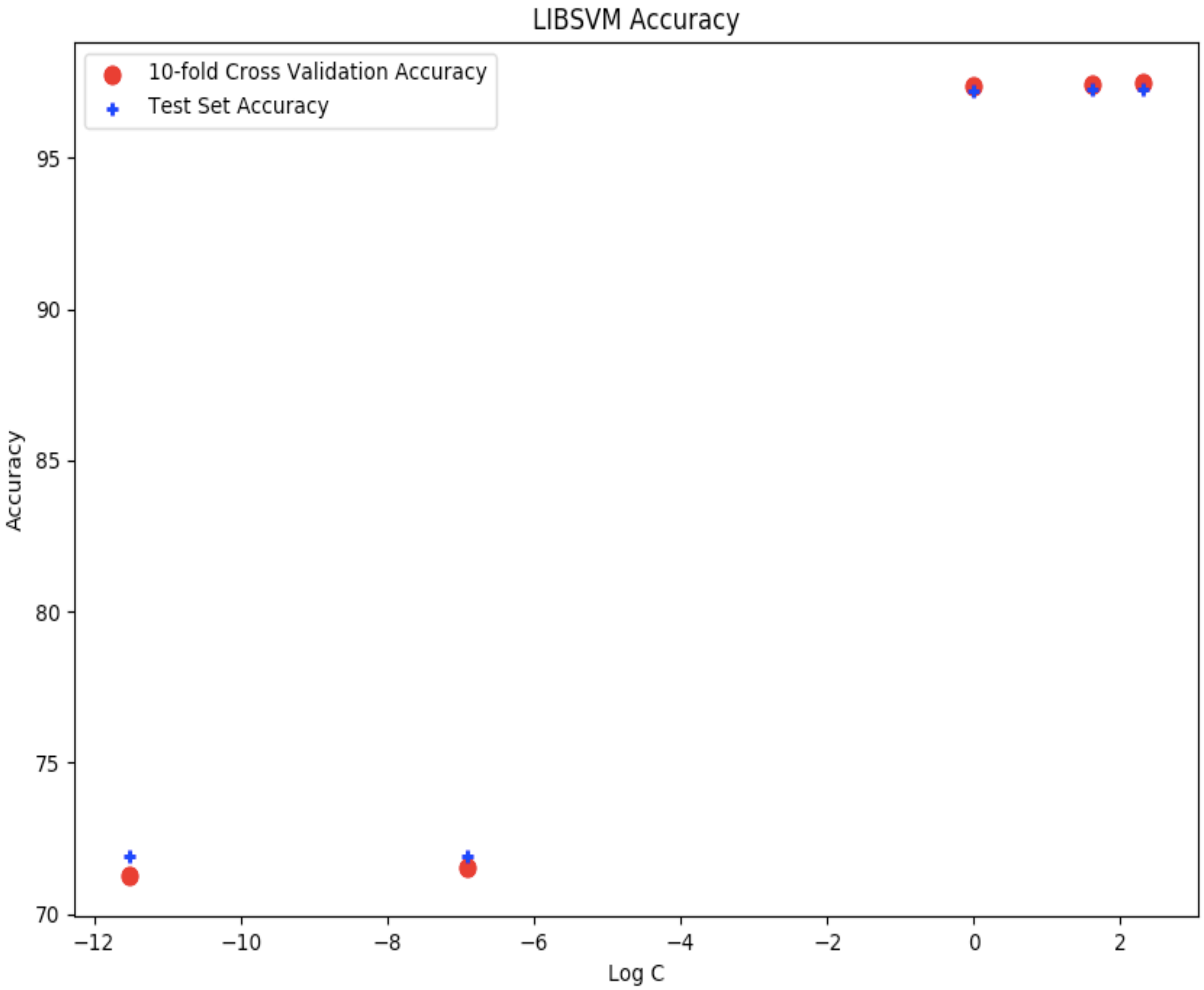
4. C = 5
Parameter to libsvm: '-s 0 -t 2 -c 5 -g 0.05 -v 10’
Cross Validation accuracy: 97.45%

Parameter to libsvm: '-s 0 -t 2 -c 5 -g 0.05'
Test set accuracy: 97.29%

5. C = 10
Parameter to libsvm: '-s 0 -t 2 -c 10 -g 0.05 -v 10’
Cross Validation accuracy: 97.5%

Parameter to libsvm: '-s 0 -t 2 -c 10 -g 0.05'
Test set accuracy: 97.29%

We have plotted the accuracies obtained for different values of C vs log scale of C.



Observations:
For lower values of C, test set is accuracy is greater than cross validation accuracy. As we increase the value of C, test set accuracy becomes smaller then cross validation accuracy.
The difference between both the accuracies is marginal.

Cross validation accuracy is highest for C = 10. This value of C also gives the highest test set accuracy. Though C = 5 also gives the same highest test set accuracy.

So test set accuracy is highest for C = 5 and 10, while validation accuracy is highest for C = 10.

We also see that in general accuracy is lower for lower value of C and it increases as we increase C.
We can say that as C increases, the data becomes more and more linearly separable. For high value of C, we have to pay more cost even for a single penalty.
As value of C decreases, penalty becomes less for violations.

Q(e)

In this section, we have drawn the confusion matrix for the result which gives the best test set accuracies in Ques 2(d) part.

Below is the confusion matrix for C = 10 case, where test accuracy is 97.29%

Confusion matrix:

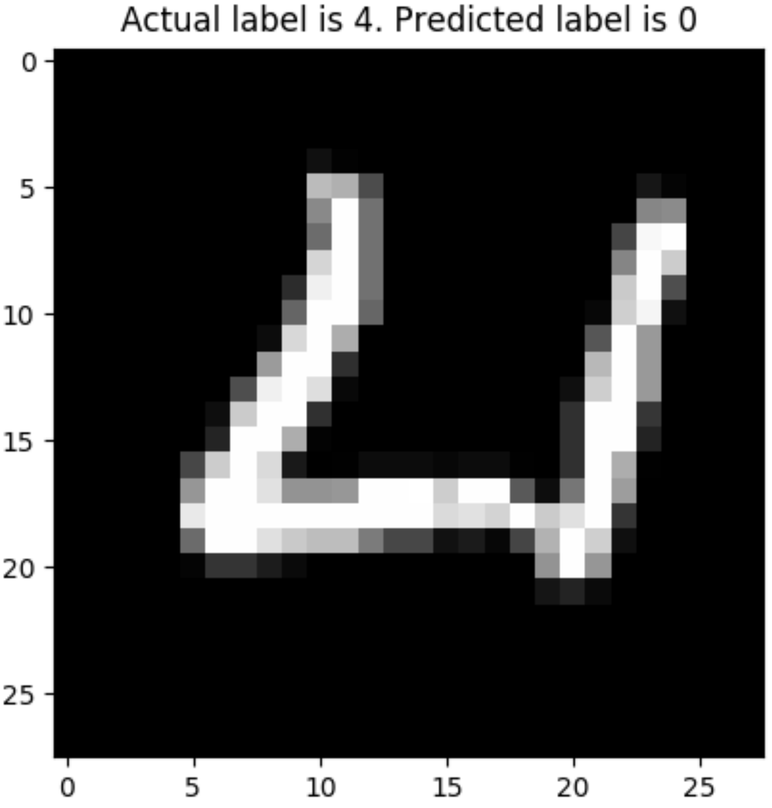
		Actual classes									
		[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]
Predicted classes	[0]	[[969	0	4	0	1	2	5	1	4	4]
	[1]	[0	1122	0	0	0	0	4	4	0	4]
	[2]	[1	3	1000	8	4	3	0	20	3	3]
	[3]	[0	2	4	985	0	6	0	2	10	8]
	[4]	[0	1	2	0	962	1	3	3	1	9]
	[5]	[3	2	0	4	0	866	4	0	5	4]
	[6]	[4	2	1	0	5	7	940	0	3	0]
	[7]	[1	0	6	7	0	1	0	986	3	9]
	[8]	[2	2	15	5	2	5	2	2	942	11]
	[9]	[0	1	0	1	8	1	0	10	3	957]]

Observations:

Last class (class 9) is most difficult to classify because the data is distributed for other classes also. (last column). For this class, the confidence of the algorithm is low.(94%)

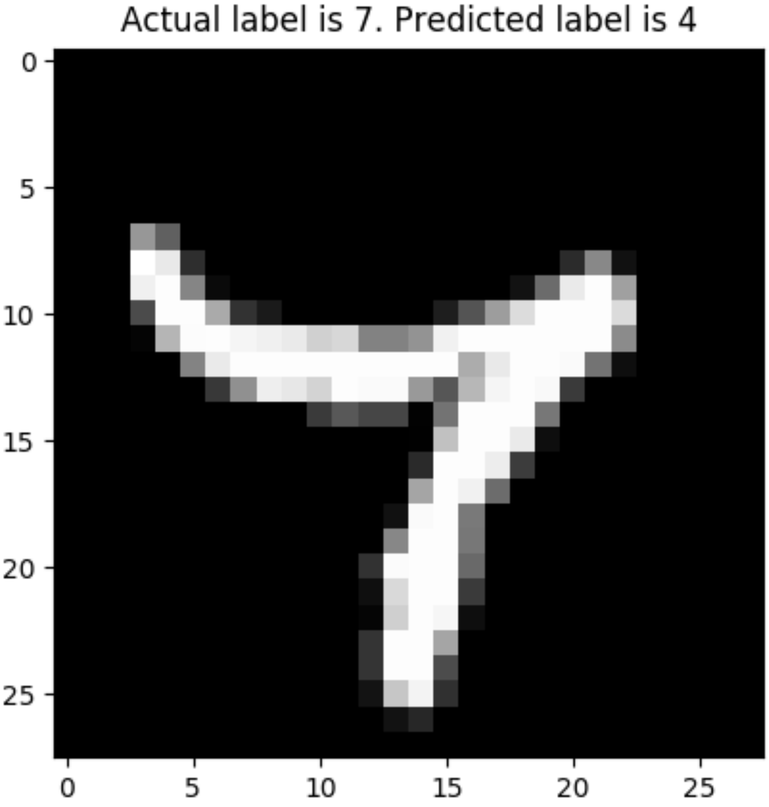
3 wrongly predicted test examples:

1. Row number 34th

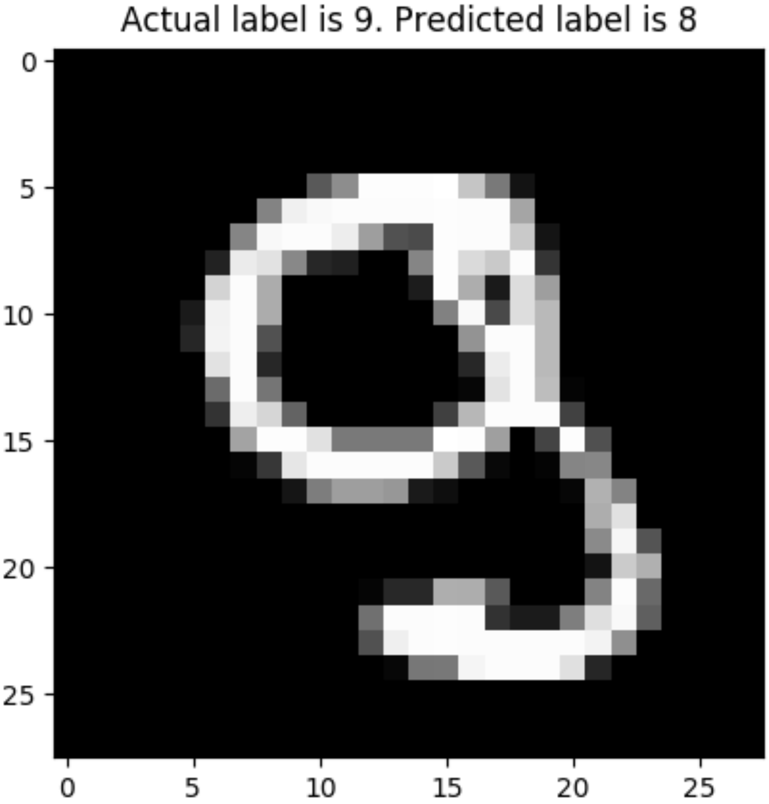


Actual label is 4. Predicted label is 0

2. Row number 125th



3. Row number 152nd



Actual label is 9. Predicted label is 8