Question - 3

-Deepanshu Garg(201501167)

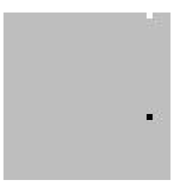
L1

The different accuracies for different lambda (which is 1/C) for L1 are:

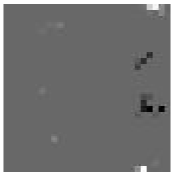
l1
lambda = 10000000.0
accuracy = 0.492882562278

l1
lambda = 1000000.0
accuracy = 0.492882562278

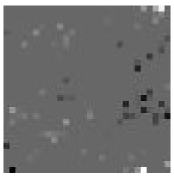
l1
lambda = 100000.0
accuracy = 0.891459074733



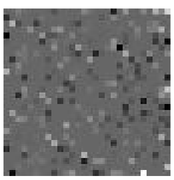
l1
lambda = 10000.0
accuracy = 0.945729537367



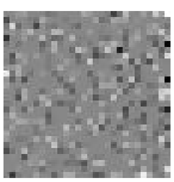
l1
lambda = 1000.0
accuracy = 0.955516014235



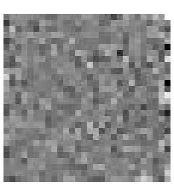
l1
lambda = 100.0
accuracy = 0.953736654804



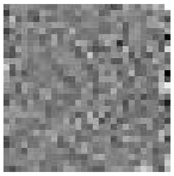
l1
lambda = 10.0
accuracy = 0.940391459075



l1
lambda = 1.0
accuracy = 0.940391459075



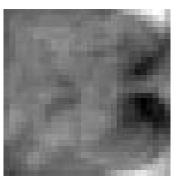
l1
lambda = 0.1
accuracy = 0.937722419929



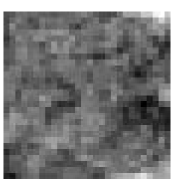
L2

The different accuracies for different lambda (which is 1/C) for L2 are:

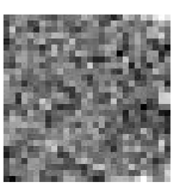
l2
lambda = 10000000.0
accuracy = 0.955516014235



l2
lambda = 1000000.0
accuracy = 0.960854092527

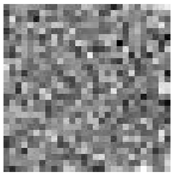


l2
lambda = 100000.0
accuracy = 0.952846975089

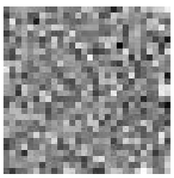


l2
lambda = 10000.0

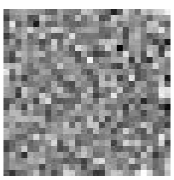
accuracy = 0.940391459075



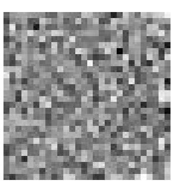
l2
lambda = 1000.0
accuracy = 0.936832740214



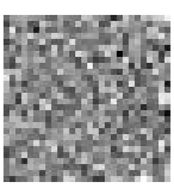
l2
lambda = 100.0
accuracy = 0.936832740214



l2
lambda = 10.0
accuracy = 0.935943060498

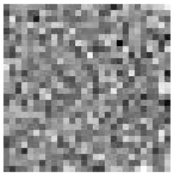


l2
lambda = 1.0
accuracy = 0.936832740214



l2
lambda = 0.1

accuracy = 0.934163701068

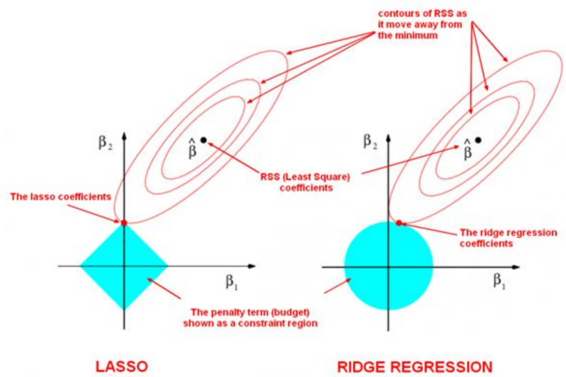


Conclusions:

1. We see that we get max accuracy on the test data with L1 loss with gamma = 1000 (C=0.001) and L2 loss with gamma = 10^7.
2. Lower the C, stricter the regularization is. We see that L2 loss shows evident difference when Gamma is high and changed a bit. Even changing Gamma by a scale of 10 shows great difference in the resulting weight matrix.
3. In both, L1 and L2 loss, we see that lower the gamma, the model starts to overfit on the data and hence bad accuracy on test data. So, after one limit (Gamma=0.1 for L1 and Gamma=1000 for L2) decreasing gamma further wont help since the model is already overfitting after this.

Inferences:

1. The images we printed are of the weights vectors. The images which appear “smoother” i.e. evenly spread out, have weights which are evenly distributed. The more “grainy” images on the other hand have weights which are more sparsely distributed i.e. some weights have very high value and some have very low values.
2. As we vary lambda, we see a change in distribution of the weights.
    a. Increasing the lambda means that we’re giving more weightage to the regularization term in the loss function, which will tend to make the weights more evenly distributed, which theoretically should make the classifier more general at cost of wrongly classifying a few of the data points.
    b. Decreasing lambda on the other hand means we’re giving more weightage to correctly classifying the training data w.r.t. to the regularization term, which may sometimes lead to overfitting.
3. We observed that our weights were in the range [-1, 1], so, between L1 and L2 we see:
    a. Since L2 squares the weights, the regularization loss would be smaller. Hence, weights would react slower to change in lambda.
    b. Whereas L1 directly takes sum of the absolute values of the weights, so, the regularization loss would be higher, hence, the change in weights would be more prominent with changing lambda.

4. 

Looking at the lp ball, we infer that l1 would intersect the loss function contour at the edges, hence it would be sparse. Since l2 ball would generally not intersect "at edges", l2 would generally not give sparse weight vectors.