Introduction to Machine Learning *Phani Teja Kesha, KP38691*

HW2

Please note that only PDF submissions are accepted. We encourage using LaTeX to produce your writeups. You'll need *mydefs.sty* and *notes.sty* which can be downloaded from the course page.

1. Explain in what case and why we need a regularizer and why l_2 norm regualzrizer is a reasonable one.

   To generalize the test data we must make sure that the model will not try to overfit. So a regualizer is introduced over the parameters of the model ,this regularizer will not let the model try to overfit. The reason why we use a l_2 norm regualizier is l_2 norm is smooth and convex . This makes its easy to calculate the gradient decendant.

2. What values of λ in the regularized loss objective will lead to overfitting? What values will lead to underfitting? Note that λ is a multiplier for the regularizer term.

   The reason why we use a regularizer is it will make the model not to overfit. Regularizer λ is the product of regularizer and R(w) (simpler model). When λ is small the product of λ and R comes small which leads the weight vector very far from the zero ,due to this the model will lead to overfitting.. When λ is large the product is more and the weight vector will tend to be near to zero ,hence model will lead to underfitting.

3. Explain why the squared loss is not suitable for binary classification problems.

   squared loss $= \sum_n (y_n - \hat{y}_n)^2$. Consider a wrong prediction then in binary prediction ,the loss will be (-1-1)=-2,when squared is equal to 4. That means when a prediction is right the loss will be zero and when the prediction is wrong the loss will be 4. This is the disadvantage of squared loss because all the wrong predictions even if they are outliers or not outliers ,the loss will be 4. Hence it is not suitable for binary classification.

4. One disadvantage of the squared loss is that it has a tendency to be dominated by outliers – the overall loss $\sum_n (y_n - \hat{y}_n)^2$, is influenced too much by points that have high $|y_n - \hat{y}_n|$. Suggest a modification to the squared loss that remedies this.

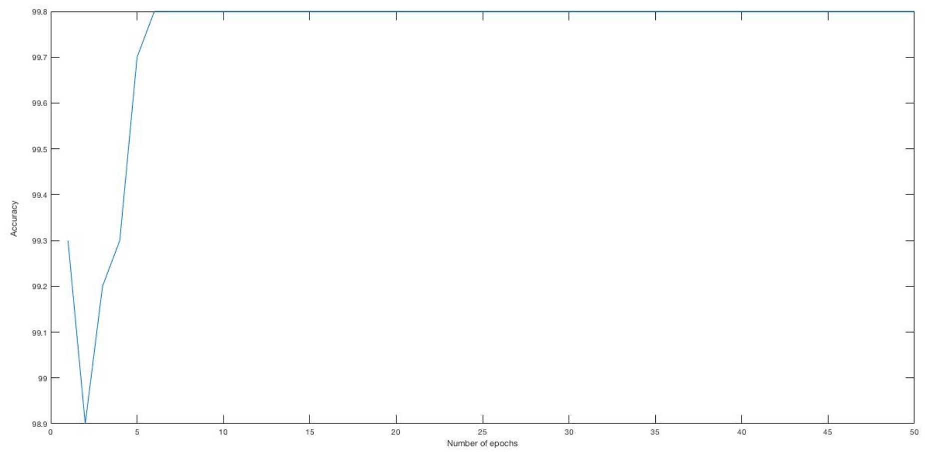
   Huber loss is a loss function used in regression, that is less sensitive to outliers in data than the squared loss error. In huber loss a threshold value is taken such that the loss is defined and the loss is calculated by using the threshold value.Loss is calcuated differently for outliers,due to this loss is decreased and penalty less .

5. Perceptron algorithm:

   (a) Implement the perceptron algorithm for binary classification.

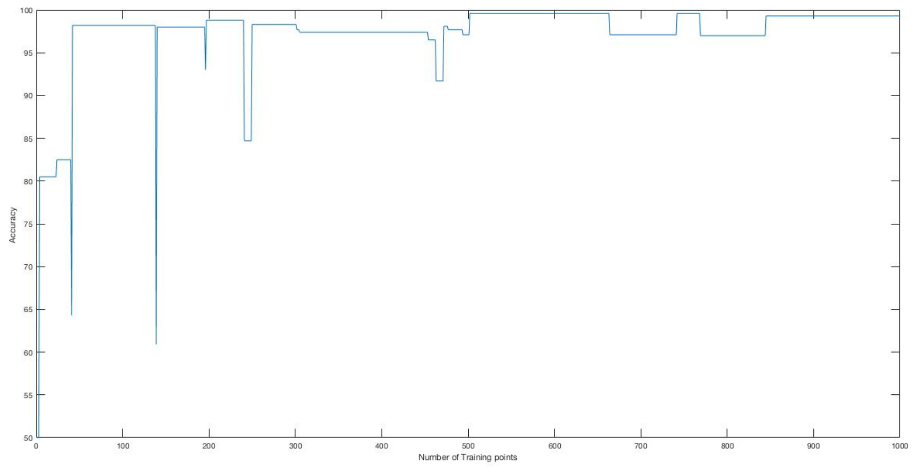
   Accuracy =99.5 for 6 Epochs

(b) Train and test it for classifying digits "1" and "6" in MNIST dataset. Note that we don't need the data of other digits in this part. Please use 1000 training examples and 1000 testing examples (500 for each class). report the accuracy after a reasonable number of iteration when the accuracy does not change or starts going down.

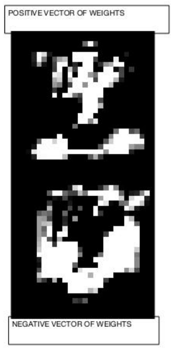


Accuracy = 99.5, after 6th Iteration the accuracy does not change or starts going down.

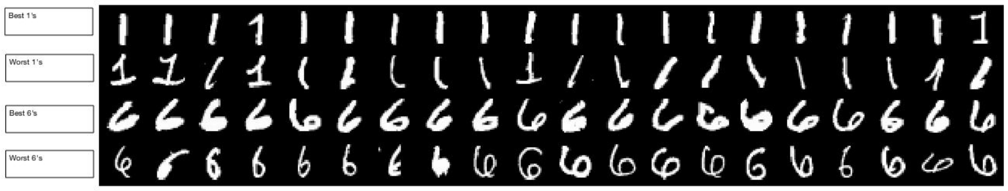
(c) Plot the accuracy on the test set w.r.t. the number of iterations. Here, processing each data-point is considered one iteration, so 1000 iterations means one pass over all training data. For this, you need to evaluate the model very often (once for each data point).



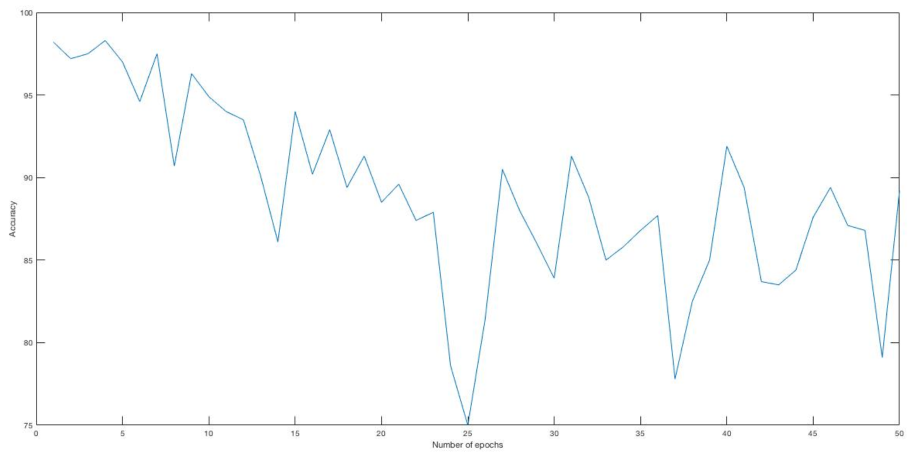
(d) Visualize the learned model in the image form to see if it makes sense. Note that the weight vector has both positive and negative values, so you should plot those on two separate planes. Note that you should use exactly the same testing data to be able to compare the results.

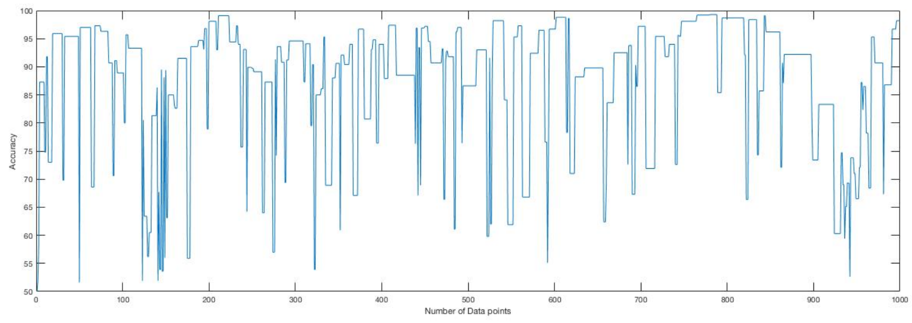


(e) Visualize the 20 best scoring and 20 worst scoring images from each class.



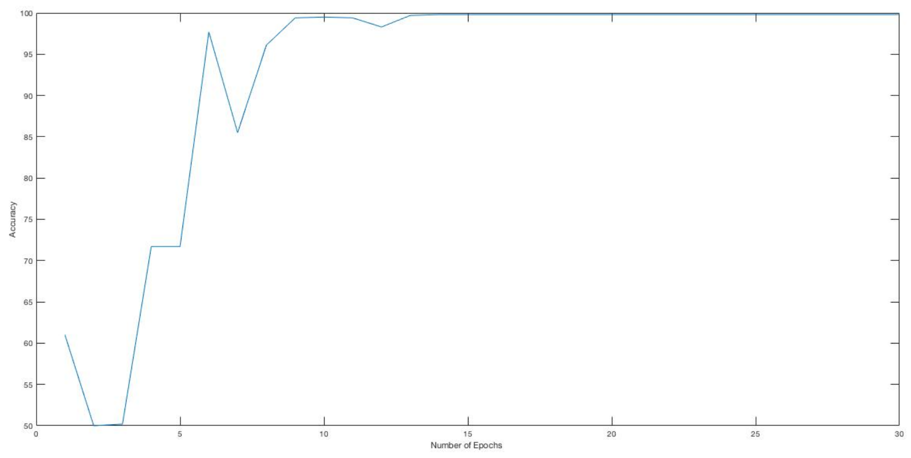
(f) Randomly flip the label (add labeling error) for 10% of the training data and repeat (b) and (c).



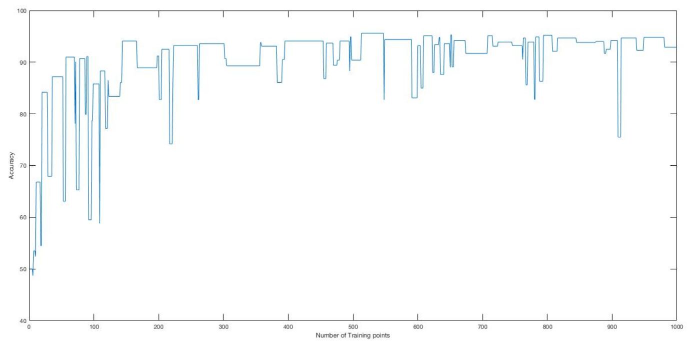


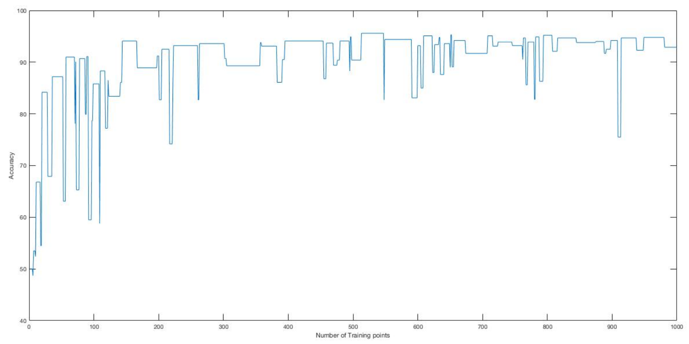
(g) Sort the data before training so that all "1"s appear before "6"s and plot the accuracy w.r.t. the number of iterations. Is this faster or slower in training? By faster, we mean the number of iterations needed to get a good model. Explain why.

It is slower in training because it is taking 14 iterations to get a good model. Since the data is in a sorted way and the model takes more time to learn from the given data .



(h) Repeat (b), (c), and (d) for classifying ”2” and ”8”. Is this faster or slower in training? Again, by faster, we mean the number of iterations needed to get a good model. Explain why.

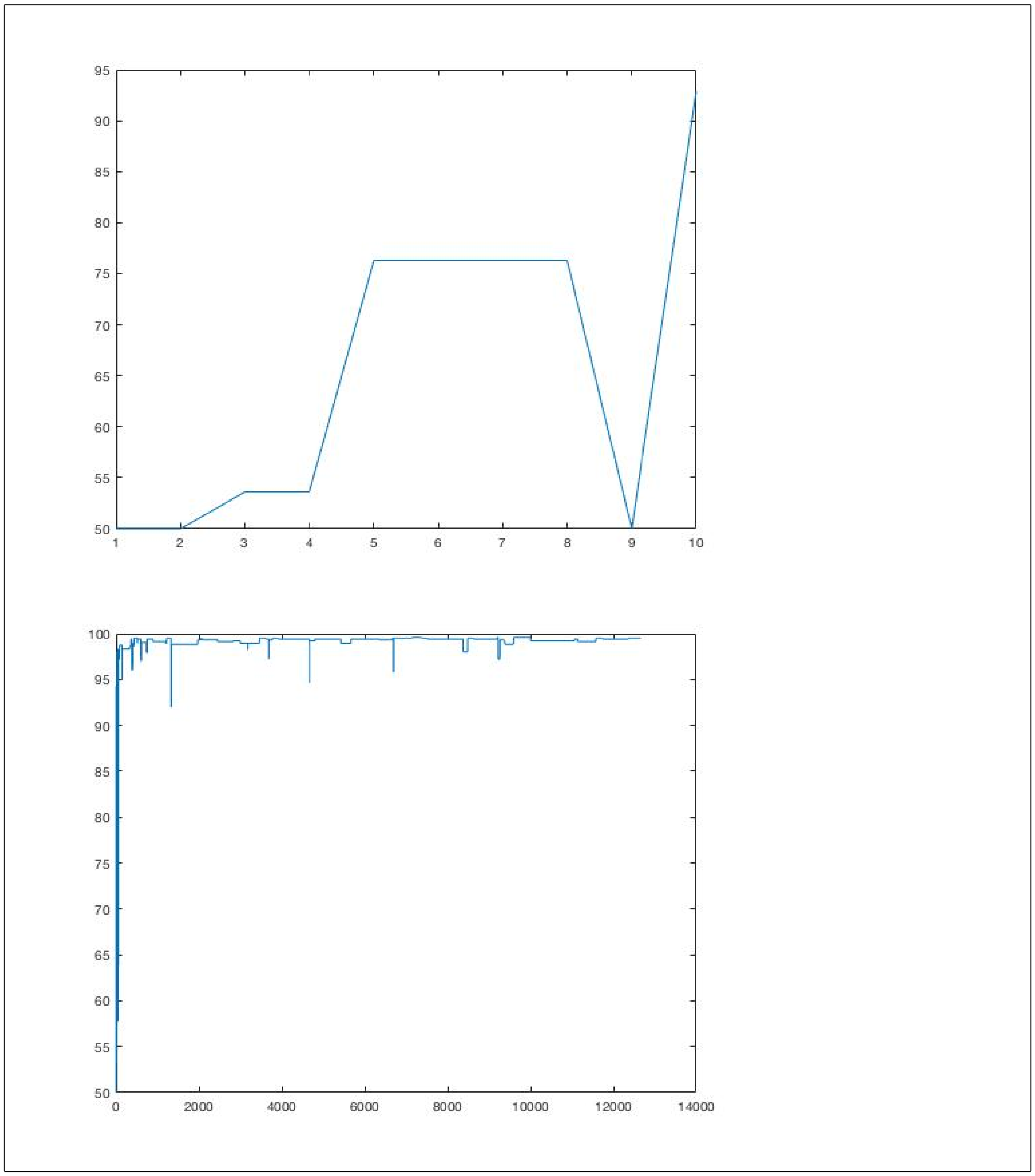




Positive Vector - 2
Negative Vector - 8

The model is taking 26 epochs for getting a good model,This change is due to the error’s in the given training data .

(i) Repeat (b) and (c) once with 10 training examples (5 for each class) and then all almost 12,000 training examples. Use the same test set that you used before.


95
90
85
80
75
70
65
60
55
50
1
2
3
4
5
6
7
8
9
10
100
95
90
85
80
75
70
65
60
55
50
0
2000
4000
6000
8000
10000
12000
14000