

ML Assignment 2

Author : Aman Bhatia

Dated : 16 Mar 2016

Internet Advertisement Classification

(a) I used threshold to be 10^{-4} for separating support vectors from alpha's. The set of support vectors obtained is reported in file “support_vector_cvx_linear”. Number of support vectors on thresholding is 360.

(b)

- Weight vector (w) : [-0.00506373, 0.00434291, -0.05313879, ..., 0.23288778, 0.44835903, -0.02973599]
- Intercept Term (b) : -1.555641474598324
- Accuracy on test data : 98.33 %

(c)

I used threshold to be 10^{-4} for separating support vectors from alpha's. The set of support vectors obtained is reported in file “support_vector_cvx_gaus”. Number of support vectors on thresholding is 947.

- Accuracy on test data : 98.72 %

As we can see that we have slight increase in the accuracy which was expected by gaussian kernel.

(d)

- Accuracy with linear kernel : 98.2028%
- Accuracy with gaussian kernel : 98.7163%

We can see that almost the same accuracies are obtained in CVX as well as LibSVM. In both of them, the accuracy for the gaussian kernel case is slightly more than linear kernel which is expected. No. of support vectors for gaussian kernel is 646 and for linear kernel is 319.

(e) One Criteria we can use is that for points for which the value is not known, we can replace it with the average value of that attribute of that class. For example, we can replace the zeros of 3rd attribute of “ad” class, with average value of the 3rd attribute of “ad” class and so on for all the three attributes and both the classes. This criteria will surely be very good compared to zero unless the average we are calculating is already around zero. This is because we are not using some random value to fill the unknown data but we are using the average value that attribute represents.

Digit Recognition

(a) Example usage for visualize script :-

```
python3 visualize.py train3 10
```

(b) Stopping Criteria : When difference between two consecutive epochs is less than 1.

(c)

- Accuracy : After 14~15 iterations, accuracy on test data is ~98%
- Training Time : Each epoch takes on an average ~14 seconds. So, total training time taken is ~ 3minutes 45 seconds

(d) We need atleast 4 output units in this case. However, I have used 10 output units and used labels as one hot vectors for simplicity.

- Accuracy : After ~25 iterations, accuracy on test data is ~96%
- Training Time : Each epoch takes on an average ~75 seconds. So, total training time taken is ~ 30 minutes

The training time for one epoch is almost 5 times compared to 3-8 binary case.

----- *END OF DOCUMENT* -----