Homework 2: CS 498 AML: ONL

Ramya Narayanaswamy (rpn2@illinois.edu), Jeff Hong (hong66@illinois.edu)

Short description of the algorithm for solving HW2 is provided here:

The assignment was coded in R. The data was preprocessed (centered and scaled) as per the requirements and is split into testing (Te), training (Tr) and evaluation(E) as per the instructions. The high-level details of the SVM with stochastic gradient is provided below

1. The loop L1 is the outer loop over a set of lambda values
   a. The coefficient of the linear classifier (a,b) are initialized to a random value
2. The loop L2 is the middle loop over the number of epoch's for every lambda value
   a. 50 samples are chosen at random as held out for epoch
   b. Learning rate is updated as per 100/(5000 + epoch_num)
3. The loop L3 is the inner loop over number of steps for every epoch
   a. A random sample is chosen at every step
   b. Sign(y(ax+b)) is calculated for the chosen sample
   c. Depending on the sign, the co-efficient (a,b) are updated
   d. Every 30 steps, accuracy and magnitude are calculated for 50 samples gathered in step 2a
4. After all the epoch's for a lambda is complete, accuracy and magnitude for evaluation set (E) is calculated
5. After L1 is complete for all lambda values, best lambda is chosen. The training (Tr) and Evaluation (E) are combined to get overall training (T)
6. Re-initialized the co-efficients (a,b) and fixed the lambda. Repeated the SVM and stochastic gradient descent on all of training data (T) to get a linear classifier model
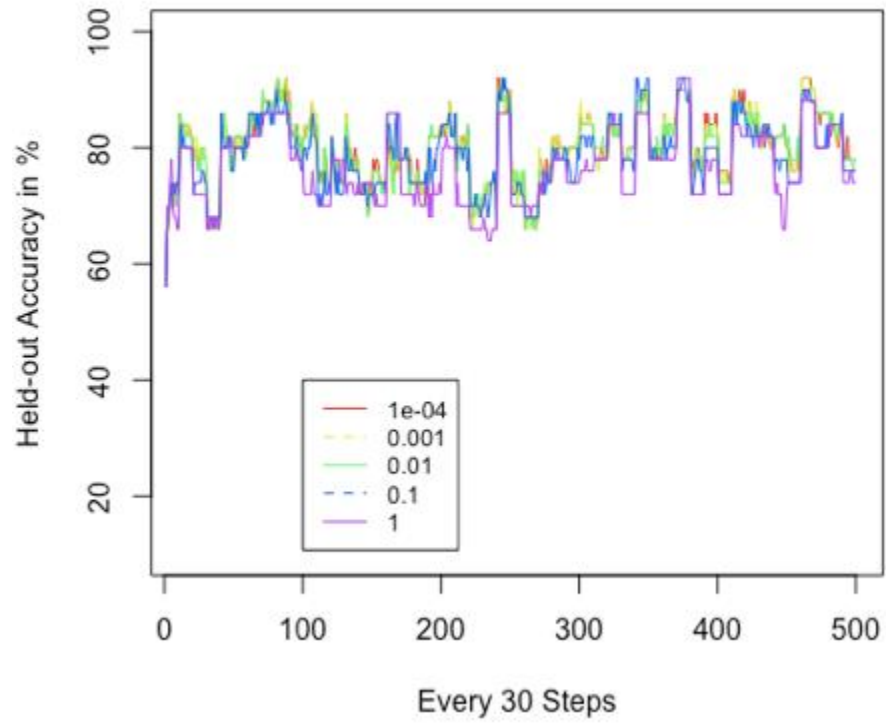7. The model from Step 6 is evaluated with the test data (Te)

List of references :
1)   https://www.r-bloggers.com/getting-data-from-an-online-source/
2)   Piazza posts : @426, @220, @236
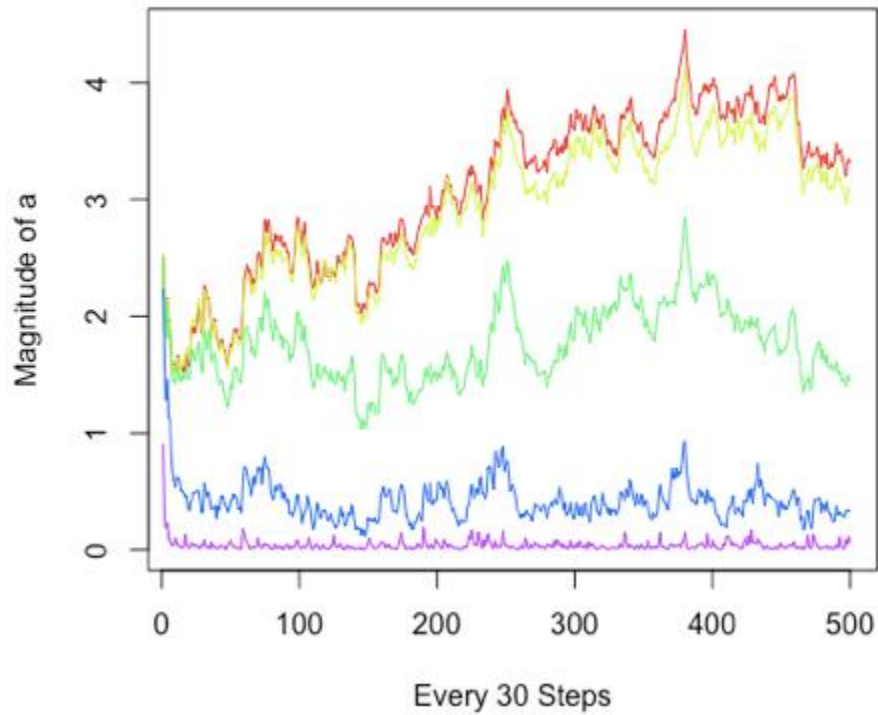
Libraries used : Caret , Rcurl

Held-out accuracy for epoch, plotted every 30 steps for each of 5 regularization constants:
Sample size is 50 as per HW guidelines

Magnitude [Sqrt(a^T.a)] of co-efficient vector, plotted every 30 steps for each of 5 regularization constants. Sample size is 50 as per HW guidelines

Accuracy and magnitude for Evaluation data (E) at the end of training for every lambda (rounded to 3 decimal places):

Lambda Value	Accuracy	Magnitude: Sqrt(a^{T}a)
1e-04	80.6	1.997
1e-03	80.972	1.885
1e-02	80.330	1.373
1e-01	78.55	0.675
1	76.711	0.325

We decided to use a lambda of 1e-02. This is the best value of lambda to use because out of the 3 values that have close enough accuracies (80%+), it has the lowest magnitude. The next value down is 1e-01: while it has a much lower magnitude, it also has a significant decrease in accuracy. Using a lambda of 1e-02, we were able to refit our SVM on the 90% of the data, and then **achieve an accuracy of 81.204%** using the test set (10%)

 Additional analysis:

As the number of samples held-out for epoch is just 50 (around 2%) of the data, the plot for accuracy is not quite smooth. We tried an experiment to increase the number of samples held out at epoch to 5000 and found that the accuracy plots are a lot smoother. A sample accuracy plot with increased sample size is shown below