CSL603– Machine Learning
Lab 4

Due on 20/11/2018 11.55pm

Instructions: Upload to your moodle account one zip file containing the following. Please do not submit hardcopy of your solutions. In case moodle is not accessible email the zip file to the instructor at ckn@iitrpr.ac.in. Late submission is not allowed without prior approval of the instructor. You are expected to follow the honor code of the course while doing this homework.

1. You have to work individually for the lab.
2. This lab must be implemented in Matlab/Python.
3. A neatly formatted PDF document with your answers for each of the questions in the homework. You can use latex, MS word or any other software to create the PDF.
4. Include a separate folder named as 'code' containing the scripts for the homework along with the necessary data files. Ensure the code is documented properly.
5. Include a README file explaining how to execute the scripts.
6. Name the ZIP file using the following convention rollnumber(s)hwnumber.zip

Naïve Bayes Classifiers

We will be using a subset of 2005 TREC Public Spam Corpus [1] containing 5000 training examples and 1000 test examples. This dataset is available in the zip folder of this lab (*nbctrain* and *nbctest*). Each line in the train/test files represents a single email in the following format.

email-id class-label word count word count …

The *email-id* is of the format /xxx/yyy. The *class-label* is either *spam* or *ham* (non-spam). *word count* pair indicate the number of occurrences of the word in the email. The data has been preprocessed to remove non-word characters such as punctuation marks.

- Compute the prior probabilities $P(spam)$ and $P(ham)$ using the training data
- Determine the vocabulary and compute the conditional probabilities $P(x_i/spam)$ and $P(x_i/ham)$ using the m-estimate, where $m = |vocabulary|$ and $p = 1/|vocabulary|$. Which are the 5 most frequently words indicative of a spam and a ham email?
- Use these probabilities to classify the test data and report the accuracy. What difficulty do you face when computing the posterior probabilities? What do you propose to overcome this problem?

- Vary the m parameter and study the changes in the accuracies as a function of m. What assumptions are we making when the value of m is very large compared to it being very small?
- Now that you have built a classifier that can automatically detect spam emails, how would you modify your emails to beat the classifiers?

[courtesy: Pedro Domingos]

K-Means Clustering

Using a k-means clustering implementation of your choice, perform k-means clustering on the MNIST hand written digits' dataset. Indicate in the report the source of your k-means clustering implementation. Perform clustering with k=10. Suppose we were to label each cluster with the most frequently occurring digit, what is the classification accuracy? What are the kinds of misclassifications (confusion matrix)? Suppose we were to increase k to 15, some of the digits which were previously represented as a single cluster will split into multiple clusters. Which are the digits that get split further? Now if we were to reduce k to 5, some of the clusters will be combined. Do your new clusters make any sense? For example, do you observe that clusters with digits 7 and 1 get combined? Discuss your observations.

An important aspect of machine learning is reproducibility of the results presented in a paper/report. Therefore, we will run your code to see if the results are closely matching with what you have presented in the report. Any deviation beyond a reasonable threshold will be considered as fudging of results and will invite severe penalty.