MACHINE LEARNING ASSIGNMENT 2

In this assignment you will build a multilayer neural network and classify whether a given message is SPAM or HAM.

The data can be downloaded from:
https://drive.google.com/file/d/1apNzfUMLA9rKRiOe_T7EvOkMffd8EBAy/view?usp=sharing
The dataset has 5574 messages, each annotated as SPAM or HAM.

Pre-process the data: (i) Break each message into tokens (any sequence of characters separated by blanks, tabs, returns, dots, commas, colons and dashes can be considered as tokens) (ii) Remove a standard set of English stopwords, (iii) Apply Porter stemming.

Consider 80% of the dataset (randomly selected) as training data, and the rest 20% as test set.

Compute the set of distinct tokens in the dataset (denoted as V). Represent each message as a (|V| x 1) vector, where each entry *j* is 1 or 0 depending on whether token *j* is present in the message. This is your input representation and should be fed into the input layer. This representation is usually referred to as 'one hot encoding'. *If you cannot manage a network with all distinct tokens, you can consider the most frequent 2000 tokens only.*

Part (A)

You should use two hidden layers of 100 neurons and 50 neurons respectively, and the output layer will have 1 neuron. The activation value of the output layer indicates the likelihood that the message is SPAM / HAM. Use a suitable threshold value, and classify a message as SPAM if the score is above threshold or HAM if it is below threshold.

Use Stochastic Gradient Descent (SGD) as the optimisation algorithm and **squared error function** as the optimisation function. Do a random initialisation of the weights. Use learning rate 0.1.

Part (A1): Use **sigmoid function** as the non-linear activation function of a neuron.
Part (A2): Use **tanh function** as the non-linear activation function.

For both parts:

- Continue the experiment till in-sample error becomes very low.
- Plot in-sample error (using training set) and out-of-sample error (using test set) of the model after every iteration.
- Do you observe any optimal value for the number of iterations?

Part (B)

Here consider a output layer of 2 neurons (rest of the neural network architecture remains the same as in part A1). There will be a **softmax function**, which will convert the outputs of the neural network in each neuron to the probability of a message being SPAM or HAM. Based on the higher probability you will classify a message to its class.

Use Stochastic Gradient Descent (SGD) as the optimisation algorithm and **squared error function** as the optimisation function. Do a random initialisation of the weights. Use learning rate 0.1.

Using this approach plot the in-sample error (using training set) and out-of-sample error (using test set) of the model after every iteration. Continue the experiment till in-sample error becomes very low.

Submission instructions:

For each part, you should submit the source code and all result files. Write a separate source code file for each part. You should include a README file describing how to execute each of your codes, so that the evaluators can test your code.

You can use C/C++/Java/Python/Matlab for writing the codes, but you cannot use any standard library implementation of neural networks. Also you should not use any code available on the Web. We will use plagiarism detection tools over the submitted source codes. Submissions found to be plagiarised will be awarded zero.

Along with the source codes and the results, you should submit a report (pdf or Word) including the following:

- The plots as described above. You can use any standard plotting tool / library to generate the plots. The data files and the scripts (if any) used to generate the plots should be included in your submission.
- Optimal number of iterations
- Which of the neural network architectures performs the best?

All source codes, data and result files, and the final report must be uploaded via the course Moodle page, as a single compressed file (.tar.gz or .zip). The compressed file should be named as: {ROLL_NUMBER}_ML_A2.{EXTENSION}.
Example: If your roll number is 16CS60R00, then your submission file should be named as 16CS60R00_ML_A2.tar.gz or 16CS60R00_ML_A2.zip

Submission deadline: April 12, 2018 (hard deadline)

For any questions about the assignment, contact the following TAs:
1. Soumya Sarkar (portkey1996 [AT] gmail [DOT] com)
2. Soumajit Pramanik (soumajit.pramanik [AT] gmail [DOT] com)