DSE 210: Probability and statistics **Winter 2015**

Worksheet 5 — Generative models 1

1. *Text classification using multinomial Naive Bayes.*

   (a) For this problem, you'll be using the *20 Newsgroups* data set. There are several versions of it on the web. You should download "20news-bydate.tar.gz" from

   `http://qwone.com/~jason/20Newsgroups/`

   Unpack it and look through the directories at some of the files. Overall, there are roughly 19,000 documents, each from one of 20 newsgroups. The label of a document is the identity of its newsgroup. The documents are divided into a training set and a test set.

   (b) The same website has a processed version of the data, "20news-bydate-matlab.tgz", that is particularly convenient to use. Download this and also the file "vocabulary.txt". Look at the first training document in the processed set and the corresponding original text document to understand the relation between the two.

   (c) The words in the documents constitute an overall vocabulary V of size 61188. Build a multinomial Naive Bayes model using the training data. For each of the 20 classes $j = 1, 2, \ldots, 20$, you must have the following:

   - π_j, the fraction of documents that belong to that class; and
   - P_j, a probability distribution over V that models the documents of that class.

   In order to fit P_j, imagine that all the documents of class j are strung together. For each word $w \in V$, let P_{jw} be the fraction of this concatenated document occupied by w. Well, almost: you will need to do smoothing (just add one to the count of how often w occurs).

   (d) Write a routine that uses this naive Bayes model to classify a new document. To avoid underflow, work with logs rather than multiplying together probabilities.

   (e) Evaluate the performance of your model on the test data. What error rate do you achieve?

   (f) If you have the time and inclination: see if you can get a better-performing model.

   - Split the training data into a smaller training set and a validation set. The split could be 80-20, for instance. You'll use this training set to estimate parameters and the validation set to decide between different options.
   - Think of 2-3 ways in which you might improve your earlier model. Examples include: (i) replacing the frequency f of a word in a document by $\log(1 + f)$, (ii) removing stopwords; (iii) reducing the size of the vocabulary; etc. Estimate a revised model for each of these, and use the validation set to choose between them.
   - Evaluate your final model on the test data. What error rate do you achieve?