Lecture "Foundations: Statistical Classification"
Prof. Dr. D. Klakow

Due: Friday 2005-12-16

Exercise

1. *"Spam, Spam, Spam!"* Unsolicited email is commonly referred to as "spam". Your task is to build a Naïve Bayes classifier that can automatically classify a given document (such as an incoming email) using the two classes `SPAM` and `HAM` (= not spam). Try out one of the two following techniques:

   (a) *[For programmers.]* Download the **LINGSPAM** dataset from `http://www.aueb.gr/users/ion/data/lingspam_public.tar.gz` and read the documentation. Each of the ten folders (`part1`, ..., `part10`) in `lingspam_public/bare` contains both spam and non-spam messages (where the class is indicated by the filename). Extend your program from last week's exercise to build a simple Bayesian classifier using word probabilities to find the most likely class. Train it on 90% of the data (= the first 9 folders), and use the final one (`part10`) for evaluation.

   (b) *[For non-programmers.]* Download the **SPAMBASE** dataset from the *UCI Machine Learning Repository* at `http://www.ics.uci.edu/˜mlearn/databases/`. Read the documentation, look at the data format and make sure you understand what the numbers, rows and columns mean. After converting the format appropriately by adding column headers, use the **Weka** machine learning toolkit[1] to load the dataset, choose `SimpleNaiveBayes` as the classifier to be used for your study, and run an experiment evaluating the task performance of Naïve Bayes on the SPAMBASE dataset, splitting your dataset into 90% for training and 10% for testing.
   Hint: If you would like more background on how to use Weka, the book by Witten and Frank (2005), *Data Mining*, is highly recommended.

[1] `http://www.cs.waikato.ac.nz/˜ml/weka/index.html`

2. The following (real) report gives details of the performance of a spam filter running at a university:

   ```
   STATISTICS REPORT FOR SPAMASSASSIN RULESET V2.64
   Classification success on test corpora, at default threshold:
   # SUMMARY for threshold 5.0:
   # Correctly non-spam:  20976  34.74% (99.80% of non-spam corpus)
   # Correctly spam:      37062  61.38% (94.15% of spam corpus)
   # False positives:        43  0.07%  (0.20% of nonspam)
   # False negatives:      2304  3.82%  (5.85% of spam)
   ```

   What is the Accuracy (total percentage of correct decisions) of the classifier?

3. The very first generation of spam fighting tools did not use any machine learning, but simple keyword-based filters.

   (a) Based on looking at some spam messages in the corpus, pick some words that you think are typical for spam. Then check if your indicator terms are useful by searching whether they only occur in SPAM documents, or also in HAM documents.
   Hint: In UNIX, you can make good use of the commands **tr** and **grep** to avoid programming for this part.[2]

   (b) Based on this evidence, what is your intuition about how well simple keywoard-based spam filter would do in general compared to your Naïve Bayes classifier developed in (1.)?

[2] If you don't know these commands, consult the online manual, e.g. "`man wc`".