

Practical Exercise 1

For this first practical, it is your task to write a program that is able to distinguish which language a given document is written in. To do this, we have set up a corpus containing 10x3 files of texts in English (*.en), Spanish (*.es) and French (*.fr). These documents are located in /g/150TP/files/PP1/.

You should do this by two different methods, described below.

- **Method 1:** Letter frequency.

Here, the learning is done by counting the frequency of every letter. There is a first phase in which the learning program is trained i.e., frequencies are computed for some texts for every language. More concretely, write a program “`train <language> <files>`” where `<language>` is one of {en,es,fr}. Your program should count the frequencies of the 26 letters in the English alphabet over the files specified in `<files>`. Train your program with the first 5 texts in the corpus (for each language). That is, use files `t0.xx`, `t1.xx`, `t2.xx`, `t3.xx`, `t4.xx` to train your program for language `xx` (with `xx=en,es,fr`).

The second phase consists in, for every new document, compute its letter frequency and classify according to which the closest language is. As a notion of distance, you can use the sum of squared differences between the frequencies of letters obtained during the training phase and the frequencies in the current document. More concretely, write a program “`test <files>`” that classifies the texts in `<files>`. Use your `test` program to check that the remaining part of the corpus is classified correctly (use files `t5.xx`, `t6.xx`, `t7.xx`, `t8.xx`, `t9.xx` with `xx=en,es,fr`).

- **Method 2:** Invent your own scheme.

Here, you should come up with a simple ad-hoc scheme, using knowledge of the three languages discussed. Write a program “`classify <files>`”. It should not take more than a few lines of code (or shell script).

You can use any programming language that you like and think is appropriate for this task.

Please submit your solution electronically to marias@eecs.tufts.edu before **Wednesday, 31st January 2001, 5pm**. You should include in your message:

- Clear, comprehensive documentation on how your programs work, including results of the tests performed by both methods.
- Source code: please attach 3 separate code files for `train`, `test` in method 1 and `classify` in method 2.