T-61.181 Special course in Information Science I
Autumn 2001: Independent Component Analysis

Instructions for the second project work

December 3, 2001

1 General

To pass the course, every participant must complete two project works. The first project is a relatively simple assignment on some theoretical aspects of standard ICA, while the second one is a more extensive assignment on the extensions and practical aspects of ICA. Those who wish to suggest their own project also have to do the first project as indicated here, and their own project then replaces the second one. **The deadline for returning both project works is January 31, 2002.**

If you have any questions on the project, you can contact the assistant `antti.honkela@hut.fi`.

2 Reporting

You must return a written report in which you provide justified answers to the questions in the assignment. Explain what you have done and why. The report should be about 2–4 pages long. The complete, sufficiently commented source code should be included as an appendix in the report. The report is to be returned to the assistant.

3 Tools

You are free to use any numerical software you want. We strongly recommend using Matlab as there are existing implementations of all the algorithms used in this work.

4 Assignment

There is an own dataset for every student at the web page of the course (`http://www.cis.hut.fi/Opinnot/T-61.181/`). Your task is to study this dataset with some known ICA algorithms.

The data sets are given in Matlab .mat files. The file contains a single 10×50000 matrix `data`. The 10 rows of the matrix are all time series of audio data sampled at 8000 Hz. You can verify this by playing some of them in Matlab using the `sound` function. You can also try saving the tracks to the disk using e.g. `wavwrite`.

Each of the rows of the data matrix is a noisy mixture of an unknown number of independent source signals. Your first task is to find the number of independent components in the mixture. Then, using **two** of the given ICA algorithms (see Sec. 5), try to separate the signals. Experiment with different parameter settings of the algorithms. You should, for instance, try different preprocessing methods that either use or ignore the known number of independent components. What kind of sources do the methods find, if you try to extract more components than there really are? Also experiment with different nonlinearities in the algorithm, is possible. Which algorithm is, in terms of separation results and/or speed, the best for the task? Hide part of the data and check how much of the data do the different algorithms need to properly separate the data.

Estimate and plot the distributions of the original mixtures and the separated sources. How do they differ? Also estimate the kurtoses. When reducing the amount of data, what kind of sources do the algorithms miss first?

5 Algorithms

The ICA algorithms used in this work are FastICA, JADE, and TDSEP. You must choose two of the three algorithms to use in your work. Matlab implementations of all the algorithms are available at the home page of the course. The TDSEP algorithm is not described in the book, but you can find more information on it in [1]. (There is a link to the article at the web page of the course.)

References

[1] Andreas Ziehe and Klaus-Robert Müller. TDSEP – an efficient algorithm for blind separation using time structure. In *Proc. Int. Conf on Atrificial Neural Networks (ICANN'98)*, pages 675–680, Skövde, Sweden, 1998.