Independent Component Analysis

In this exercise the *Infomax Principle* for Independent Component Analysis (ICA) is implemented using gradient based learning. The files `sound1.dat` and `sound2.dat` in `sounds.zip` contain recordings of two acoustic sources. Your program should contain the following steps:

5.1 Initialization (2 points)

(a) Load the sound files. Each of the $N = 2$ sources is sampled at at 8192 Hz and contains $p = 18000$ samples.

(b) Create a random (& invertible) $N \times N$ mixing matrix $\mathbf{A}$ and mix the sources: $\mathbf{x}^{(\alpha)} = \mathbf{A}\mathbf{s}^{(\alpha)}$

(c) Remove the temporal structure by permuting the columns of the $N \times p$ matrix $\mathbf{X}$ randomly.

(d) Calculate the correlations between the sources and the mixtures: $\rho_{s_i,x_j} = \frac{\mathrm{cov}(s_i,x_j)}{\sigma_{s_i}\sigma_{x_j}}$

(e) Center the data to zero mean.

(f) Initialize the unmixing matrix $\mathbf{W}$ with random values.

5.2 Optimization (4 points)

Implement a *matrix version* of the ICA *online* learning algorithm. For $\hat{f}$ use the logistic function (see lecture notes). This should reduce your code for this part to a few lines with one loop (over the samples). Implement two variants of this learning algorithm:

(a) Compute the update matrix $\Delta\mathbf{W}$ using the "regular" gradient.

(b) Compute the update matrix $\Delta\mathbf{W}$ using the *natural gradient* as described in the lecture notes.

(c) Choose a suitable learning rate ε and apply both versions to the data to unmix the sources.

5.3 Results (4 points)

(a) Plot & Play (i) the original sounds (e.g. use `scipy.io.wavfile` to save playable files), (ii) the mixed sources (before and after the data permutation), and the recovered signals (estimated sources) $\hat{\mathbf{s}} = \mathbf{W}\mathbf{x}$ using the unpermuted data.

(b) Calculate the correlations (as above) between the true sources and the estimations.

(c) For every 1000th update, plot $||\Delta\mathbf{W}||_F^2 := \sum_{i=1,j=1}^{N}(\Delta w_{i,j})^2$ to compare the convergence speed for the two gradient methods. Whiten your data before applying ICA and compare the learning speeds again. Describe the differences between the two variants of the learning algorithm.

(d) Plot the density of the mixed, unmixed, and true signals & interpret your results.

total: 10 points