The Neural Code

Summer term 2022, Christian Leibold, Yuk-Hoi Yiu

Problem Set 9 — **Independent Component Analysis** — **12.07.2022**

Due date: 21.07.2022

---

1. (Entropy Minimization) The file `X_q1.txt` contains a data set X, which is generated by a process

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} = AS,$$

where A is a linear transformation that mixes different independent components of S to produce X. The goal of ICA is to recover the independent sources S just by observing X, without knowing A beforehand.

(a) One common preprocessing step for ICA is whitening, which transforms the data to unit covariance. Generate a "white" version $\tilde{X}$ of the data X by

$$\tilde{X} = B\, D^{-1/2}\, B^T X,$$

where B is the orthogonal matrix of eigenvectors of the covariance matrix of X, and D is the diagonal matrix of its eigenvalues.

(b) Consider a rotation matrix $R(\theta)$

$$R(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

Rotate the whitened data $\tilde{X}$ by computing $Y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = R\tilde{X}$ for θ from $0°$ to $360°$ with a step size of $0.5°$. Verify the rotation results by plotting some rotated examples.

(c) Compute the sum of marginal entropies $H(Y_1) + H(Y_2)$ for all the θ above. For that you could use histograms to estimate the probability distributions $p(Y_i)$. Plot the $\sum_i H(Y_i)$ against θ and interpret the result.

(d) Kurtosis is a measure for non-Gaussianity. It has negative values for sub-Gaussian ("flatter") distributions, positive values for super-Gaussian ("spikier") distributions and a value of 0 for a Gaussian distribution. Use `scipy.stats.kurtosis` to compute the kurtosis K_i of Y_i for every θ and plot the the sum of the kurtosis values $K_1 + K_2$ against θ. Where is the peak and why is it there?

(e) Obtain the $R_0 = R(\theta_0)$ from your plot in (c), where θ_0 minimizes the sum of margiunal entropies $H(Y)$. In this case, $R_0\tilde{X}$ should be an estimate of the independent source S. Let $S = R_0\tilde{X}$, find the mixing matrix A. Plot the column vectors of A as arrows (by `numpy.quiver`) overlaid on the data X to verify the result.

2. (Blind Source Separation)

The file `X_q2.txt` contains three audio signals $X(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix}$ with a sampling rate of $44100s^{-1}$ and duration of $10s$. The audio signals are part of the open source dataset Children's Song Dataset from KAIST Music and Audio Computing Lab.

(a) Use the function `scipy.io.wavfile.write('Xi.wav', 44100,` x_i`)` for $i = 1, 2, 3$ to generate and play the audio files. There should be multiple songs being sung simultaneously.

(b) Perform ICA with `sklearn.decomposition.FastICA()` and retrieve the source signals

$$S(t) = \begin{bmatrix} s_1(t) \\ s_2(t) \\ s_3(t) \end{bmatrix}$$

by the `fit_transform()` function.

(c) Use `scipy.io.wavfile.write('Si.wav', 44100, (`s_i `/` s_i`.max()*10000).astype(numpy.int16)` `)` to save the audio signals as .wav files and listen to them again. The voices should be separated now.