
TP n°2

Séparation de sources

2^e partie

2. Séparation de sources

Les fichiers pour coder les fonctions demandées sont présents dans la deuxième archive à l'adresse ci-dessous :

<http://gjankowiak.github.com/files/opt/2013/tp2bis.tar.gz>

Cette archive comprend également des corrigés des fonctions pour les questions 2, 5 et 6 de la 1^{re} partie. Vous pouvez les utiliser si vous ne les aviez pas faites.

On demande de remplir le script global `TP2bis_NomsBinome.m` (lire attentivement les indications déjà données dans le script).

1. Utiliser la fonction `wavread` pour extraire les 4 signaux `micro1.wav` à `micro4.wav`. Lire et comprendre le fonctionnement de la fonction `Decorrele` fournie. Calculer la moyenne et la variance des signaux, et utiliser la fonction `Decorrele` pour obtenir 4 signaux décorrelés et de variance 1.
2. Programmer une fonction `OptimiseKurtosis` qui prend en arguments une matrice D (qui représente n signaux décorrelés, de variance 1) et les arguments ε , tol , $p0$, et `recherche1D` comme dans `DescenteGradient`, et qui renvoie un signal qui maximise $|H|$ (cf. TD) sous la contrainte de variance 1. La tester sur les signaux décorrelés de la première question. Pour lire le son associé, utiliser `wavwrite` ou `sound`, en n'oubliant pas de lui donner une variance similaire à celle des signaux initiaux (question 1).
3. En s'inspirant de la structure de la fonction `Decorrele`, programmer une fonction `SepareSources` qui prend en entrée n signaux (sous la forme d'une matrice) dont on suppose qu'au plus un a le kurtosis d'une gaussienne, ainsi que les arguments ε , tol , $p0$, et `recherche1D` comme dans `DescenteGradient`, et qui retourne les n sources séparées.
L'utiliser avec les 4 signaux de départ pour récupérer les 4 sources.
4. S'il vous reste du temps, finissez les questions de la 1^{re} partie que vous n'avez pas pu terminer.

3. FastICA

Programmez la méthode FastICA présentée dans la fin du TD pour obtenir une autre manière de séparer les sources.