TP : Décliquage d'un signal audio

Cédric Févotte

Ce TP est réalisé en MATLAB. Un rapport présentant conclusions expérimentales et éléments de discussions, illustré de figures et tableaux numériques, est demandé pour chaque binôme ou trinôme. Les fichiers source MATLAB ayant permis la production des résultats devront être fournis en annexe du rapport. Le rapport, accompagné du fichier audio restauré produit en Section 3, devront être téléversés dans Moodle.

1 Modélisation AR d'un segment audio

1.1 Chargement et visualisation des données

Télécharger le fichier `segment.dat` sur Moodle (parcours IATI, section Audio et Musique). Il s'agit d'un court segment audio (non-altéré) avec lequel nous allons travailler dans la première partie du TP. Un cas de débruitage en situation réelle sera traité en Section 3.

Étape 1. Charger le signal avec la commande `load`. Visualiser le signal avec la commande `plot`. On notera $\mathbf{s} = [s_1, \dots, s_L]^T$ le signal.

Étape 2. Calculer à l'aide de la fonction `fft` le *spectre de puissance* (appelé aussi *périodogramme*) du signal, donné par :

$$S_k = \frac{1}{L} \left| \sum_{t=1}^{L} s_t \, \mathrm{e}^{-j\,2\pi\,\frac{(k-1)}{L}\,(t-1)} \right|^2, \quad k = 1, \dots, \frac{L}{2} + 1.$$

Le périodogramme est essentiellement le module au carré de la transformée de Fourier discrète (DFT). On n'en conserve que les $\frac{L}{2} + 1$ premiers coefficients par symétrie hermitienne de la DFT appliquée à des signaux réels.

```
>>   S = abs(fft(s)).^2/L;
>>   S = S(:,1:L/2+1);
```

Afficher le logarithme du spectre de puissance.

1.2 Étude de l'adéquation d'un modèle AR au signal observé

Étape 1. Déterminer avec la commande `aryule` les paramètres d'un modèle AR d'ordre P tel que

$$s_t = \sum_{p=1}^{P} a_p \, s_{t-p} + e_t, \quad t = P+1, \dots, L,$$

où e_t est un bruit blanc de variance v_e. On notera $\mathbf{a} = [1, -a_1, \dots, -a_P]$.

```
>>   P = 40;
>>   [a,v_e] = aryule(s,P);
>>   a = a.';
```

Étape 2. Soit $A(z) = 1 - a_1\, z^{-1} - a_2\, z^{-2} \dots - a_P\, z^{-P}$ la transformée en z du filtre AR. Calculer maintenant la *densité spectrale de puissance* (DSP) du modèle estimé, donnée par :

$$D_k = \frac{v_e}{\left| A\left(\mathrm{e}^{j\,2\pi\,\frac{(k-1)}{L}}\right)\right|^2} = \frac{v_e}{\left|1 - \sum_{p=1}^{P} a_p \, \mathrm{e}^{-j\,2\pi p\,\frac{(k-1)}{L}}\right|^2}, \quad k = 1, \dots, \frac{L}{2} + 1.$$

On peut à nouveau utiliser la commande `fft`, appliquée cette fois au vecteur $\mathbf{a}$.

```
>>   D = v_e./abs(fft(a,L)).^2;
>>   D = D(:,1:L/2+1);
```

Étape 3. Le périodogramme est un estimateur de la DSP. On peut donc les comparer pour juger de la justesse du modèle pour le signal observé. Tracer les logarithmes du périodogramme et de la DSP calculés précédemment sur la même figure, commenter. Par ailleurs, reconstruire l'erreur de prédiction $\hat{e}_t = s_t - \sum_{p=1}^{P} a_p\, s_{t-p}$ à l'aide de la commande `filter`.

```
>>  err = filter(a,1,x);
```

Afficher l'erreur de prédiction, calculer sa variance et son auto-corrélation (fonction `xcorr`). Le résultat s'accorde-t-il entièrement avec la théorie ? Que doit-on en déduire ?

Répéter les étapes précédentes avec d'autres valeurs de P et analyser l'influence de P.

2 Décliquage d'un segment audio

2.1 Création d'un clic

Créer un signal $\mathbf{x}$ artificiellement altéré en ajoutant au signal $\mathbf{s}$ un défaut de durée L_{al} et commençant à l'échantillon t_0 :

```
>>  x = s;
>>  t0 = 201; L_al = 200; amp = 0.5;
>>  x(t0: t0+L_al-1) = s(t0: t0+L_al-1) + amp * randn(L_al,1)
```

Calculer un filtre AR d'ordre P modélisant le signal $\mathbf{x}$ et reconstruire comme précédemment l'erreur de prédiction $\hat{e}_t$. Afficher l'erreur, calculer sa variance et son auto-corrélation. Commenter.

Implémenter un détecteur de clics en appliquant un *test du 3 σ* à $\hat{e}_t$: [1]

```
>>  detecteur = err > 3*sqrt(v_e);
```

Afficher `detecteur` et `x` sur la même figure, commenter. Répéter l'expérience en faisant varier la valeur des paramètres `L_al, t0, amp`, analyser les résultats obtenus.

2.2 Interpolation

Nous allons maintenant interpoler le segment noté $\mathbf{s}_{al}$, entre le segment $\mathbf{s}_{av}$ avant la zone altérée et le segment $\mathbf{s}_{ap}$ après la zone altérée. On utilisera la méthode itérative de minimisation du critère des moindres carrés de l'erreur quadratique vue présentée en cours, reposant sur une modélisation AR du signal.

Étape 1. Initialiser le segment recherché $\hat{\mathbf{s}}_{al}$, de dimension L_{al}, avec des zéros (commande `zeros`).

Étape 2. Ajuster un filtre AR d'ordre P au vecteur $\hat{\mathbf{s}} = [\mathbf{s}_{av}^T,\ \hat{\mathbf{s}}_{al}^T,\ \mathbf{s}_{ap}^T]^T$.

Étape 3. En utilisant la commande `toeplitz` construire la matrice

$$\mathbf{A} = \begin{bmatrix} \tilde{\mathbf{a}} & 0 & 0 \\ & \ddots & \\ 0 & 0 & \tilde{\mathbf{a}} \end{bmatrix},$$

où $\tilde{\mathbf{a}} = [-a_P, \ldots, -a_1, 1]$ est le vecteur de coefficients AR "retourné".

```
>>  R = zeros(1,L);
>>  R(1:P+1) = a(end:-1:1);
>>  C = zeros(L-P,1);
>>  C(1) = a(P+1);
>>  A = toeplitz(C,R);
```

Étape 4. Définir les matrices $\mathbf{A}_{av}$, $\mathbf{A}_{al}$, $\mathbf{A}_{ap}$ telles que $\mathbf{A} = [\mathbf{A}_{av},\ \mathbf{A}_{al},\ \mathbf{A}_{ap}]$.

1. Si $x \sim \mathcal{N}(0, \sigma^2)$ alors $\mathrm{Prob}(|x| < 3\,\sigma) = 99.7\%$. En conséquence, une variable présentant un nombre significatif de valeurs supérieures à 3σ ou inférieures à -3σ ne répond vraisemblablement pas à une hypothèse gaussienne.

Étape 5. En utilisant la commande “\” (voir `help mldivide`), calculer le segment interpolé $\hat{\mathbf{s}}_{al}$ défini par

$$\hat{\mathbf{s}}_{al} = -\underbrace{(\mathbf{A}_{al}^T\,\mathbf{A}_{al})^{-1}\,\mathbf{A}_{al}^T}_{\text{pseudo-inverse de } \mathbf{A}_{al}}\;(\mathbf{A}_{av}\,\mathbf{s}_{av} + \mathbf{A}_{ap}\,\mathbf{s}_{ap}).$$

Étape 6. Tracer sur la même figure le signal original $\mathbf{s}$ et le signal reconstruit $\hat{\mathbf{s}} = [\mathbf{s}_{av}^T,\ \hat{\mathbf{s}}_{al}^T,\ \mathbf{s}_{ap}^T]^T$. Calculer et afficher l'erreur de reconstruction $\|\hat{\mathbf{s}} - \mathbf{s}\|_2$, avec la commande `norm`.

Étape 7. Itérer les Étapes 2 à 6 pendant 2 ou 3 itérations afin d'affiner l'estimation de $\hat{\mathbf{s}}$. Afficher le signal reconstruit et l'erreur de reconstruction $\|\hat{\mathbf{s}} - \mathbf{s}\|_2$ à chaque itération. Analyser les résultats obtenus, en répétant l'expérience avec différentes valeurs de t_0, L_{al} et P.

3 Restauration d'un enregistrement réel

Téléchargez le fichier `donnees.wav` sur Moodle. Écoutez, visualisez et restaurez l'enregistrement, en utilisant l'approche suivie en Section 2 (on pourra identifier les clics visuellement, sans utiliser de détecteur). Présentez votre démarche dans le rapport de TP, en indiquant les paramètres de réglage utilisés pour chaque clic (P, L_{al}, L_{av}, L_{ap}). Produisez un fichier audio restauré (au format WAV) qui sera téléversé dans Moodle avec le rapport de TP. Les commandes MATLAB `audioread` et `audiowrite` vous permettront de lire et créer un fichier WAV dans MATLAB. La commande `soundsc` vous permettra de jouer un signal audio dans MATLAB.