Master Mathématiques et Applications, 1-ière année, Aix-Marseille Université

Modélisation en Traitement du Signal, Préparation du TP

Année 2015-16

---

1 Algèbre linéaire

Il s'agit ici de retrouver les résultats du cours sur l'approximation, en prenant comme espace de départ non plus $L^2([0,1])$ mais $E = \mathbb{C}^L$ (ou $E = \mathbb{R}^L$), où L est un entier, très grand, et un sous-espace d'approximation $F \subset E$ de dimension $N < L$, et d'écrire toutes les opérations sous forme matricielle.

On suppose donnée une base $\{\varphi_0, \dots \varphi_{N-1}\}$ de F, et on note $\phi_0, \dots \phi_{N-1} \in \mathcal{M}_{L,1}(\mathbb{C})$ les vecteurs colonne de leurs composantes dans la base canonique. Ainsi, l'élément de matrice ϕ_{mn} représente la m-ième composante du vecteur φ_n.

Soit $\Phi \in \mathcal{M}_{L,N}(\mathbb{C})$ la matrice obtenue par concaténation de $\phi_0, \dots \phi_{N-1}$. Pour les calculs ci-dessous, il suffit de bien écrire explicitement toutes les opérations en faisant intervenir les composantes des vecteurs, pour ensuite obtenir les expressions matricielles. Par exemple, étant donnés deux vecteurs $x, y \in \mathbb{C}^L$, et les colonnes correspondantes $X, Y \in \mathcal{M}_{L,1}(\mathbb{C})$, le produit scalaire s'écrit $\langle x, y\rangle = \sum_{\ell=0}^{L-1} X_\ell \overline{Y}_\ell = Y^* X$, où $Y^* = \overline{Y}^t \in \mathcal{M}_{1,L}(\mathbb{C})$ est l'adjoint (c'est à dire le complexe conjugué de la transposée) de Y.

1. Montrer que la matrice de Gram G de la famille $\{\varphi_0, \dots \varphi_{N-1}\}$ s'écrit sous forme matricielle
$$G = \Phi^*\Phi \ .$$
2. On note $\{\tilde\varphi_0, \dots \tilde\varphi_{N-1}\}$ la base duale (ou biorthogonale) de $\{\varphi_0, \dots \varphi_{N-1}\}$ dans F, et $\tilde\Phi$ la matrice correspondante. Montrer que
$$\tilde\Phi = \Phi\left(\Phi^*\Phi\right)^{-1}$$
3. Soit $x \in E$, soit $y = \Pi_F(x) = \sum_{n=0}^{N-1} a_n\varphi_n \in F$. Soient $X, Y \in \mathcal{M}_{L,1}$ les vecteurs colonnes associés à x et y respectivement, et soit $A \in \mathcal{M}_{N,1}$ la matrice colonne associée. Montrer que
$$A = (\Phi^*\Phi)^{-1}\Phi^* X \ , \qquad \text{et en déduire que} \qquad Y = \Phi(\Phi^*\Phi)^{-1}\Phi^* X \ .$$

2 Premiers pas vers l'implémentation sous Octave/Matlab

1. Trois fonctions Octave/Matlab permettant de générer des matrices Φ telles que décrites ci-dessus sont fournies : `BaseSplineConst.m` génère une famille engendrée par translations de l'indicatrice d'un intervalle (entier), et `BaseSplineAff.m` génère une famille engendrée par translations d'un vecteur dont les composantes sont fonction affine par morceaux de l'indice. `BaseDeBosses` génére des translatées de fonctions gaussiennes.
   - Etudier ces deux fonctions et les tester ; tracer quelques exemples des vecteurs ainsi générés.
   - Vérifier numériquement que les colonnes forment des familles linéairement indépendantes.
   - Tester ces fonctions pour différentes valeurs des paramètres, et calculer la matrice de Gram correspondante. Etudier le conditionnement de cette dernière (utiliser la fonction `cond`, se renseigner).
2.
   - Ecrire une fonction `baseduale.m` permettant de calculer la matrice de la base duale, la tester sur les deux exemples ci-dessus. La syntaxe pourra être de la forme
     `Phitilde = baseduale(Phi);`
   - Tracer quelques exemples de vecteurs des bases duales.

Correction rapide

Algèbre linéaire

1. Calculons l'élément de matrice générique de $\Phi^*\Phi$:
$$(\Phi^*\Phi)_{mn} = \sum_k \overline{\Phi}_{km}\Phi_{kn} = \phi_m^*\phi_n = \langle\varphi_n, \varphi_m\rangle = G_{mn} \ .$$
Sous Octave/Matlab ceci se traduit par l'instruction
```
G = Phi'*Phi;
```
2. On a déjà vu que $G = \Phi^*\Phi$, et donc $H = G^{-1} = (\Phi^*\Phi)^{-1}$. Par ailleurs, en composantes, on a pour tout $k = 0, \ldots L-1$
$$\tilde{\Phi}_{km} = \sum_{n=0}^{N-1} H_{nm}\Phi_{kn} = (\Phi H)_{km} \ ,$$
donc
$$\tilde{\Phi} = \Phi(\Phi^*\Phi)^{-1} \ .$$
Sous Octave/Matlab ceci se traduit par exemple par l'instruction
```
Phitilde = Phi*inv(Phi'*Phi);
```
Remarque : il est aussi possible d'écrire cela sous forme condensée, à savoir
```
Phitilde = Phi/(Phi'*Phi);
```
qui est parfois plus efficace (voir la documentation).
3. Soit $c_m = \langle x, \varphi_m\rangle$, et soit $C \in \mathcal{M}_{N,1}$ le vecteur colonne composé des nombres c_n. Vérifions tout d'abord que C s'écrit $C = \Phi^*X$. Pour cela, calculons en composantes
$$c_m = \sum_{k=0}^{L-1} x_k\overline{\Phi}_{km} = \sum_{k=0}^{L-1}(\Phi^*)_{mk}x_k = (\Phi^*X)_m \ .$$
Sous Octave/Matlab ceci se traduit par l'instruction
```
C = Phi'*X;
```
On a maintenant
$$a_n = \sum_m H_{nm}c_m = (H\Phi^*X)_n$$
donc $A = H\Phi^*X = (\Phi^*\Phi)^{-1}\Phi^*X$, ce qui se code par exemple en Octave/Matlab sous la forme `A = inv(Phi'*Phi)*Phi'*X;`
Finalement, de $y_m = \sum_k a_k\Phi_{mk}$ on déduit
$$Y = \Phi(\Phi^*\Phi)^{-1}\Phi^*X \ ,$$
expression que l'on peut coder en Octave/Matlab comme
```
Y = Phi*inv(Phi'*Phi)*Phi'*X;
```

Premiers pas vers l'implémentation sous Octave/Matlab

Une question importante est celle du comportement des valeurs propres de la matrice de Gram. On sait qu'elles sont nécessairement positives ou nulles. Ceci étant, même lorsque la plus petite valeur propre est non-nulle, elle peut être assez petite pour que des problèmes numériques interviennent. C'est pour cela qu'il est important de tester le conditionnement.