SI 221

Bases de la Reconnaissance des Formes

TP "Décision Bayésienne"

Laurence Likforman-Sulem

02 Octobre 2008

1. Préliminaires

1.1 Quelques Rappels

• pour lancer matlab, allumer le PC, cliquer sur l'icône matlab située sur le bureau. Puis faire *cd d:\user* pour vous mettre dans un répertoire de travail.

• pour voir les variables et leur dimension taper *whos*. Les variables n'ont pas besoin d'être déclarées au préalable. Les commentaires sont précédés de %
x=[1 2 3] ;   % crée le vecteur ligne x=(1, 2, 3)

• pour effacer les variables taper *clear*.

• utiliser les flèches pour rappeler des commandes

• *help* commande affiche la documentation correspondant à une commande donnée.
Exemple : *help eig*

• Vous pouvez éditer vos commandes matlab dans un m-file *nom.m*. Pour exécuter le programme, taper *nom*.

• mettre un ';' à la fin des commandes (sauf si vous voulez que le résultat s'affiche).

1.2 Liste des opérations ou fonctions dont vous pouvez avoir besoin

• s=[1 2; 2 5] crée la matrice s= $\begin{bmatrix} 1 & 2 \\ 2 & 5 \end{bmatrix}$

• x=[x1;x2;x3], x1, x2, x3 étant des matrices ayant même nombre de colonnes j, est une matrice de j colonnes constituée de la concaténation de x1, x2 et x3.

• r=*sqrt*(s), s étant une matrice, retourne la matrice r constituée de la racine des éléments de s.

• r=*sqrtm*(s), s étant une matrice, retourne la matrice r telle que $s = r\, r^t$

• x=*ones* (1,N) crée le vecteur ligne x=(1,1,.....,1)

• [V, S] = *eig*(s) retourne dans la matrice V les vecteurs propres de s, et dans S la matrice diagonalisée

• fonctions trigonométriques : *cos*, *sin* , *atan*

• x= *randn*(N, d) retourne dans la matrice x , N vecteurs lignes composés de d variables aléatoires indépendantes, centrées sur l'origine et de variance 1 (matrice de covariance égale à l'identité).

• m= *mean*(x) , x étant une matrice, est un vecteur ligne contenant la valeur moyenne de chaque colonne de la matrice x.

• s= *var*(x) , x étant un vecteur, calcule l'estimateur non biaisé de la variance. Voir aussi les fonctions *cov* et *std*.

• c = *cov*(x), x étant une matrice, est la matrice de covariance des échantillons placés en ligne dans la matrice x.

• is = *inv*(s), is est la matrice inverse de s.

• y= *min*(x), x étant une matrice, est un vecteur ligne y constitué des éléments minima de chaque colonne de x.

• la fonction *linspace* permet de créer une grille de points.

• i=*find*(tab(:,2)==val) retourne dans le vecteur i les indices ligne de la matrice tab dont l'élément de 2$^{\text{ème}}$ colonne est égal à val. x=tab(*find*(tab(:,2)==val)) sélectionne dans x les lignes de la matrice tab dont l'élément de 2$^{\text{ème}}$ colonne est égal à val.

1.3 Fonctions d'affichage

• *figure*(n) ; ouvre une nouvelle fenêtre d'affichage de numéro n. Par défaut, on affiche dans la fenêtre 1.

• *plot* (x(:,1), x(:,2), '*');
x étant une matrice avec les échantillons placés en ligne. La commande affiche la deuxième composante des échantillons en fonction de la première. Les points sont affichés à l'écran avec le symbole *.

• *hold on* permet aux commandes graphiques qui suivent de se superposer au graphe existant. *hold off* permet de revenir au mode par défaut où les nouvelles commandes effacent le graphe précédent.

• *mesh* (X, Y, F') ;
où X (resp. Y) est un vecteur de dimension m (resp.n) et F une matrice de dimension n x m . F' est la transposée de F. *mesh* affiche en 3-D les triplets (X(i), Y(j), F(i,j)). On peut modifier l'angle de vue par la fonction *view*.

• *contour* (X, Y, F') ;
Mêmes paramètres que précédemment. *contour* affiche en 2-D les lignes de contours dans les éléments de la matrice F.

2. Génération d'une variable aléatoire gaussienne

Utiliser les fonctions *randn* et *ones* pour générer N échantillons d'une variable aléatoire gaussienne de moyenne 3 et de variance 4. Etudier l'évolution de la moyenne et de la variance empiriques en fonction de N.

3. Génération de vecteurs aléatoires gaussiens

On va générer en dimension 2, un ensemble d'apprentissage constitué de trois classes d'échantillons suivant des lois normales, de vecteurs moyenne et de matrices de covariance donnés.

1) Générer un échantillon de taille N=100 d'un vecteur aléatoire gaussien défini par le vecteur moyenne m=[4 9] et la matrice de covariance égale à l'identité. Afficher les échantillons.

2) Donner l'expression permettant de générer N=100 échantillons d'un vecteur aléatoire de moyenne m=[4 9] et de matrice de covariance diagonale s= [1 0 ; 0 6]
Vérifier votre résultat en utilisant les fonctions *mean* et *cov* et afficher les échantillons.

3) Soit X un vecteur aléatoire gaussien centré de matrice de covariance égale à l'identité. Chercher une transformation linéaire X'= U X qui permette d'obtenir un vecteur aléatoire centré de matrice de covariance Σ .

4) Générer des échantillons dont la matrice de covariance est égale à s= $\begin{bmatrix} 2 & 2 \\ 2 & 5 \end{bmatrix}$

Vérifier votre résultat avec les fonctions *mean* et *cov*.
Calculer l'orientation de l'ellipsoide de Mahalanobis asscocié à s.
Vérifier la relation liant s à sa matrice diagonalisée S_d : $s=V.S_d.V^t$
avec V, matrice des vecteurs propres de s.

5.) Générer dans les matrices x1, x2, x3, 3x100 échantillons de trois vecteurs aléatoires gaussiens. On donne:
m1 = [4 9] s1 = $[2 \quad 2, \ 2 \ \ 5]$
m2 = [8.5 7.5] s2 = $[2 \ -2, \ -2 \ \ 5 \quad]$

m3 = [6 3.5]   s3 = $\begin{bmatrix} 7 & -4, & -4 & 7 \end{bmatrix}$

Afficher l'ensemble des échantillons, chaque classe dans une couleur différente.

4. Courbes d'équidensité

1). Ouvrir une autre fenêtre. Créer une grille de points X(i), Y(j) répartis régulièrement dans l'espace de taille 57x57 entre les valeurs 0.27 et 12.5 pour la première mesure et -2 et 15 pour la deuxième mesure.

2) Pour la classe 1 de la question 3.5, construire la matrice dens1(i,j) contenant la densité de probabilité conditionnelle en chaque point (X(i), Y(j)) de la grille.

3) Afficher les courbes d'équidensités pour la classe 1 à l'aide de la fonction *contour*. Quelle est la forme de ces courbes ?

4). Faire de même pour les deux autres classes. Représenter en 3-D sur la même figure les trois lois de densités conditionnelles (utiliser la fonction *mesh*).
Expliquer l'allure des amplitudes maximales des lois de densités

5. Visualisation des frontières

1) Classer les points (X(i), Y(j)) de la grille précédente. On utilisera une matrice Z(i,j) dont les éléments contiendront l'étiquette (1, 2 ou 3) de la classe obtenue pour chacun des éléments de la grille.

2) Pour faire apparaître les frontières entre classes, appliquer la fonction *contour* à Z^t et aux vecteurs X et Y. Indiquez les régions de l'espace correspondant à chacune des classes ?
On pourra superposer les frontières aux points échantillons à l'aide des fonctions *hold on* et *plot*.

6. Application

Cette application est issue du Cars dataset (collecté par E. Ramos et D. Donoho et entreposé dans la bibliothèque StatLib. Cet ensemble contient environ 400 échantillons représentant des modèles de voitures et 8 variables les décrivent. La dernière variable est la classe des échantillons, c-a-d le continent d'origine du modèle : USA (1), Europe (2), Asie (3).

On veut prédire la classe du modèle (le continent d'origine) à partir de deux variables :

- MPG : miles per gallon relatif à la consommation du véhicule (en position 1)
- weight : le poids de la voiture (en position 5)

Déterminer le vecteur moyenne et la matrice de covariance de chacune des classes. Tracer les frontières entre classes à partir des échantillons fournis dans le fichier *voitures.mat* (*load voitures* renvoie dans la variable *cars* l'ensemble des échantillons).