從機率角度看群組分析
Linear and Quadratic Discrimant Analysis

December 14, 2018

本章探討監督式學習中的群組分析，並從群組界線的後驗機率相等切入，建立群組界線劃分資料空間，以利分群判別。從理論的推敲，到程式碼撰寫，到直接利用 MATLAB 的 Machine Learning 套件，一步步了解問題並學習解決問題的方式。而依假設情況之不同，本章探討線性與非線性的群組空間的界線。

Machine Learning 套件使用、特殊線性方程式的繪圖、MATLAB 矩陣式的計算技巧。

〈本章關於 MATLAB 的指令與語法〉
指令：classify, fitcdiscr, fimplicit, mvnpdf, mvnrnd, predict。

1 基本觀念

後驗機率是運用機率概念解決分群問題最直覺的出發點，但卻是個很不友善的東西。一般而言，並沒有足夠的訊息寫出完整的函數。不過山不轉路轉，有一個可愛又有一點討厭的貝氏定理撐腰，譬如後驗機率 $P(G = k|X)$ 可以改寫爲[1]

$$P(G = k|X) = \frac{P(X|G = k)P(G = k)}{\sum_l P(X|G = l)P(G = l)} \tag{1}$$

其中 $P(X|G = k)$ 表示第 k 組資料發生的機率密度函數，而 $P(G = k)$ 代表每一個群組所佔的比例（發生的機率）。[2]相較於後驗機率 $P(G = k|X)$，這兩個機率函數比較合理的原因在於它們比較容易從已知的資料中估計得到。當然估計過程的假設與計算品質的好壞也間接影響了這個方案的準確度。本章假設 $P(X = \mathbf{x}|G = k) = f_k(\mathbf{x})$ 服從多變量常態（Multivariate Normal Distribution），寫成

$$f_k(\mathbf{x}) = \frac{1}{(2\pi)^{p/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\mu_\mathbf{k})^T\Sigma_k^{-1}(\mathbf{x}-\mu_\mathbf{k})} \tag{2}$$

其中 $\mathbf{x} \in R^p$，$\mu_\mathbf{k}$ 與 Σ_k 分別代表第 k 群資料常態假設的均值與共變異矩陣。這是對於母體的假設，至於眞實的資料是否具備這個分配的特性是需要進一步檢驗的。爲簡化問題的複雜性，更進一步假設所有群組的共變異矩陣（Covariance Matrix）都相等，即 $\Sigma_k = \Sigma, \forall k$。當然這個假設的合理性也是需要從實際的資料中做進一步的檢驗。

當給定已知資料 $(X = \mathbf{x})$，想在 p 度空間中劃分群組的領地，或說在 p 度空間中找出群組間的分界線，[3]譬如，兩個群組 k 與 l 分界線上的資料滿足以下條件:

$$Pr(G = k|X = \mathbf{x}) = Pr(G = l|X = \mathbf{x}) \tag{3}$$

也就是，在資料存在的空間裡，群組 k 與群組 l 出現機率相同的地方。或者說，分割群組 k 與群組 l 的線上的點必須滿足以上的條件。在後驗機率相等 (3) 的群組分界原則下，結合由貝式定理 (1) 與資料的常態分配假設 (2)，分界線的函數可以從以下的轉換得到:

[1]其中 X 代表多變量資料變數，G 是群組變數。

[2]一般稱 $P(X|G = k)$ 爲概似函數，稱 $P(G = k)$ 爲先驗機率。

[3]這裡提到的「分界線」並非指平面空間上的一條線，而是更廣泛的「Separating Hyperplanes」，可能是線、面或更高維度的集合，一般通稱爲 hyperplane 的幾何平面。

$$\begin{aligned}\log\frac{Pr(G=k|X=\mathbf{x})}{Pr(G=l|X=\mathbf{x})} &= \log\frac{f_k(\mathbf{x})}{f_l(\mathbf{x})}+\log\frac{Pr(G=k)}{Pr(G=l)}\\ &= \log\frac{Pr(G=k)}{Pr(G=l)}-\frac{1}{2}(\mu_k+\mu_l)^T\Sigma^{-1}(\mu_k-\mu_l)+\\ &\quad \mathbf{x}^T\Sigma^{-1}(\mu_k-\mu_l)=0 \qquad (4)\end{aligned}$$

這裡巧妙的運用對數轉換（logit transformation）的技巧，並令 log-odds 爲零去除指數，得到一組線性的方程式（請注意：線性關係來自「不同群組有相同共變異矩陣」的假設）。於是從資料的後驗機率相等，得到式 (4) 的線性方程式，也決定群組的分野，當然也可供判斷一筆新資料的究竟落在哪個群組。這個判斷問題寫成以下的最大值問題，也稱爲 Linear Discrminant Analysis (LDA)

$$\begin{aligned}G(\mathbf{x}) &= \arg\max_k \log Pr(G=k|X=\mathbf{x})\\ &= \arg\max_k \log(Pr(X=\mathbf{x}|G=k)Pr(G=k))\\ &= \arg\max_k \ \mathbf{x}^T\Sigma^{-1}\mu_k-\frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k+\log Pr(G=k) \qquad (5)\end{aligned}$$

第一行的意思很直覺；一筆新資料 $X=\mathbf{x}$ 出現在哪一個群組的機率最高？利用 (1) 的關係，去除與 k 無關的分母，變成了第二行。再將 (2) 代入，同樣去除與 k 無關的項目，變成了第三行，稱爲第 k 群的線性判別式函數（Linear Discriminant Function），即

$$\delta_k(\mathbf{x})=\mathbf{x}^T\Sigma^{-1}\mu_k-\frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k+\log Pr(G=k) \qquad (6)$$

在式 (5) 中，除了資料 **x** 已知外，其餘都未知，即便如此，我們仍可以利用已知的資料來估計這些值：譬如 μ_k 用第 k 組的資料的樣本平均值，Σ 可以用各組資料算出來的 Sample Covariance Matrices 的加權平均，$Pr(G=k)$ 則是已知資料中各組數量的比例，即

- $Pr(G=k)\approx\hat{\pi}_k=N_k/N$ ，其中 N_k 代表第 k 組的數量，N 代表所有資料的總數。
- $\hat{\mu}_k=\sum_{group=k}\mathbf{x}_i/N_k$
- $\hat{\Sigma}=\sum_{k=1}^{K}\sum_{group=k}(\mathbf{x}_i-\hat{\mu}_k)(\mathbf{x}_i-\hat{\mu}_k)^T/(N-K)$ ，K 代表群組數。這個估計又稱爲 Pooled within-group covariance matrix。

上述的共變異矩陣來自個群組的共變異矩陣相同的假設。如果個群組的共變異矩陣不同時，則如式 (6) 的線性判別是函數將改寫爲：

$$\delta_k(\mathbf{x}) = -\frac{1}{2}\log|\Sigma_k| - -\frac{1}{2}(\mathbf{x}-\mu_k)^T\Sigma_k^{-1}(\mathbf{x}-\mu_k) + \log Pr(G=k) \tag{7}$$

式 (7) 稱爲二次判別式函數。稱爲二次（quadratic）的原因來自而介於群組 k 與群組 l 間的分界線不是直線，而是變數的二次方，這條分界線寫爲:

$$\{\mathbf{x}|\delta_{\mathbf{k}}(\mathbf{x}) = \delta_{\mathbf{l}}(\mathbf{x})\} \tag{8}$$

若令 $\mathbf{x} = [\mathbf{X_1}\ \mathbf{X_2}]$，則式 (8) 經過妥善地展開、推導爲

$$c = c_1X_1 + c_2X_2 + c_3X_1X_2 + c_4X_1^2 + c_5X_2^2 \tag{9}$$

式 (9) 便是一條平面上的二次曲線。

2 練習

範例 1 下載測試資料 la_1.txt 如圖，這是一組內含兩個已知群組的雙變量資料。想看看 LDA（式 (4)）的效果如何，即畫出所示的 LDA 的分界線。

首先估計出分界線函數 (4) 所需的 $\mu_1, \mu_2, \Sigma, Pr(G=1), Pr(G=2)$:

- 估計之前，先檢視該資料的結構與內容，譬如群組的記號以 0 與 1 表達。
- 當兩個群組的數量相等時，Σ 的估計可以採 $\hat{\Sigma} = (\hat{\Sigma}_1 + \hat{\Sigma}_2)/2$。

```
D=load('la_1.txt')
n=size(D,1); n1=sum(D(:,3)==0); n2=n-n1;  % 總樣本數與兩群組數
C1=D(D(:,3)==0,1:2); % 取得第 3 欄資料等於 0 的第一組資料
C2=D(D(:,4)==1,1:2); % 取得第 3 欄資料等於 1 的第二組資料
gscatter(D(:,1), D(:,2), D(:,3), 'br', 'op') % 畫出群組資料的散佈圖
pi1=n1/n; pi2=n2/n; % 估計先驗機率 P(G=1) 及 P(G=2)
mu1=mean(C1); mu2=mean(C2);
Sigma=(cov(C1)+cov(C2))/2; % 近似的共變異矩陣
```

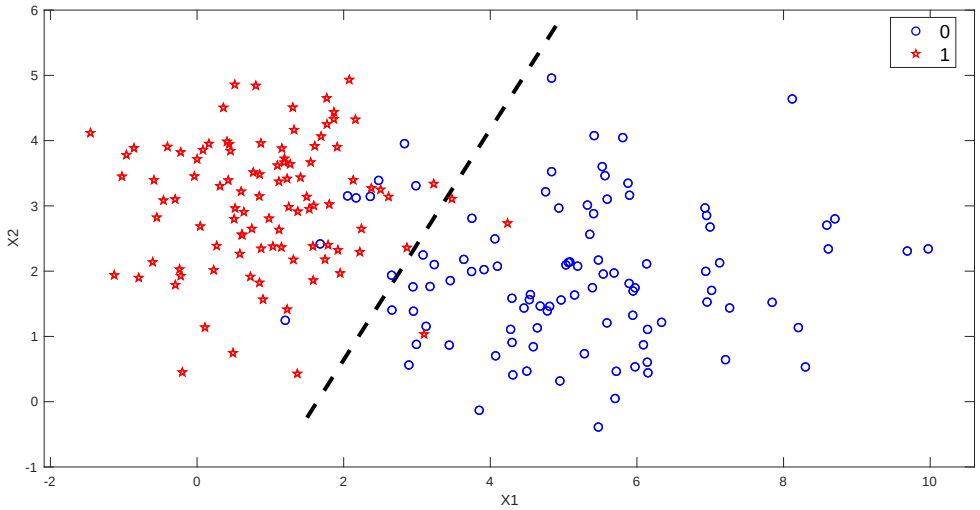

圖 1: LDA 群組分界線

在只有兩個變數的情況下，將式 (4) 中的 $\mathbf{x}^T$ 以 $[x_1\ x_2]$ 代入，改寫爲 $K + [x_1\ x_2]L$，其中常數 K 與向量 L 分別爲

$$K = \log \frac{Pr(G=1)}{Pr(G=2)} - \frac{1}{2}(\mu_k + \mu_l)^T \Sigma^{-1}(\mu_k - \mu_l)$$

$$L = \Sigma^{-1}(\mu_k - \mu_l)$$

於是直線方程式寫成

$$K + L(1)x_1 + L(2)x_2 = 0$$

當 $\mu_1, \mu_2, \Sigma, Pr(G=1), Pr(G=2)$ 以估計值帶入計算 K 與 L 時，便能畫出圖 (1) 所示中間那條直線。[4]上述程式碼擴增下列的直線繪圖:

```
K=log(pi1/pi2)-0.5*(mu1+mu2)*pinv(Sigma)*(mu1-mu2)';
L=pinv(Sigma)*(mu1-mu2)';
f=@(x1,x2) K+L(1)*x1+L(2)*x2;
fimplicit(f,[-1 5], 'LineWidth', 3, 'color', 'black', 'LineStyle', '-')
```

其中 Σ^{-1} 的計算指令採 pinv(Sigma)，即 pseudo inverse，以避免反矩陣不存在時，仍能以估計方式取代。

MATLAB 自 2014 年起將統計工具箱擴增，加入 Machine Learning 功能。以下程式碼展示 MATLAB 提供的做法。

[4]直線方程式的符號 $K + L(1)x_1 + L(2)x_2 = 0$ 來自 MATLAB 手册。

```
D=load('la_1.txt')
n=size(D,1);                                % 總樣本數
g=cell(n,1);                                % 製作群組標示
g(D(:,3)==0)={'Group A'};                   % 第一組資料稱爲 Group A
g(D(:,3)==1)={'Group B'};                   % 第二組資料稱爲 Group B
gscatter(D(:,1), D(:,2), g, 'br', 'op')     % 畫出群組資料的散佈圖
X=D(:,1:2);                                 % 準備資料矩陣：n x 2
Lda = fitcdiscr(X, g);                      % 根據資料與群組類別分群
k = Lda.Coeffs(1,2).Const;                  % LDA 分界線的常數
d = Lda.Coeffs(1,2).Linear;                 % LDA 分界線的線性係數
f=@(x1,x2) k+[x1 x2]*d;                     % 式 (4)
hold on, fimplicit(f,'LineWidth',3), hold off
```

上述程式碼執行結果與圖 1 完全一樣。其中關鍵指令 fitcdiscr 是群組分析器 (classifier)，根據矩陣資料 X 與相對應的群組類別資料 g （可以是數字或文字）計算出群組間分界線的相關係數。讀者可以在命令視窗觀察輸出結果的結構型變數 Lda 的內容，譬如圖 2 所示。本範例使用資料只有兩群，因此在輸出結果能看到分界線的係數 Coeffs 是一個 2x2 的結構型矩陣變數，其 Coeffs(1,2) 或 Coeffs(2,1) 都是指群組 1 與群組 2 的分界線係數。

群組分類器根據已知的群組資料建立分類標準，也就是上述的分界線係數 (即式 (5))，其用途當然是拿來對未知群組別的資料做分群。以下範例展示這個做法。

範例 2 式 (5) 是所謂的 Linear Discriminant Function，可用來判斷新資料的組別。其判別方式必須針對每一個組別分別計算出一個數值（即對數後驗機率值），產生最大值的組別就是判爲該群組。利用式 (5) 即前一個範例的結果，製作一個預測群組別的程式，再試著使用 MATLAB 提供的預測器指令 predict。

假設有 k 個群組，要同時爲每個群組計算式 (5) 的對數後驗機率值，可以將 $p \times 1$ 的向量 μ_k 換成 $p \times k$ 的矩陣 $U = [\mu_1\ \mu_2\ \cdots\ \mu_k]$，最後一項換成 $\pi = [\log Pr(G = 1)\ \log Pr(G = 2)\ \cdots\ \log Pr(G = k)]$，直接套入式 (5) 計算，即 k 個對數後驗機率值記爲 1xk 的向量

$$\mathbf{G} = \mathbf{x}^T \Sigma^{-1} U - \frac{1}{2} U^T \Sigma^{-1} U + \pi$$

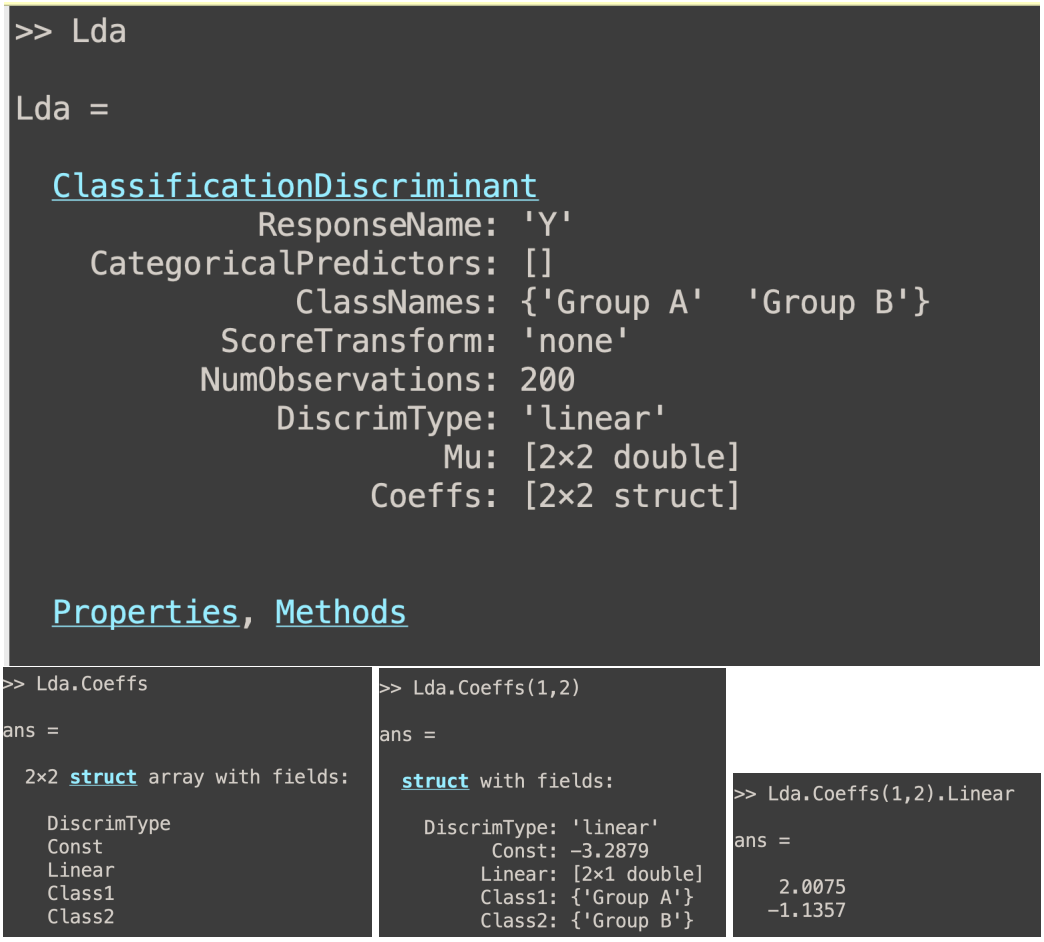

圖 2: LDA 群組分析器 fitcdiscr 的輸出結果。

再利用 max 指令，對向量 **g** 找出最大值出現的項次，最爲群組的判斷。於是分類器的程式碼可以這樣寫（承上一個範例的程式碼）：

```
x=input('Input a new data in the form [x1 x2]:'); % 輸入欲預測的新資料點
MU=[mu1' mu2'];
PI=[log(pi0) log(pi1)];
G=x*pinv(Sigma)*MU- 0.5*diag(MU'*pinv(Sigma)*MU)'+ PI;
[m,i]=max(G); % 判定資料 x 屬於群組 i
```

從 MATLAB 程式設計的角度，最好是運用矩陣的特性，同時計算出所有的數值並放在一個向量裡，不需要採迴圈的方式一個個計算。不過練習之初，倒可以先以迴圈寫寫看，再從迴圈的結構找出矩陣可以運用的地方。

除了以式 (5) 的後驗機率值作爲群組判斷的依據外，也可以利用群組間的分界線作爲資料空間的界線，判斷新的資料點落在哪個空間。讀者可以試著自己做做看。

MATLAB 在 Machine Learning 套件的指令是一整套的。延續前一個範例的分群器指令

fitcdiscr 之後，利用預測器 predict 進行新資料的群組預測。以下程式碼呈現整個資料空間群組預測結果。承接上一個分群器的程式碼:

```
[XX,YY]=meshgrid(-2:0.5:10,-5:0.5:5); % 給定空間範圍內的資料點
xx=XX(:); yy=YY(:); % 矩陣轉爲向量
M=predict(Lda, [xx yy]); % 對所有資料點做群組預測
gscatter(xx, yy, M, 'rb', '..') % 根據群組預測結果 M 塗顏色。
```

執行結果如圖 3 所示。其中紅色與藍色的點是根據每個位置做出群組預測後，畫上去的。當空間資料點給得愈密集時，整個空間變塗滿紅色與藍色。讀者可以試試將 meshgrid 內的向量改爲更密集的點，譬如 meshgrid(-2:0.05:10,-5:0.05:5);。

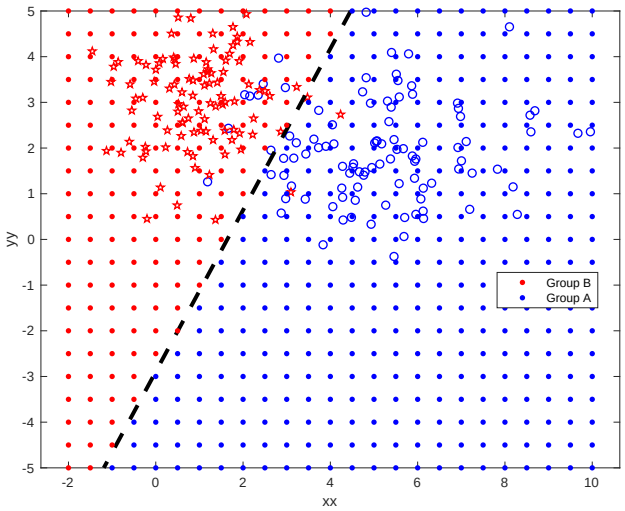

圖 3: LDA 群組預測與空間劃分。

MATLAB 另提供一個分群的指令 classify 將已知的群組資料與未知待分群的資料一並輸入。參考指令如下。執行結果如圖 2。

```
D=load('la_1.txt')
n=size(D,1);                                  % 總樣本數
g=cell(n,1);                                  % 製作群組標示
g(D(:,3)==0)={'Group A'};                     % 第一組資料稱爲 Group A
g(D(:,3)==1)={'Group B'};                     % 第二組資料稱爲 Group B
gscatter(D(:,1), D(:,2), g, 'br', 'op')       % 畫出群組資料的散佈圖
X=D(:,1:2);                                   % 準備資料矩陣：n x 2
[x,y] = meshgrid(-2:0.5:10,-5:0.5:5);         % 給定空間範圍內的資料點
x = x(:); y = y(:);
M = classify([x y], X, g);                    % 根據 X 與 g 對 [x y] 分群
hold on, gscatter(x, y, M, 'rb', '..'), hold off
```

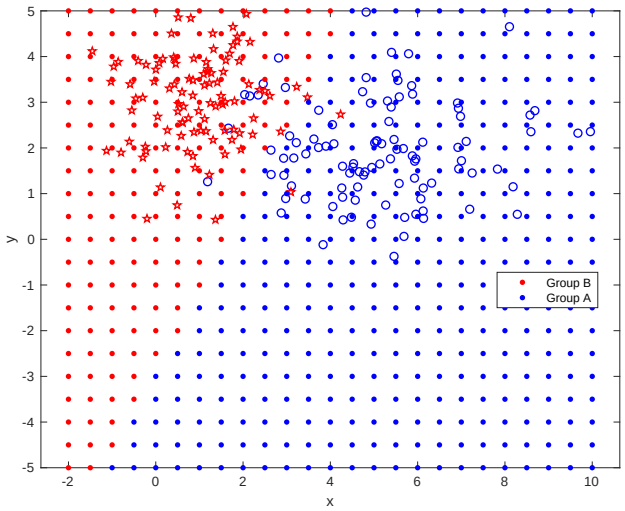

圖 4: 使用指令 classify 代替 fitcdiscr 與 predict。

分群指令 classify 也能輸出分界線的係數，使用方式詳見手册。

範例 3 相對於前述範例根據式 (4) 畫出一條直線分界線，式 (9) 則是一條曲線。MATLAB 的指令 fitcdiscr 已經預留二次判別式的功能。本範例援用相同的資料，畫一條二次方程式的分界線。

在此直接採用 MATLAB 指令 fitcdiscr，並加入判別式選項爲"quadratic"（內設選項爲"linear"）。因爲判別式爲二次式的關係，其輸出結果的係數中增加了二次項係數。假設輸出的常數項爲 K，一次項係數爲向量 L，二次項係數爲 2 x 2 矩陣 Q，則對比式 (9)

的方程式，寫爲

$$0 = K + L(1)X_1 + L(2)X_2 + Q(1,1)_1^2 + Q(1,2)X_1X_2 + Q(2,2)X_2^2$$

繪出這條二次方程式的分界線程式碼如下，結果如圖

```
D=load('la_1.txt')
n=size(D,1);  % 總樣本數
g=cell(n,1); % 製作群組標示
g(D(:,3)==0)={'Group A'}; % 第一組資料稱爲 Group A
g(D(:,3)==1)={'Group B'}; % 第二組資料稱爲 Group B
gscatter(D(:,1), D(:,2), g, 'br', 'op') % 畫出群組資料的散佈圖
X=D(:,1:2); % 準備資料矩陣: n x 2
Qda = fitcdiscr(X, g, 'DiscrimType', 'quadratic'); % 加入判別選項
K=Qda.Coeffs(1,2).Const; % 常數項
L=Qda.Coeffs(1,2).Linear;  % 2 x 1 的線性係數
Q=Qda.Coeffs(1,2).Quadratic; % 2 x 2 的二次項係數矩陣
f=@(x1,x2) K+L(1)*x1+L(2)*x2+ Q(1,1)*x1.^2+...
    (Q(1,2)+Q(2,1))*x1.*x2+Q(2,2)*x2.^2;
hold on, fimplicit(f,'LineWidth',3), hold off
```

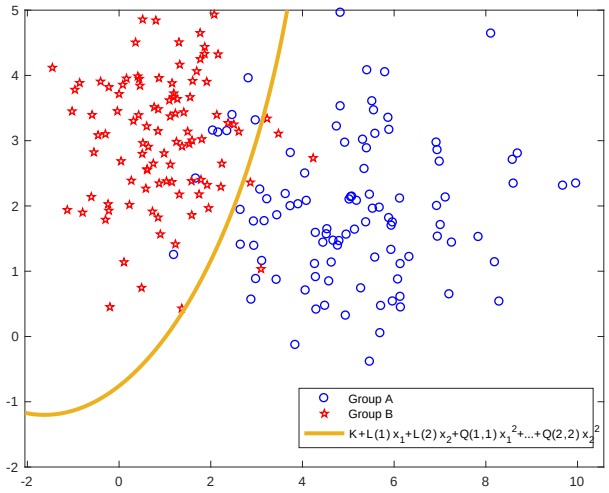

圖 5: 使用指令 fitcdiscr 產生二次判別式的分界線係數所繪製的曲線。

範例 4 爲區別群組間的直線與曲線分界線之差異，測試資料 la_2.txt 的群組散佈情況可以看得很清楚。請下載資料並試著做出如圖 6 的結果。

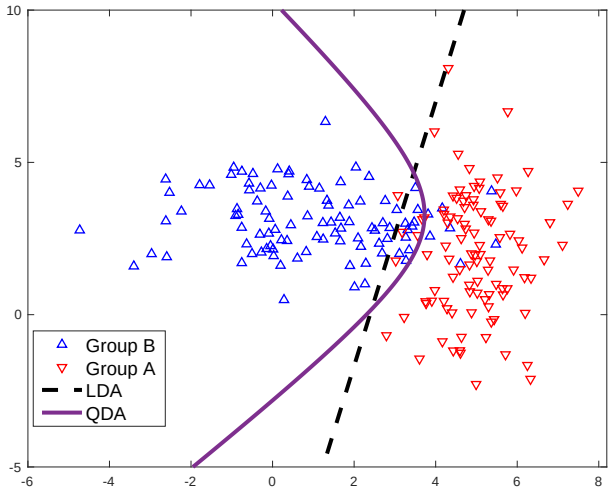

圖 6: 應用在資料 la_2.txt 的直線與曲線分界線。

從圖 6 隱約能看出曲線分界線的表現比較好。所謂「表現比較好」的意思是分群的錯誤率較低。MATLAB 亦提供指令 resubLoss 從分群的輸出結果計算分群的錯誤率。接續上述程式碼，加入計算分群錯誤率與表達：[5]

```
LdaErr=resubLoss(Lda);                          % 直線分界線的錯誤率
QdaErr=resubLoss(Qda);                          % 曲線分界線的錯誤率
fprintf('The classification errors are:\n')
fprintf('LDA: %f \n', LdaErr)
fprintf('QDA: %f \n', QdaErr)
```

錯誤率低的判別式不一定代表這個判別式比較好，只能說面對現有的資料（一般稱爲訓練資料）表現表較好。至於面對未知資料是否依然表現比較好，必須仰賴近一步的測試。一般的做法是從原始資料中撥出一部分做爲訓練資料（通常占大部分），讓「機器學習」，學習好之後的判別式再拿來面對剩下的那部分資料（稱爲測試資料），最後優劣的定奪仍以測試資料的錯誤率爲主。

範例 5 前面的範例中使用等量的群組資料，但當群組大小不同時，結果會有什麼差別呢？不妨自己產生模擬資料，模擬兩個群組的位置與規模，再利用這些人工資料看看群組間距與規模大小對 LDA 的表現的影響？同樣的是否也試試三個群組或以上的分群狀況。

---

[5]指令 resubLoss 名稱中的 resub 代表 resubstitution 的意思，即代回。也就是將原始資料代回判別式做分群判斷，並將結果與原始群組別做比較，比對結果不一樣的記錄一次錯誤。最後輸出錯誤比率。

假設兩個群組的中心點與共同的共變異矩陣爲:

$$\mu_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \mu_2 = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \Sigma = \begin{bmatrix} 1 & 0.4 \\ 0.4 & 1 \end{bmatrix}$$

兩群組大小分別爲 $n_1 = 500, n_2 = 250$。以下程式碼模擬了 750 筆資料，最後並存檔當作訓練資料（training data），執行結果如圖 7。

```
n1=500; n2=250; n=n1+n2;
mu1 = [1   -1]; mu2 = [-1   1];
Sigma = [1   0.4; 0.4   1];
g=cell(n,1);                          % 製作群組標示
g(1:n1)={'Group A'};                  % 第一組資料稱爲 Group A
g(n1+1:n)={'Group B'};                % 第二組資料稱爲 Group B
A = mvnrnd(mu1, Sigma, n1);           % 第一組資料 Group A
B = mvnrnd(mu2, Sigma, n2);           % 第二組資料 Group B
X=[A ; B]                             % 資料矩陣: n x 2
gscatter(X(:,1), X(:,2), g, 'br', 'op')  % 畫出群組資料的散佈圖
save trainingData X g
```

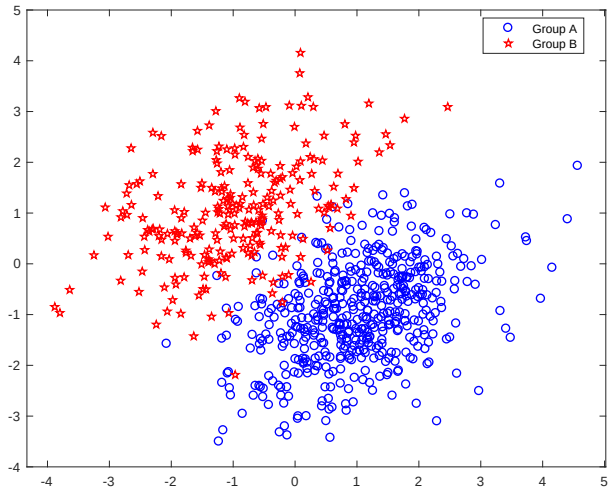

圖 7: 數量不同的兩群組模擬資料。

3 觀察與延伸

1. 在兩個群組的情況下，如果訓練資料中兩群組的數量不同，會產生什麼狀況？譬如: $Pr(G=1)=2Pr(G=2)$。
2. LDA 做出來的分界線與第一單元的迴歸模式（最小平方法）都是線性的，在只有兩個群組的情況下，有何相似之處嗎？試著針對同一組資料把這兩條線同時畫出來，比較看看。要仔細看喔!
3. Logistic regression 是處理類別輸出資料一項很基本但也很重要的工具。運用在群組分析上可以避免如 LDA 對概似函數 (likelihood function) 的常態性假設。這算是對常態分配的解套，不過另一方面，它也做了對 posterior probability $P(G|X)$ 的假設

$$p = Pr(G=1|X,\beta) = \frac{e^{\beta_0+\beta_1 x}}{1+e^{\beta_0+\beta_1 x}} \tag{10}$$

這個假設對許多的資料來源而言，具有相當程度的合理性。令其 log odds ratio 等於 0 時，求得最佳的分界線:

$$c = \ln\left(\frac{p}{1-p}\right) = \beta_0+\beta_1 x = 0 \tag{11}$$

參數 β_0, β_1 由已知的資料估計而得。前面的單元已經介紹過它的理論基礎與計算方法，原來的程式只要稍作些微的修改即可應用在本單元的資料。

4 習題

1. 將你的程式運用在 mix.mat 這組資料，畫出分界線，並允許輸入新資料，且立即輸出該資料經判別後的組別。
2. 請比較 logistic regression 與 LDA 在假設上、方法上及最後做出的分界線有何不同。Hint: 式 (4) 與 (11)。
3. 試著自己產生三組別的資料各 100 筆，之間的距離自己拿捏。利用 LDA 的方法在不同的兩個群組間畫一條線，共可畫出三條分界線以區隔三個群組。這三條線是否交於同一點？

References

[1] T. Hastie, R. Tibshirani, J. Friedman, ”The Elements of Statistical Learning:Data Mining, Inference, and Prediction,” Springer.