Practice#1：Classification with Multi-class

目的：

前面幾個單元將類別群組侷限在兩組，主要目的是簡化一些數學上的推導及程式的寫作。當類別群組增加時，是否會產生新的問題？是否原來表現好的方法會失效了呢？程式設計的困難度到底增加多少？程式的執行是否也會遭遇困難？沒有實際去做看看很難判斷出來。本單元將前面的幾個方法用在三個群組的類別資料，實際去體驗數學推導與程式撰寫的複雜度。當未來面對更多的群組時，比較能知道問題的難度。

本單元將討論三個群組的資料分析，資料來自兩個自變數X_1,X_2與一個類別型態的因變數 Y。討論的結果將可以輕易的推展到三個以上的群組及多個自變數。

Linear regression and least squares

在前面的單元裡，兩個變數的 linear regression model 定義為

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 \qquad (1)$$

當資料只有兩個類別時，Y 值可以被紀錄為 y=0 或 1，透過最小平方法的估算，找出最佳的參數值 $\hat{\beta}$，而兩群組間的分界線為 $\underline{x}^T \underline{\hat{\beta}} = 0.5$。但當資料分為三個群組時，Y 值的表達固然可以採數值資料如 y=1, 2 或 3（或-1,0,1），並且同樣的估算出參數值，同樣的找出分界線，但此時只有一組分界線，將無法有效的區別兩個以上的群組。

若將輸出改為三個因變數 Y_1,Y_2,Y_3，記錄為(y_1,y_2,y_3)=(1,0,0)、(0,1,0)或(0,0,1)分別代表當輸出資料為第一、第二與第三組。如此進行最小平方法的估算，將得出三組參數值，代表三組分界線：如下

$$\mathrm{B} = [\underline{\hat{\beta}}_1 \ \underline{\hat{\beta}}_2 \ \underline{\hat{\beta}}_3] = (X^T X)^{-1} X^T Y = (X^T X)^{-1} X^T [\underline{y}_1 \ \underline{y}_2 \ \underline{y}_3] \qquad (2)$$

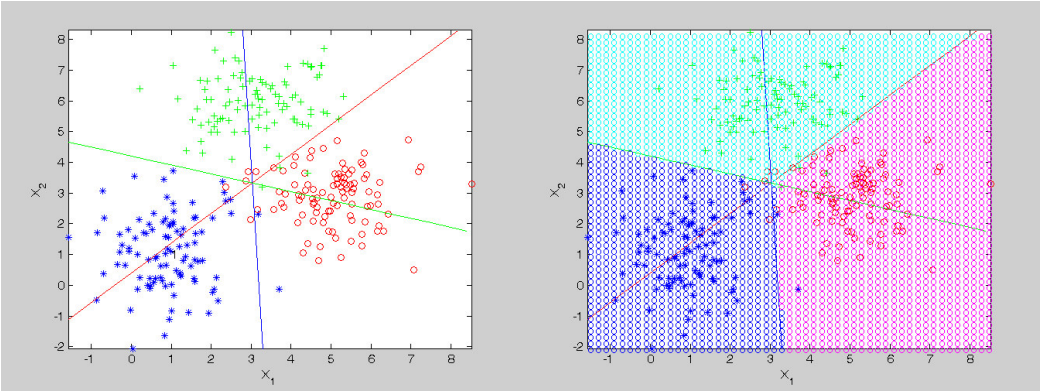
三條分界線方程式分別為：

$$\begin{aligned} \underline{\hat{\beta}}_1^T \underline{x} &= \underline{\hat{\beta}}_2^T \underline{x} \\ \underline{\hat{\beta}}_2^T \underline{x} &= \underline{\hat{\beta}}_3^T \underline{x} \\ \underline{\hat{\beta}}_1^T \underline{x} &= \underline{\hat{\beta}}_3^T \underline{x} \end{aligned} \qquad (3)$$

從三組參數估計值 $\hat{\underline{\beta}}_1$ $\hat{\underline{\beta}}_2$ $\hat{\underline{\beta}}_3$ ，對新資料的類別判斷為：

$$\arg \max_k \hat{\underline{\beta}}_k^T \underline{x} \tag{4}$$

透過分界線將(x1,x2)平面區分為三個部分如下圖所示：



Logistic regression

Logistic regression 適合應用在類別型的輸出資料。前面單元針對比較簡單的兩個類別，在此試著擴大到三個類別，最後的結果可以直接擴展的多個類別的判定。以三個類別而言，對於因變數 Y 的條件機率密度函數的 log odds ratio 定義如下：

$$\log \frac{\Pr(G = CLASS\#1 \mid X = \underline{x}, \underline{\beta})}{\Pr(G = CLASS\#3 \mid X = \underline{x}, \underline{\beta})} = \beta_{10} + \beta_{11} x_1 + \beta_{12} x_2 = \underline{\beta}_1^T \underline{x}$$

$$\log \frac{\Pr(G = CLASS\#2 \mid X = \underline{x}, \underline{\beta})}{\Pr(G = CLASS\#3 \mid X = \underline{x}, \underline{\beta})} = \beta_{20} + \beta_{21} x_1 + \beta_{22} x_2 = \underline{\beta}_2^T \underline{x}$$

where

$$\Pr(G = CLASS\#1 \mid X = \underline{x}, \underline{\beta}) = \frac{e^{\underline{\beta}_1^T \underline{x}}}{1 + e^{\underline{\beta}_1^T \underline{x}} + e^{\underline{\beta}_2^T \underline{x}}} = p_1(\underline{x}, \underline{\beta})$$

$$\Pr(G = CLASS\#2 \mid X = \underline{x}, \underline{\beta}) = \frac{e^{\underline{\beta}_2^T \underline{x}}}{1 + e^{\underline{\beta}_1^T \underline{x}} + e^{\underline{\beta}_2^T \underline{x}}} = p_2(\underline{x}, \underline{\beta})$$

$$\Pr(G = CLASS\#3 \mid X = \underline{x}, \underline{\beta}) = \frac{1}{1 + e^{\underline{\beta}_1^T \underline{x}} + e^{\underline{\beta}_2^T \underline{x}}} = 1 - p_1(\underline{x}, \underline{\beta}) - p_2(\underline{x}, \underline{\beta})$$

$$\underline{\beta} = \begin{bmatrix} \underline{\beta}_1 \\ \underline{\beta}_2 \end{bmatrix} \tag{5}$$

概似函數寫成

$$L(\underline{\beta}) = f(\underline{y} \mid \underline{x}, \underline{\beta}) = \prod_{k=1}^{N} f_k(y_k \mid x_k, \underline{\beta}) = \prod_{k=1}^{N} p_1(x, \underline{\beta})^{y_{1k}} p_2(x, \underline{\beta})^{y_{2k}} (1 - p_1(x, \underline{\beta}) - p_2(x, \underline{\beta}))^{1-y_{1k}-y_{2k}} \tag{6}$$

取對數後經化簡（作業 1），變成

$$\ln L(\underline{\beta}) = \sum_{k=1}^{N} (y_{1k} \underline{\beta}_1^{\,T} \underline{x}_k + y_{2k} \underline{\beta}_2^{\,T} \underline{x}_k - \ln(1 + e^{\underline{\beta}_1^{\,T} \underline{x}_k} + e^{\underline{\beta}_2^{\,T} \underline{x}_k})) \tag{7}$$

採最大概似法(maximum likelihood method)求最佳的 $\underline{\hat{\beta}}$ 值時，不論迭代的方向選擇為 steepest descent 或 Newton-Raphson 均需要先計算梯度向量(作業 1)：

$$\nabla \ln L(\underline{\beta}) = \begin{bmatrix} \nabla \ln L(\underline{\beta}_1) \\ \nabla \ln L(\underline{\beta}_2) \end{bmatrix} = \begin{bmatrix} \sum_{k=1}^{N} (y_k - \frac{e^{\underline{\beta}_1^{\,T} \underline{x}_k}}{1 + e^{\underline{\beta}_1^{\,T} \underline{x}_k} + e^{\underline{\beta}_2^{\,T} \underline{x}_k}}) \underline{x}_k \\ \sum_{k=1}^{N} (y_k - \frac{e^{\underline{\beta}_2^{\,T} \underline{x}_k}}{1 + e^{\underline{\beta}_1^{\,T} \underline{x}_k} + e^{\underline{\beta}_2^{\,T} \underline{x}_k}}) \underline{x}_k \end{bmatrix} \tag{8}$$

而 Newton-Raphson method 所需要的 Hessian matrix 如（作業 1）

$$\nabla^2 \ln L(\underline{\beta}) = \begin{bmatrix} \nabla^2 \ln L(\underline{\beta}_1) & \nabla_{\underline{\beta}_1} (\nabla \ln L(\underline{\beta}_2)^T) \\ \nabla_{\underline{\beta}_2} (\nabla \ln L(\underline{\beta}_1)^T & \nabla^2 \ln L(\underline{\beta}_2) \end{bmatrix}$$

$$= \begin{bmatrix} -\sum p_{1,k}(1 - p_{1,k}) \underline{x}_k \underline{x}_k^{\,T} & \sum p_{1,k} p_{2,k} \underline{x}_k \underline{x}_k^{\,T} \\ \sum p_{1,k} p_{2,k} \underline{x}_k \underline{x}_k^{\,T} & -\sum p_{2,k}(1 - p_{2,k}) \underline{x}_k \underline{x}_k^{\,T} \end{bmatrix} \tag{9}$$

$$where \quad p_{j,k} = \frac{e^{\underline{\beta}_j^{\,T} \underline{x}_k}}{1 + e^{\underline{\beta}_j^{\,T} \underline{x}_k} + e^{\underline{\beta}_j^{\,T} \underline{x}_k}}, \qquad j = 1,2$$

當 $\underline{\hat{\beta}} = \begin{bmatrix} \underline{\hat{\beta}}_1 \\ \underline{\hat{\beta}}_2 \end{bmatrix}$ 估計出來後，三條分界線方程式如：

$$\underline{\beta}_1^T \underline{x} = 0$$
$$\underline{\beta}_2^T \underline{x} = 0 \qquad (10)$$
$$\underline{\beta}_1^T \underline{x} = \underline{\beta}_2^T \underline{x}$$

分別為第一、三組，第二、三組及第一、二組的界線。

練習：

1. 利用之前寫過的最小平方法的複迴歸程式，擴展為三個因變數輸出。畫出三條回歸線。
2. 畫出回歸線後，平面空間已被切割三塊區域，分屬三個群組，請利用分類法則(4)，符號及顏色將整個平面塗上不同的顏色，如上面的圖。
3. 畫完圖之後的程式可以加入群組判斷的程式碼，讓使用者輸入新資料，並且隨即在畫面上輸出判斷的結果。這個動作可以持續進行，直到使用者不想繼續。參考以下的程式碼：

```
while 1                                          % 1 代表 TRUE，始終會進入 while 迴圈
    x_new=input('Enter a new data as in the form [x1 x2] or hit Enter to quit : ');
    if  isempty(x_new); break; end   %未輸入任何資料，即跳出迴圈離開
    ……開始判斷的程式
end
```

觀察：

1. 採邏輯斯迴歸的方式比較複雜，計算上比較麻煩。其中 Steepest Descent 與 Newton Method 是兩個經常被拿來比較的「移動方向。」不妨觀察這兩個方法的收斂情形。必要時當然也可以把函數圖畫出來，看看最低點附近的坡度比較適合哪一種方法。
2. 做完作業後，不妨觀察利用最小平方法的複迴歸及利用最大概似函數的邏輯斯迴歸，對於分類結果的差異。

作業：

1. 推導式子(7)、(8)及(9)，並進一步推導出如以下 MATLAB 程式所需要的矩陣型態：

$$\ln L(\underline{\beta}) = \underline{\beta}_1{}^T X \underline{y}_1 + \underline{\beta}_2{}^T X \underline{y}_2 - sum(\log(1+\exp(\underline{\beta}_1{}^T X) + \exp(\underline{\beta}_2{}^T X)))$$

$$\nabla \ln L(\underline{\beta}) = \begin{bmatrix} X(\underline{y}_1 - \underline{z}_1) \\ X(\underline{y}_2 - \underline{z}_2) \end{bmatrix},$$

$$where \;\; X = [\underline{x}_1 \quad \underline{x}_2 \quad \cdots \quad \underline{x}_N], \qquad \underline{y}_k = \begin{bmatrix} y_{k1} \\ y_{k2} \\ \vdots \\ y_{kN} \end{bmatrix}, \quad k = 1,2$$

$$\underline{z}_k = \begin{bmatrix} \dfrac{e^{\underline{\beta}_k{}^T \underline{x}_1}}{1 + e^{\underline{\beta}_1{}^T \underline{x}_1} + e^{\underline{\beta}_2{}^T \underline{x}_1}} \\ \dfrac{e^{\underline{\beta}_k{}^T \underline{x}_2}}{1 + e^{\underline{\beta}_1{}^T \underline{x}_2} + e^{\underline{\beta}_2{}^T \underline{x}_2}} \\ \vdots \\ \dfrac{e^{\underline{\beta}_k{}^T \underline{x}_N}}{1 + e^{\underline{\beta}_1{}^T \underline{x}_N} + e^{\underline{\beta}_2{}^T \underline{x}_N}} \end{bmatrix}, k = 1,2$$

$$\nabla^2 \ln L(\underline{\beta}) = \begin{bmatrix} -XW_1X^T & XW_3X^T \\ XW_3X^T & -XW_2X^T \end{bmatrix}$$

其中 W1, W2, 及 W3 為對角矩陣(diagonal matrix)，其第(k,k)個位置分別為：$p_{1,k}(1-p_{1,k})$, $p_{2,k}(1-p_{2,k})$, $p_{1,k}p_{2,k}$， $p_{1,k}$，$p_{2,k}$ 的定義如(9)。

2. 利用網路上提供的資料或是自行產生適當的資料，含三個群組。畫出其邏輯斯迴歸的分界線，如本單元所繪製的圖。
3. 當給一組新的資料，邏輯斯迴歸法如何將其歸類呢？請寫出數學的表示式，並在程式的最後加上這一段。

參考文獻：

1. T. Hastie, R. Tibshirani, J. Friedman, “The Elements of Statistical Learning” .
2. A. Webb, ”Statistical Pattern Recognition,” second edition, John Wiley & Sons。