Maximum a posteriori (MAP) Classifier

Author: Ming-Ru Wu

過程:

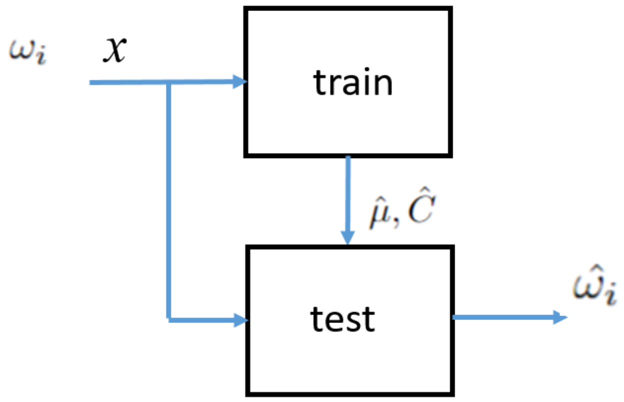

圖 1. MAP 分類器(classifier)

圖 1 的輸入資料為 d 維常態分佈向量 x, 其產生器 Matlab 程式 **xGen.m** 如下:

```
function xd = xGen(dim, N, mu, rho)
r = rho.^(linspace(0,dim-1,dim));
C = toeplitz(r); % 共變異數矩陣C: stationary random process

[U,D] = eig(C); % 將共變異數矩陣進行eigendecomposition
A = U * diag(sqrt(diag(D))); % A = U*D^0.5

xd = zeros(dim, N);
for k=1:N
    y = randn(dim,1); % y ~ N(0, I), 產生一標準常態分布資料
    x0 = A * y;       % 經A矩陣轉換使其共變異數矩陣為C
    x = x0 + mu;      % 將y ~ N(0, I) 轉換至 x ~ N(mu, C)
    xd(:, k) = x;
end
```

圖 1 中的 MAP 分類器(classifier)可分為以下 2 模組:
訓練(train)模組: 使用 x 的前半段資料, 來產生平均值向量(μ)與共變異數矩陣(C)的估計值; **測試(test)模組:** 使用 x 的後半段資料, 來決定類別(ω_i)並計算錯誤率.

MAP 分類器(classifier)的原理如下所述:

$$p(x|\omega_i) = \frac{1}{(2\pi)^{\frac{d}{2}}|C_i|^{\frac{1}{2}}} \exp\left[-\frac{1}{2}(x-\mu_i)'C_i^{-1}(x-\mu_i)\right]$$

MAP: $$p(\omega_1)p(x|\omega_1) \underset{\omega_2}{\overset{\omega_1}{\gtrless}} p(\omega_2)p(x|\omega_2)$$

$$2\ln[p(\omega_1)] - \ln|C_1| - (x-\mu_1)'C_1^{-1}(x-\mu_1) \underset{\omega_2}{\overset{\omega_1}{\gtrless}} 2\ln[p(\omega_2)] - \ln|C_2| - (x-\mu_2)'C_2^{-1}(x-\mu_2)$$

可以使用如下的矩陣分解法來簡化 MAP 分類器的計算:

$$C = UDU' = UD^{\frac{1}{2}}D^{\frac{1}{2}}U' = AA'$$

$$A = UD^{\frac{1}{2}}$$

$$C^{-1} = (AA')^{-1} = (A')^{-1}A^{-1} = UD^{-\frac{1}{2}}D^{-\frac{1}{2}}U' = BB'$$

$$\boxed{B' = D^{-\frac{1}{2}}U'}$$

因此 MAP 分類器的主要計算公式可簡化為

$$(x-\mu)'C^{-1}(x-\mu) = (x-\mu)'BB'(x-\mu) = z'z$$

$$\boxed{z = B'(x-\mu)}$$

MAP (Maximum a Posteriori)的 discriminant function 為

$2\ln[p(\omega_1)] - \ln|C_1| - (x-\mu_1)'C_1^{-1}(x-\mu_1)$, $2\ln[p(\omega_2)] - \ln|C_2| - (x-\mu_2)'C_2^{-1}(x-\mu_2)$

依此設計程式 logMAP.m 如下:

```
function discriminant = logMAP(x, c, mv, Bh)

z = Bh * (x - mv);   % Bh=B'
discriminant = c - z' * z;
```

其中 $z=B'(x-\mu)$, $c=2\ln[p(\omega)] - \ln|C|$.

產生 B'(亦即程式中 Bh)與常數項 $\ln|C_1|$ 的 Matlab 程式碼(disc.m)如下:

```
function [log_detC, Bh] = disc(C)

[U,D] = eig(C);
d = diag(D);
sqrt_d = sqrt(d);
log_detC = log(prod(d));
Bh = diag(1 ./ sqrt_d) * U';
```

圖 1 中 train 模組對應的 Matlab 程式 **train.m**, 計算 xmean (μ)與 Cx (C)的估計值如下:

```
function [xmean, Cx] = train(x)
d = size(x,1);
N = size(x,2);
xmean = zeros(d,1);
for k=1:N
    xmean = xmean + x(:, k);
end
xmean = xmean / N;

Cx = zeros(d);
for k=1:N
    x0 = x(:, k) - xmean;
    Cx = Cx + x0 * x0';
end
Cx = Cx / N;
```

主程式 **ex2.m** 呼叫上述的 xGen.m 與 train.m 模組如下:

```
N = 10000;          % number of data
Nhalf = N/2;        % half data for training; half for testing
d = 50;             % dimension
rho1 = 0.9;         % parameter for covariance matrix in distribution 1
mu1 = 0;            % mean vector for distribution 1
rho2 = 0.7;         % parameter for covariance matrix in distribution 1
mu2 = 0.5;          % mean vector for distribution 2
p1 = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]; % prior prob

% generate samples
xd1 = xGen(d, N, mu1, rho1);
xd2 = xGen(d, N, mu2, rho2);

% training phase
[xmean1, Cx1] = train(xd1(:, 1:Nhalf)) % mean, covariance of class-1
[xmean2, Cx2] = train(xd2(:, 1:Nhalf)) % mean, covariance of class-2

% discriminator
[log_detC1, Bh1] = disc(Cx1);
[log_detC2, Bh2] = disc(Cx2);
```

接著主程式 **ex2.m** 進入圖 1 的測試(test)模組, 根據不同的事前機率 P(ω_i), 使用 MAP discriminant function (logMAP)來判定類別(class-1 或 class-2), 並計算錯誤率 P(error), 並畫出圖 2.

```
% test phase
Np = length(p1);
er1 = zeros(Np,1);
er2 = zeros(Np,1);
for i = 1: Np
    q1 = p1(i); % prior probability for class 1
    q2 = 1-q1;  % prior probability for class 2
    c1 = 2*log(q1) - log_detC1;
    c2 = 2*log(q2) - log_detC2;

    %%%%%%%%%%%%%% test
    class = 1;
    err1 = 0.;
    for k=Nhalf+1:N   % use MAP classifier for class-1
        [val, decision] = max([logMAP(xd1(:, k), c1, xmean1, Bh1) , ...
            logMAP(xd1(:, k), c2, xmean2, Bh2)]);
        if(class ~= decision)
            err1 = err1 + 1;
        end
    end
    fprintf("class-1 error rate = %f\n", err1/Nhalf); % class-1 error
    er1(i) = err1/Nhalf;

    class = 2;
    err2 = 0.;
    for k=Nhalf+1:N   % use MAP classifer for class-2
        [val, decision] = max([logMAP(xd2(:, k), c1, xmean1, Bh1) , ...
            logMAP(xd2(:, k), c2, xmean2, Bh2)]);
        if(class ~= decision)
            err2 = err2 + 1;
        end
    end
    fprintf("class-2 error rate = %f\n", err2/Nhalf); % class-2 error
    er2(i) = err2/Nhalf;
end

plot(p1, er1, 'b-', p1, er2, 'r-.'), legend('class-1', 'class-2'), ...
    xlabel('p(w_1)'), ylabel('P(error)'), title('MAP Classifier')
```

結果:

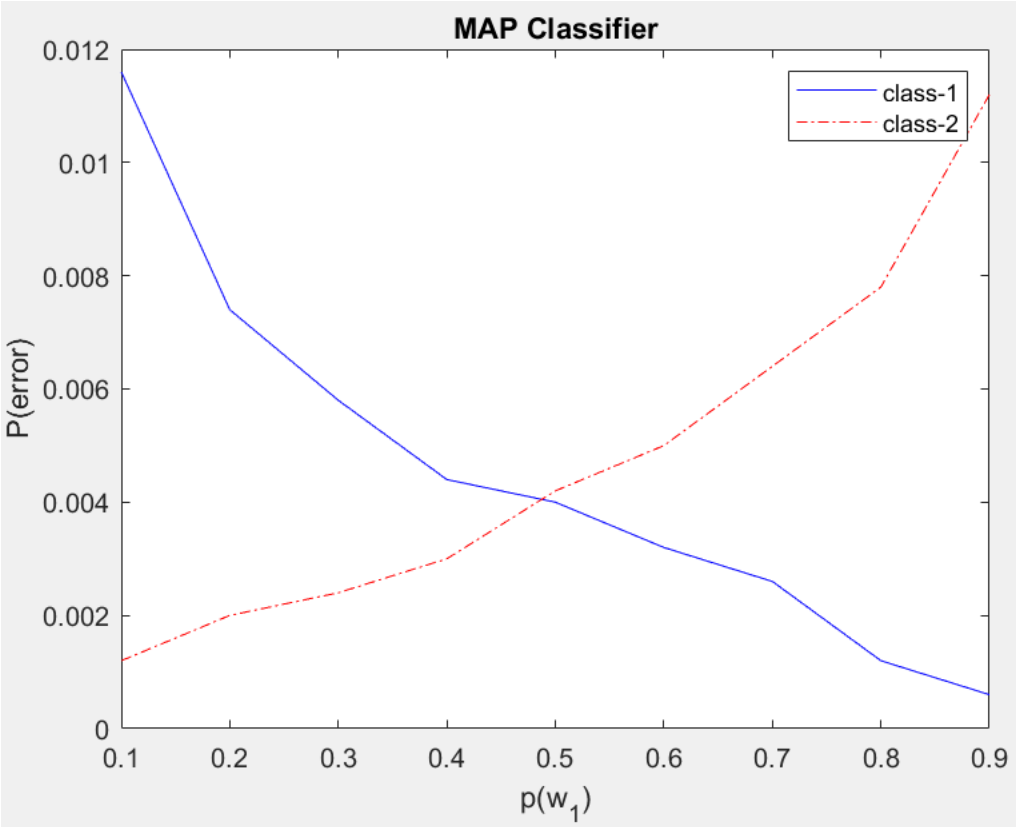

圖 2. 根據不同數值 0.1 至 0.9 的事前機率 P(ω_1), 產生並繪製錯誤機率 P(error)曲線圖. 藍線代表給予屬於類別 1 的資料 x 被誤判成類別 2 的錯誤率; 紅線代表給予屬於類別 2 的資料 x 被誤判成類別 1 的錯誤率