

Übungen zur Vorlesung  
Grundlagen der Bilderzeugung und Bildanalyse (Mustererkennung)  
WS 05/06

Musterlösung 11

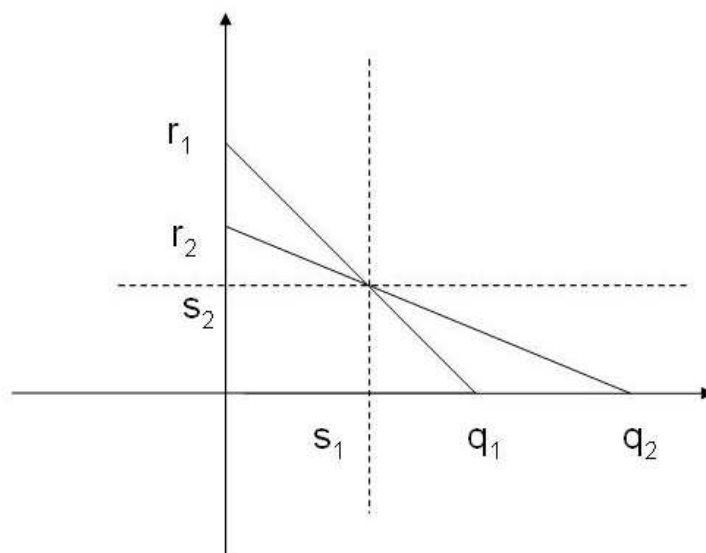
Aufgabe 11.1: Bayes-Klassifikator mit Zurückweisung

1. Da die Summe der Verteilungsdichten 1 ergeben muss und  $p(x|\omega_i)$  durch eine Gerade mit Steigung  $r_i/q_i$  und Schnittpunkt mit der  $y$ -Achse  $r_i$  dargestellt werden kann, lautet der Zusammenhang zwischen  $q_i$  und  $r_i$ :

$$\frac{1}{2}r_iq_i = 1$$

Dies ist genau die Fläche unter der Geraden.

2. Um die Bereiche von  $x$  zu finden, in denen man die Klasse  $\omega_1$  bzw.  $\omega_2$  nach dem MLE wählt, muss man den Schnittpunkt  $\mathbf{s} = (s_1, s_2)$  der zwei Verteilungsdichten finden:



Wir berechnen nun den Schnittpunkt  $\mathbf{s}$ :

$$\begin{aligned} p(x|\omega_1) &= p(x|\omega_2) \\ \Leftrightarrow r_1 - \frac{r_1}{q_1}x &= r_2 - \frac{r_2}{q_2}x \\ \Leftrightarrow r_1 - \frac{1}{2}r_1^2x &= r_2 - \frac{1}{2}r_2^2x \end{aligned}$$

$$\begin{aligned}\Leftrightarrow r_1 - r_2 &= \left(\frac{1}{2}r_1^2 - \frac{1}{2}r_2^2\right)x \\ \Leftrightarrow x &= 2\frac{r_1 - r_2}{r_1^2 - r_2^2} \\ \Leftrightarrow x &= \frac{2}{r_1 + r_2}\end{aligned}$$

Und wir erhalten:

$$s_1 = \frac{2}{r_1 + r_2}$$

Wir nehmen O.B.d.A. an, dass  $r_2 < r_1$  ist und somit  $q_1 < q_2$ . Dann wird nach dem MLE Klassifikator im Bereich  $[0, s_1)$  für die Klasse  $\omega_1$  entschieden und im Bereich  $(s_1, q_2]$  für die Klasse  $\omega_2$ .

3. Der Schnittpunkt muss für den MAP Klassifikator neu berechnet werden:

$$\begin{aligned}\frac{1}{3}p(x|\omega_1) &= \frac{2}{3}p(x|\omega_2) \\ \Leftrightarrow r_1 - \frac{r_1}{q_1}x &= 2 \cdot \left(r_2 - \frac{r_2}{q_2}x\right) \\ \Leftrightarrow r_1 - \frac{1}{2}r_1^2x &= 2r_2 - r_2^2x \\ \Leftrightarrow r_1 - 2r_2 &= \left(\frac{1}{2}r_1^2 - r_2^2\right)x \\ \Leftrightarrow x &= \frac{r_1 - 2r_2}{\frac{1}{2}r_1^2 - r_2^2}\end{aligned}$$

Wir nehmen O.B.d.A. an, dass  $r_2 < r_1$  ist und somit  $q_1 < q_2$ . Dann wird nach dem MAP Klassifikator im Bereich  $[0, \frac{r_1 - 2r_2}{\frac{1}{2}r_1^2 - r_2^2})$  für die Klasse  $\omega_2$  entschieden und im Bereich  $(\frac{r_1 - 2r_2}{\frac{1}{2}r_1^2 - r_2^2}, q_2]$  für die Klasse  $\omega_1$ .

4. Jetzt muss das System für eine Stichprobe nicht nur einfach klassifizieren sondern auch entscheiden, ob sich die Klassifikation überhaupt lohnt.

Wir definieren  $C_{KL}(\mathbf{n})$  als die zu erwartenden Kosten, die entstehen, falls man die Probe  $\mathbf{n}$  klassifiziert und  $C_{ZW}(\mathbf{n})$  als Kosten falls man zurückweist. Somit gilt:

$$\begin{aligned}C_{ZW}(\mathbf{n}) &= k \\ C_{KL}(\mathbf{n}) &= 3k \cdot P_{Fehler}(\mathbf{n})\end{aligned}$$

Dabei steht  $P_{Fehler}(\mathbf{n})$  für die Wahrscheinlichkeit, dass  $\mathbf{n}$  falsch klassifiziert wird.

Um die totalen Kosten zu minimieren, muss die Entscheidungsfunktion das Minimum der beiden Kosten wählen:

$$C_{TOTAL}(\mathbf{n}) = \min(C_{ZW}(\mathbf{n}), C_{KL}(\mathbf{n}))$$

und entscheiden, ob  $\mathbf{n}$  klassifiziert oder zurückgewiesen wird:

$$E_{TOTAL}(\mathbf{n}) = \arg \min(C_{ZW}(\mathbf{n}), C_{KL}(\mathbf{n}))$$

Nun muss  $C_{KL}(\mathbf{n})$  genau berechnet werden:

$$C_{KL}(\mathbf{n}) = 3k \cdot P_{Fehler}(\mathbf{n}) = 3k \cdot \frac{\min(P_1(\mathbf{n}), P_2(\mathbf{n}))}{P_1(\mathbf{n}) + P_2(\mathbf{n})}$$

( $P_i(\mathbf{n})$  steht für  $p(\mathbf{n}|\omega_i)$ .) Wir berechnen den Schnittpunkt  $\mathbf{n}_s$ :

$$\mathbf{n}_s = \frac{2}{r_1 + r_2} = \frac{2}{1 + 2/3} = \frac{6}{5}$$

Dann ist:

$$C_{KL}(\mathbf{n}_s) = 3k \cdot \frac{1}{2} = \frac{3k}{2} > k$$

Um diesen Punkt finden wir den Bereich, für den  $C_{KL}(\mathbf{n}) > C_{ZW}(\mathbf{n})$  ist. Wir unterscheiden zwei Fälle:

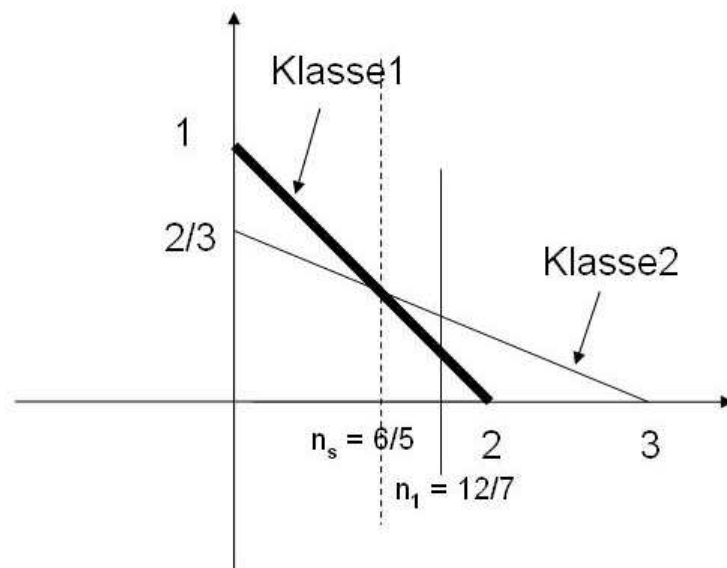
a.)  $\min(P_1(\mathbf{n}), P_2(\mathbf{n})) = P_1(\mathbf{n})$ :

$$\begin{aligned} 3k \cdot \frac{P_1(\mathbf{n}_1)}{P_1(\mathbf{n}) + P_2(\mathbf{n})} &= k \\ \Leftrightarrow \frac{1 - \frac{1}{2}\mathbf{n}_1}{(1 - \frac{1}{2}\mathbf{n}_1) + (\frac{2}{3} - \frac{2}{9}\mathbf{n}_1)} &= \frac{1}{3} \\ \Leftrightarrow \mathbf{n}_1 &= \frac{12}{7} \end{aligned}$$

b.)  $\min(P_1(\mathbf{n}), P_2(\mathbf{n})) = P_2(\mathbf{n})$ :

$$\begin{aligned} 3k \cdot \frac{P_2(\mathbf{n}_2)}{P_1(\mathbf{n}) + P_2(\mathbf{n})} &= k \\ \Leftrightarrow \frac{\frac{2}{3} - \frac{2}{9}\mathbf{n}_2}{(1 - \frac{1}{2}\mathbf{n}_2) + (\frac{2}{3} - \frac{2}{9}\mathbf{n}_2)} &= \frac{1}{3} \\ \Leftrightarrow \mathbf{n}_2 &= -6 \end{aligned}$$

Da  $\mathbf{n}_2$  negativ ist, bedeutet dies, dass es sich gar nicht lohnt, sich für Klasse 1 zu entscheiden. Die neuen Bereiche für die Klassifikation nach MLE sind dann: im Bereich  $[0, \frac{12}{7}]$  Zurückweisung und im Bereich  $(\frac{12}{7}, 3]$  Entscheidung für die Klasse  $\omega_2$ .



## Aufgabe 11.2: Programmieraufgabe: Mahalanobis-Klassifikator

1. Eine Implementation, die gewünschtes liefert ist folgende

```
function [classes, cert] = mahalanobis_class(X,K,M);
//-----
// function [classes, cert] = mahalanobis_class(X,K,M);
//
// function implementing mahalanobis classification of arbitrary
// dimensional data and arbitrary number of classes.
//
// input:   X           Matrix of columnwise vectors to be classified
//          K           Covariance-matrix used for Mahalanobis-distance
//          M           Matrix of columnwise mean-vectors of the classes
//
// output:  classes    vector of estimated class labels for
//                  every vector in X
//          cert       vector of estimated "certainties" of each
//                  classification
//-----

k = size(M,2); // number of classes
N = size(X,2); // number of test-examples

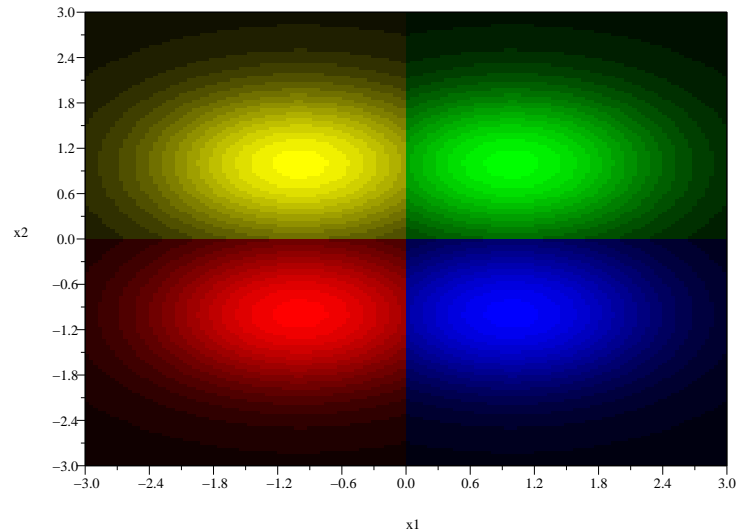
// (squared) mahalanobis-distance-Matrix
//  $D_{ij} = (x_j - \mu_i)^T * K^{-1} * (x_j - \mu_i)$ 
D = zeros(0,N);
for i = 1:k;
    MM = M(:,i).*ones(1,N);
    T = inv(K)*(X-MM);
    d = sum(T.*(X-MM), 'r');
    D = [D; d];
end;

// determine minimum distance and classes
[di, cl] = min(D, 'r');
// determine results
classes=cl';
cert = exp(-di)';

endfunction;
```

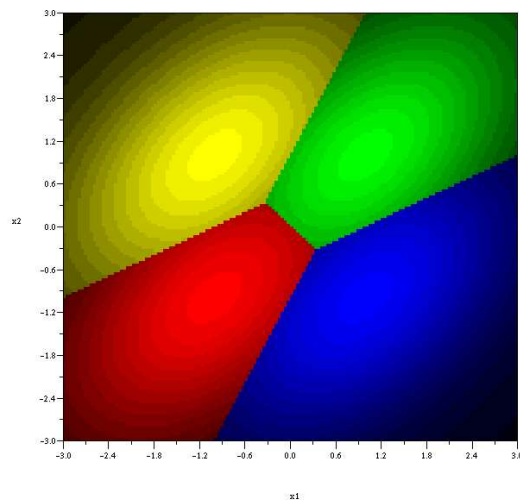
Es wurde versucht, moeglichst wenig und kurze Schleifen zu verwenden, in diesem Fall reduziert sich dies auf eine Schleife über die Anzahl der Klassen. Für jede Klasse wird ein Vektor mit quadratischen Mahalanobis-Abständen aller Testvektoren und dem Mittel der aktuellen Klasse berechnet. Dies ist durch Vervielfältigen des Mittelwertvektors mit der `.*` Operation und anschließenden Matrixoperationen ohne weitere Schleife möglich. Diese Vektoren werden zu einer Distanz-Matrix **D** zusammengefügt, in der nun lediglich für jedes Testbeispiel der minimale Klassenabstand und zugehörige Position (= Klassennummer) ermittelt werden. Zuletzt wird noch das angegebene heuristische Sicherheitsmaß für jede Klassenentscheidung berechnet.

2. Zur Visualisierung wurde der 2d-Modus von `visclass` mit Auflösung  $150 \times 150$  gewählt, und die vierte Klassenfarbe von Schwarz auf Gelb geändert. Die Klassifikationsgebiete mit dem ersten Parametersatz sind nun



Zunächst einmal bestätigt sich, daß die Grenzlinien stückweise lineare Kurven sind im Gegensatz zum allgemeineren Bayes-Klassifikator für normalverteilte Klassen, der stückweise quadratische Kurven erzeugt. Man kann es zeigen, dass falls die Verbindungslinien der Klassenmittelpunkte parallel zu einem Eigenvektor der Matrix  $\mathbf{K}$  ist, spielen die Größe der Eigenwerten keine Rolle mehr für das Mahalanobis-Distanz-Klassifikator. Die Grenzlinien fallen also mit denen des Euklidischen Abstands-Klassifikators zusammen.

Das Ergebnis des zweiten Parametersatzes ist dieses:



In diesem Fall ist die Linie zwischen  $\mu_1$  und  $\mu_3$  wieder parallel zu einem Eigenvektor, so dass diese Grenzlinie orthogonal zu der Klassenmittel-Verbindungsline ist.