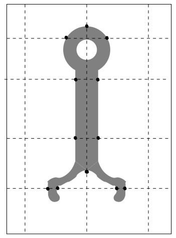
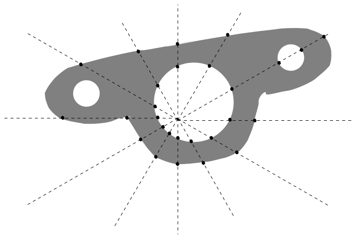
2 Intuitive Klassifikation

“intuitiv” bedeutet nicht schlecht oder primitiv,
sondern bezeichnet einen leicht zu verstehenden Zugang
zum Problem der Merkmalsgewinnung und Klassifikation

2.1 Einfache heuristische Merkmale

2.1.1 Schnittpunkte mit Testlinien



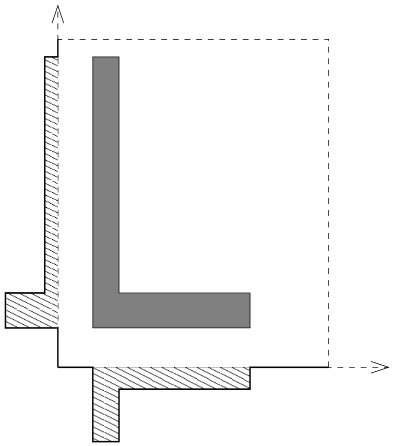


damit z.B.:

- Anzahl Schnittpunkte
- Länge der Testlinie(n) innerhalb des Musters
- Koordinaten der Schnittpunkte

z.B. Testlinien sternförmig um den Schwerpunkt

2.1.2 Projektion auf Geraden



daraus z.B.:

- Minima und Maxima
- Projektionskurve
- Projektion:

$$p_j^{(x)} = \sum_{k=0}^{M_x-1} f_{jk}$$

2.1.3 Globale Parameter, Kennzahlen

TODO: Abbildung

z.B. Formfaktor

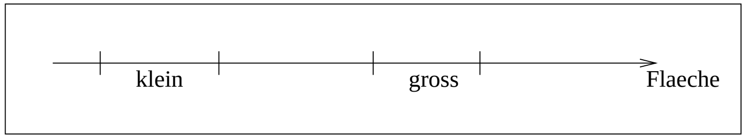
$$\begin{aligned} C &= \frac{L^2}{A} (Kompaktheit) \\ L &= \text{Länge der Umrißlinie} \\ A &= \text{Fläche} \end{aligned}$$

2.2 Klassifikation mit Hyperquadern

Für jede Komponente des Merkmalvektors wird je Klasse ein zulässiges Intervall vorgegeben.
Klassifikation (oder Rückweisung) kann dann einfach erfolgen.

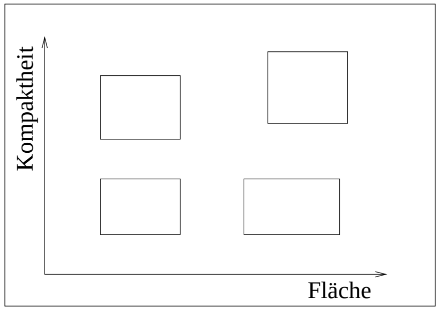
Beispiel: Unterscheidung von großen und kleinen Kreisen

Als Merkmal: Fläche



Beispiel: Unterscheidung von großen und kleinen Kreisen und Rechtecken

Merkmale: Fläche und Kompaktheit



Hyperquader-Klassifikatoren lassen sich sehr einfach realisieren, aber sind in ihren Möglichkeiten zur Trennung des Merkmalsraums sehr eingeschränkt.

2.3 Abstandsmessung zur Klassifikation

quantisiere die Ähnlichkeit von Mustern mit einer Metrik. z.B.:

$$d^{(r)}(\vec{c}_i, \vec{c}_j) = \sqrt[r]{\sum_{\nu} |c_{i_\nu} - c_{j_\nu}|^r}$$

$c_i, c_j \in \omega$
$c_{i_\nu}, c_{j_\nu} \in \omega$ sind die Komponenten der Merkmalsvektoren

$r = 1$ City-Block Metrik
$r = 2$ euklidischer Abstand
$r = \infty$ Maximum Norm

2.3.1 Nächster-Nachbar-Klassifikator

Bei gegebener klassifizierter Stichprobe klassifizieren wir eine (neues) Muster in die Klasse des nähesten Musters innerhalb dieser Stichprobe:
$\vec{c}$ ordnen wir Ω_κ zu, falls

$$i = \operatorname*{argmin}_{j=1,\ldots,N} \{d(\vec{c}, \vec{c}_j)\} \text{ und } \vec{c}_i \in \Omega_\kappa$$
$$\text{und } \omega = \{\vec{c}_j | j = 1, \ldots, N\}$$

Bemerkung:

- gibt gute Klassifikationsergebnis (was immer das heißen mag an dieser Stelle), wenn die Stichprobe (sehr) groß ist.
- keine Rückweisung
- einfach zu Lernen, großer Speicher- und Rechenaufwand für die Klassifikation
  Es gibt Methoden zum Ausdünnen der Stichprobe

mNN-Klassifikator

Modifikation, um Ausreißer oder Störungen an der Klassengrenzen besser zu Berücksichtigen:

- Entscheidung gemäß der Majorität der m nächsten Nachbarn
- Rückweisung, falls das Maximum von Vektoren aus einer Klasse kleiner als Schwelle $l < m$

Auch hier wird i.d.R. in Gebieten des Merkmalsraumes klassifiziert, die nicht in der Stichprobe/Problemkreis enthalten sind.

2.3.2 Klassifikation durch Vektorquantisierung

- Idee ist hier, Merkmalsvektoren im Zentrum der Klassengebiete zu finden.
- Einfachste Variante: bestimme den Schwerpunkt jeder Klasse.
  (Das gibt nur sehr einfache, insbesondere zusammenhängende Klassengebiete.)
- Vektorquantisierung dient (eigentlich) der kompakten Repräsentation oder Codierung von (hochdimensionalen) Vektoren durch Repräsentanten, den Codebook-Vektoren.

2.3.3 k-Means

Teile jede Klasse Ω_κ in L_κ (Teil-)Gebiete, die jeweils durch ihren Schwerpunkt oder Mittelpunktvektor $\vec{\mu}^l_\kappa$ repräsentiert werden.

Im restlichen Abschnitt lassen wir die Klassenindizes κ zur Vereinfachung weg:

- sei $\Omega = \{\vec{c}_i | i = 1, \ldots, N\}$ die (einzige) Klasse
- sie wird in L disjunkte Teilgebiete R^l zerlegt:
  $\Omega = R^1 \dot\cup R^2 \dot\cup \ldots \dot\cup R^L$
- jedes Teilgebiet R^l wird durch seinen Schwerpunkt repräsentiert:
  $\vec{\mu}^l = \frac{1}{|R^l|} \sum_{\vec{c}_i \in R^l} \vec{c}_i$
- Ziel ist Minimierung des Quantisierungsfehlers

$$\epsilon = \frac{1}{N} \sum_{l=1}^{L} \sum_{\vec{c}_i \in R^l} d(\vec{c}_i, \vec{\mu}^l)$$

- wähle aufgrund von Vorwissen oder zufällig (z.B. die ersten L Vektoren der Stichprobe mit der Größe N) initiale Mittelpunkte $\vec{\mu}^l, l = 1, \ldots, L$
- $\epsilon^0 := \infty$; der Quantisierungsfehler in der Iteration 0
  $t = 0$; Iterationszähler
  - $t := t + 1, \quad \epsilon^{(t)} := 0$
  - FOR alle Gebiete $l = 1, \ldots, L$
    - $N^l := 0; \quad \hat{\vec{\mu}}^l := \vec{0}$
  - FOR alle Vektoren $\vec{c}_i$ der Stichprobe
    - bestimme $\vec{\mu}^l$ mit minimalem Abstand zu $\vec{c}_i$
    - $\epsilon^{(t)} := \epsilon^{(t)} + d\left(\vec{c}_i, \vec{\mu}^l\right)$
    - berechne neuen Schätzwert für den Mittelpunkt $\hat{\vec{\mu}}^l := \hat{\vec{\mu}}^l + \vec{c}_i$
    - $N^l := N^l + 1$
  - $\epsilon^{(t)} := \epsilon^{(t)}/N$
  - FOR alle Gebiete $l = 1, \ldots, L$
    - $\vec{\mu}^l := \hat{\vec{\mu}}^l / N^l$
- UNTIL $(\epsilon^{(t-1)} - \epsilon^{(t)})/\epsilon^{(t)} \leq \varepsilon$

2.3.4 LBG-Verfahren

bestimme Anzahl der Gebiete je Klasse mit dem LBG-Verfahren auf der Basis von K-means

Anzahl der Gebiete ist $L = 1$, wobei das Gebiet R_1 durch $\vec{\mu}_1 = \frac{1}{N} \sum\limits_{i=1}^{N} \vec{c}_i$ repräsentiert wird		
	FOR alle Gebiete R^l (oder für die $\tilde{L}$ Gebiete mit größtem Quantisierungsfehler)	
		zerlege R^l in 2 Gebiete mit den Mittelpunkten $\vec{\mu}^l + \vec{\delta}$ und $\vec{\mu}^l - \vec{\delta}$ ($\vec{\delta}$ ist zufälliger betragsmäßig kleiner Vektor)
	$L := L * 2 \quad (L := L + \tilde{L})$	
	wende k-means zur Optimierung von L Gebieten an	
UNTIL Quantisierungsfehler $\epsilon^{(t)} \leq \varepsilon$ ODER $L \geq$ maximale Klassenanzahl		

Wenn die Mittelpunktvektoren gelernt sind, erfolgt die Klassifikation von $\vec{c}$ nach Ω_κ, falls

$$\vec{\mu}^l_\kappa = min_{\lambda,j} \vec{\mu}^j_\lambda$$

2.3.5 Learning-Vector-Quantization

betrachte beim Lernen die gesamte Stichprobe (nicht Klassen unabhängig)

LVQ1-Algorithmus

wähle auf der Basis einer klassifizierten Stichprobe L Vektoren $\vec{\mu}_l$ als initiale Mittelpunkte, d.h. für jeden Vektor $\vec{\mu}_l$ ist die zugehörige Klasse bekannt (z.B. berechne je Klasse ω_κ L/K Mittelpunkte $\vec{\mu}^l$ mit Hilfe des k-means-Algorithmus aus Abschnitt **??**)		
Iterationszähler $t = 0$		
	wähle zufällig $\vec{c}$ aus der Stichprobe aus	
	bestimme $\vec{\mu}^l$ mit kleinstem Abstand zu $\vec{c}$	
	IF	$\vec{\mu}^l$ und $\vec{c}$ sind aus derselben Klasse
	THEN	$\vec{\mu}^l := \vec{\mu}^l + \alpha(t)[\vec{c} - \vec{\mu}^l]$
	ELSE	$\vec{\mu}^l := \vec{\mu}^l - \alpha(t)[\vec{c} - \vec{\mu}^l]$
	$t := t + 1$	
UNTIL $t > \beta * L$		

LVQ2-Algorithmus

initialisiere L Vektoren $\vec{\mu}_l$ (analog zu LVQ1)		
Iterationszähler $t = 0$		
	wähle zufällig $\vec{c}_i$ aus der Stichprobe aus	
	bestimme die beiden nächsten Nachbarn $\vec{\mu}_l$ und $\vec{\mu}_k$	
	IF	$\vec{\mu}_l$ gehört zur richtigen Klasse UND $\vec{\mu}_k$ zur falschen Klasse UND $\min\left(\frac{m(\vec{c}_i,\vec{\mu}_l)}{m(\vec{c}_i,\vec{\mu}_k)}, \frac{m(\vec{c}_i,\vec{\mu}_k)}{m(\vec{c}_i,\vec{\mu}_l)}\right) > \frac{1-w}{1+w}$ mit $0.2 < w < 0.3$
	THEN	$\vec{\mu}_l := \vec{\mu}_l + \alpha(t)[\vec{c}_i - \vec{\mu}_l]$
		$\vec{\mu}_k := \vec{\mu}_k - \alpha(t)[\vec{c}_i - \vec{\mu}_k]$
	$t := t + 1$	
UNTIL $t > \beta * L$		

LVQ3-Algorithmus

initialisiere L Vektoren $\vec{\mu}_l$ (analog zu LVQ1)		
Iterationszähler $t = 0$		
	wähle zufällig $\vec{c}_i$ aus der Stichprobe aus	
	bestimme die beiden nächsten Nachbarn $\vec{\mu}_l$ und $\vec{\mu}_k$	
	IF	$\vec{\mu}_l$ gehört zur richtigen Klasse UND $\vec{\mu}_k$ zur falschen Klasse UND $\min\left(\frac{m(\vec{c}_i,\vec{\mu}_l)}{m(\vec{c}_i,\vec{\mu}_k)}, \frac{m(\vec{c}_i,\vec{\mu}_k)}{m(\vec{c}_i,\vec{\mu}_l)}\right) > \frac{1-w}{1+w}$ mit $0.2 < w < 0.3$
	THEN	$\vec{\mu}_l := \vec{\mu}_l + \alpha(t)[\vec{c}_i - \vec{\mu}_l]$
		$\vec{\mu}_k := \vec{\mu}_k - \alpha(t)[\vec{c}_i - \vec{\mu}_k]$
	IF	$\vec{\mu}_l$ und $\vec{\mu}_k$ gehören zur richtigen Klasse
	THEN	$\vec{\mu}_l := \vec{\mu}_l + \varepsilon\alpha(t)[\vec{c}_i - \vec{\mu}_l]$
		$\vec{\mu}_k := \vec{\mu}_k + \varepsilon\alpha(t)[\vec{c}_i - \vec{\mu}_k]$
	$t := t + 1$	
UNTIL $t > \beta * L$		

mit $0.1 < \varepsilon < 0.3$

2.3.6 Bemerkungen

1. die Wahl guter Merkmale ist schwierig (Postulat 2)
2. gestörte Muster
3. optimale Klassifikatoren
4. Bei der (deterministischen) Klassifikation wird jeweils der Merkmlasraum in Klassengebiete (ev. inklusive Rückweisungsklasse) partitioniert. Hier finden wir die Partitionierung des Problemkreises wieder, wobei die aber i.A. der Problemkreis nur einen Teil des vollständigen Merkmalsraumes abdeckt.