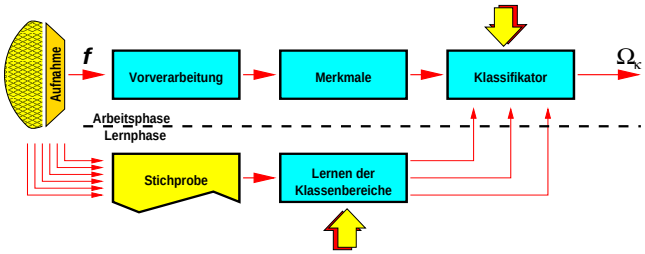
Vorlesung im Sommersemester

MUSTERERKENNUNG

E.G. Schukat-Talamazzini

4. Klassifikation einfacher Muster

Entscheidungstheorie • Statistische Klassifikatoren • Verteilungsfreie Klassifikatoren • Nichtparametrische Klassifikatoren • Entwurfsmethoden • Entscheidungsbäume • Neuronale Netze • Supportvektormaschinen • Unüberwachtes Lernen • Inkrementelles Lernen



NUMERISCHE KLASSIFIKATION

- Merkmalvektoren $\boldsymbol{x} \in \mathbb{R}^D$
- Musterklassen $\Omega_1, \ldots, \Omega_K$
- Klassifikationsaufgabe:

$$\delta_{\text{Klassenname}} : \begin{cases} \mathbb{R}^D & \to \quad \{1, \ldots, K\} \\ \boldsymbol{x} & \mapsto \quad \kappa^\star = \delta(\boldsymbol{x}) \end{cases}$$

Automatische Bestimmung der (unbekannten) Klassenzugehörigkeit eines Musters aufgrund seiner Merkmalrepräsentation

- **Entwurfsziel**:
möglichst hohe Anzahl *korrekt klassifizierter* Muster

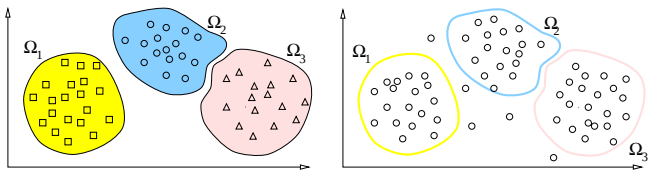
$$\delta_{\text{Klassenname}}(\boldsymbol{x}) \;=\; \kappa^\star \quad \Leftrightarrow \quad \boldsymbol{x} \in \Omega_{\kappa^\star}$$

ÜBERWACHTES LERNEN

Es steht eine etikettierte **Lernstichprobe**

$$\{\omega_\kappa \mid \kappa = 1, \ldots, K\}\,, \quad \omega_\kappa \subseteq \Omega_\kappa \;(\forall \kappa)$$

zur *Dimensionierung* des Klassifikators δ zur Verfügung.



UNÜBERWACHTES LERNEN

Die Lernstichprobe $\omega \subseteq \boldsymbol{\Omega}$ ist *nicht* etikettiert.

TESTEN

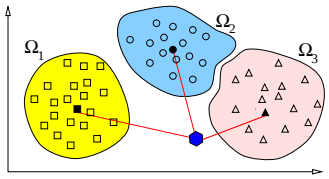
Es steht eine etikettierte **Teststichprobe** $\{\omega'_\kappa\}$
zur Evaluierung der *Fehlerrate* von δ zur Verfügung:

$$\hat{\mathrm{P}}_\varepsilon(\{\omega'_\kappa\}) \stackrel{\text{def}}{=} \frac{\text{falsch klassifiziert}}{\text{alle Muster}} = \frac{\sum_{\kappa=1}^{K} \#\{\boldsymbol{x} \in \omega'_\kappa \mid \delta(\boldsymbol{x}) = \kappa\}}{\sum_{\kappa=1}^{K} |\omega'_\kappa|}$$

Abstandsklassifikatoren

Wähle geeignete Klassenprototypen $\boldsymbol{\mu}_\kappa$ und setze

$$\delta(\boldsymbol{x}) \;=\; \operatorname*{argmin}_{\kappa} \|\boldsymbol{x} - \boldsymbol{\mu}_\kappa\|$$



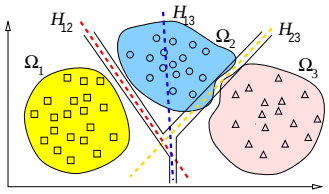
Trennfunktionen

Wähle geeignete Hyperflächen

$$h_{\kappa\lambda}(\boldsymbol{x}) \;=\; r \text{ mit } \begin{cases} r \geq 0 & \boldsymbol{x} \in \omega_\kappa \\ r < 0 & \boldsymbol{x} \in \omega_\lambda \end{cases}$$

zur Trennung der Klassengebiete Ω_κ, Ω_λ und setze

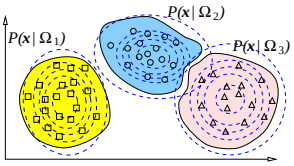
$$\delta(\boldsymbol{x}) \;=\; \kappa^\star\,, \qquad \text{falls } h_{\kappa^\star\lambda}(\boldsymbol{x}) \geq 0 \text{ für alle } \lambda \neq \kappa^\star$$

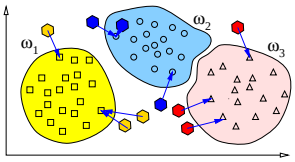


Wahrscheinlichkeitsverteilungen

Schätze die klassenbedingten Verteilungsdichten $\mathrm{P}(\boldsymbol{x}|\Omega_\kappa)$ und entscheide nach der *Bayesregel*

$$\delta(\boldsymbol{x}) \;=\; \operatorname*{argmax}_{\kappa} \mathrm{P}(\Omega_\kappa|\boldsymbol{x}) \;=\; \operatorname*{argmax}_{\kappa} \frac{\mathrm{P}(\Omega_\kappa)\cdot\mathrm{P}(\boldsymbol{x}|\Omega_\kappa)}{\mathrm{P}(\boldsymbol{x})}$$





Nächster-Nachbar-Regel

Ermittle zum Eingabemuster $\boldsymbol{x}$ den *nächsten Nachbarn*

$$\boldsymbol{x}_{\mathrm{NN}} \;=\; \operatorname*{argmin}_{z\in\bigcup\omega_\kappa} \|\boldsymbol{z}-\boldsymbol{x}\|$$

in der Lernstichprobe und setze

$$\delta(\boldsymbol{x}) \;=\; \kappa \quad \text{für} \quad \boldsymbol{x}_{\mathrm{NN}} \in \omega_\kappa$$

Funktionsapproximatoren (z.B. Neuronale Netze)

Approximiere die *ideale Diskriminanzfunktion*

$$\boldsymbol{u}_{\mathrm{ideal}}(\boldsymbol{x}) \;=\; \begin{pmatrix} u_1(\boldsymbol{x}) \\ \cdots \\ u_K(\boldsymbol{x}) \end{pmatrix} \quad \text{mit} \quad u_\kappa(\boldsymbol{x}) = \begin{cases} 1 & \boldsymbol{x}\in\Omega_\kappa \\ 0 & \boldsymbol{x}\notin\Omega_\kappa \end{cases}$$

in einer geeigneten Funktionenklasse und entscheide gemäß

$$\delta(\boldsymbol{x}) \;=\; \operatorname*{argmax}_{\kappa} \tilde{\boldsymbol{u}}_{\mathrm{ideal}}(\boldsymbol{x})$$

OPTIMALE ENTSCHEIDUNGEN

- Klassifikation eines Musters $\widehat{=}$ *Entscheidung*
- Jede Entscheidung verursacht individuelle Kosten
- ➪ Minimiere die mittleren Kosten („Risiko") des Klassifikators

Wir setzen als bekannt voraus:

- die **a priori Wahrscheinlichkeiten**
  $$\mathrm{P}(\Omega_\kappa) \;=\; p_\kappa\;, \qquad \kappa = 1,\ldots,K$$
  mit der die Muster einer Klasse auftreten,
- die **klassenbedingten** (multivariaten) **Verteilungsdichten**
  $$\mathrm{P}(\boldsymbol{x}|\Omega_\kappa)\;, \qquad \kappa = 1,\ldots,K$$
  der Merkmalvektoren von Mustern einer Klasse,
- die Matrix
  $$\begin{pmatrix} r_{1,0} & r_{1,1} & r_{1,2} & \cdots & r_{1,K} \\ \vdots & \cdots & \cdots & \ddots & \vdots \\ r_{K,0} & r_{K,1} & r_{K,2} & \cdots & r_{K,K} \end{pmatrix}$$
  der **Kosten** $r_{\kappa\lambda}$ einer Entscheidung für die Klasse Ω_λ bei Vorlage eines Musters aus Ω_κ. Wir fordern
  $$0 \le r_{\kappa\kappa} < r_{\kappa 0} < r_{\kappa\lambda} \qquad (\lambda \neq \kappa)$$

Definition 4.1 *Für einen Problemkreis mit K Musterklassen und dem Merkmalraum $\mathbb{R}^n$ heißt eine Abbildung*

$$\boldsymbol{\delta} : \begin{cases} \mathbb{R}^n & \to & \mathbb{R}^{K+1} \\ \boldsymbol{x} & \mapsto & \boldsymbol{\delta}(\boldsymbol{x}) = (\delta_0(\boldsymbol{x}), \delta_1(\boldsymbol{x}), \ldots, \delta_K(\boldsymbol{x}))^\top \end{cases}$$

Entscheidungsregel, *falls die Normierungsbedingung*

$$\sum_{\lambda=0}^{K} \delta_\lambda(\boldsymbol{x}) \;=\; 1\;, \quad (\kappa = 1,\ldots,K)$$

gilt. Die Regel heißt **scharf** *(hart), wenn alle $\boldsymbol{\delta}(\boldsymbol{x})$ Einheitsvektoren sind; andernfalls heißt die Regel* **unscharf** *(weich, randomisiert).*

Wir schreiben scharfe Entscheidungsregeln auch in der Form

$$\delta : \mathbb{R}^n \;\to\; \{0,1,\ldots,K\}$$

Eine unscharfe Regel liefert zum Beispiel Aussagen der Form

$$\boldsymbol{\delta}(\boldsymbol{x}) \;=\; (1/10, 2/10, 7/10, 0)^\top \qquad \text{oder} \qquad \boldsymbol{\delta}(\boldsymbol{x}) \;=\; (1/2, 1/8, 1/8, 1/4)^\top$$

Satz 4.1
Die zu erwartenden Kosten (das **Risiko***) der Klassifikation von Mustern durch die Entscheidungsregel*

$$\boldsymbol{\delta}(\boldsymbol{x}) : \mathbb{R}^n \;\to\; \mathbb{R}^{K+1}$$

auf Grundlage der Kostenmatrix $[r_{\kappa\lambda}]$ betragen

$$\mathfrak{R}(\boldsymbol{\delta}) \;=\; \sum_{\kappa=1}^{K} p_\kappa \sum_{\lambda=0}^{K} r_{\kappa\lambda} \int_{\mathbb{R}^n} \mathrm{P}(\boldsymbol{x}|\Omega_\kappa)\,\delta_\lambda(\boldsymbol{x})\,d\boldsymbol{x}$$

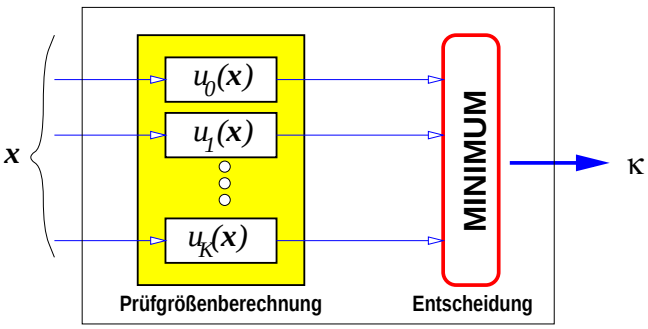
Satz 4.2 *Die optimale Entscheidungsregel $\boldsymbol{\delta}^\star$, welche das Risiko $\mathfrak{R}(\boldsymbol{\delta})$ hinsichtlich einer gegebenen Kostenmatrix $[r_{\kappa\lambda}]$ minimiert, klassifiziert die Merkmalvektoren gemäß*

$$\delta^\star_\kappa(\boldsymbol{x}) \;=\; \begin{cases} 1 & u_\kappa(\boldsymbol{x}) = \min_\lambda u_\lambda(\boldsymbol{x}) \\ 0 & \textit{sonst} \end{cases}\;, \quad \boldsymbol{x}\in\mathbb{R}^n$$

mit der **Prüfgröße**

$$u_\lambda(\boldsymbol{x}) \;\stackrel{\text{def}}{=}\; \sum_{\kappa=1}^{K} r_{\kappa\lambda} p_\kappa \mathrm{P}(\boldsymbol{x}|\Omega_\kappa)\;, \quad \lambda = 0,1,\ldots,K$$

BEMERKUNG: Diese Entscheidungsregel ist scharf!



Spezialisierung I:

$$r_{\kappa\lambda} = \begin{cases} r_w & \kappa = \lambda \\ r_z & \lambda = 0 \\ r_f & \text{sonst} \end{cases}$$

Vereinfachte Prüfgrößen

$$\begin{aligned} u_0(\boldsymbol{x}) &= r_z \cdot \sum_{\kappa=1}^{K} p_\kappa \cdot \mathrm{P}(\boldsymbol{x}|\Omega_\kappa) = r_z \cdot \mathrm{P}(\boldsymbol{x}) \\ u_\lambda(\boldsymbol{x}) &= r_f \cdot \sum_{\kappa\neq\lambda} p_\kappa \cdot \mathrm{P}(\boldsymbol{x}|\Omega_\kappa) + r_w \cdot p_\lambda \cdot \mathrm{P}(\boldsymbol{x}|\Omega_\lambda) \\ &= r_f \cdot \mathrm{P}(\boldsymbol{x}) + (r_w - r_f) \cdot p_\lambda \cdot \mathrm{P}(\boldsymbol{x}|\Omega_\lambda) \end{aligned}$$

Spezialisierung II: (erzwungene Entscheidung)

$$r_{\kappa\lambda} = \begin{cases} 0 & \kappa = \lambda \\ 1 & \kappa \neq \lambda \end{cases}$$

Dann gilt die Aussage

$\Re(\boldsymbol{\delta})$ = Klassifikationsfehlerwahrscheinlichkeit

und die Prüfgrößen lauten

$$u_\lambda(\boldsymbol{x}) = \sum_{\kappa\neq\lambda} p_\kappa \cdot \mathrm{P}(\boldsymbol{x}|\Omega_\kappa)$$

Satz 4.3 (Bayesregel)

Der Maximum a posteriori-Klassifikator

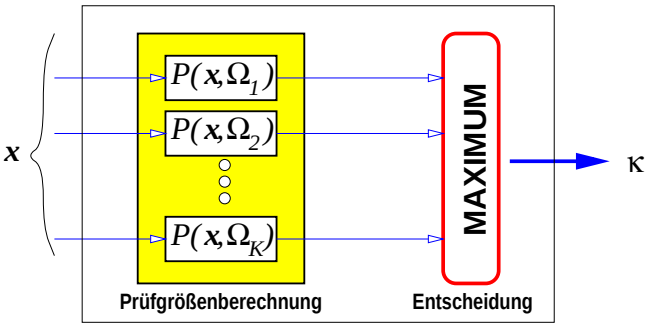
$$\delta(\boldsymbol{x}) = \underset{\lambda}{\operatorname{argmax}} \underbrace{\frac{p_\lambda \cdot \mathrm{P}(\boldsymbol{x}|\Omega_\lambda)}{\mathrm{P}(\boldsymbol{x})}}_{\mathrm{P}(\Omega_\lambda|\boldsymbol{x})}$$

ist derjenige Klassifikator mit der kleinstmöglichen Fehlerrate („Bayesfehlerrate“).

BEWEIS zur Bayesregel:

Die Spezialisierung II liefert uns den fehlerminimalen Klassifikator, und es gilt:

$$\begin{aligned} \delta(\boldsymbol{x}) &= \underset{\lambda}{\operatorname{argmin}}\, u_\lambda(\boldsymbol{x}) \\ &= \underset{\lambda}{\operatorname{argmin}} \sum_{\kappa\neq\lambda} p_\kappa \cdot \mathrm{P}(\boldsymbol{x}|\Omega_\kappa) \\ &= \underset{\lambda}{\operatorname{argmin}} \left(\mathrm{P}(\boldsymbol{x}) - p_\lambda \cdot \mathrm{P}(\boldsymbol{x}|\Omega_\lambda)\right) \\ &= \underset{\lambda}{\operatorname{argmax}} \underbrace{\left(p_\lambda \cdot \mathrm{P}(\boldsymbol{x}|\Omega_\lambda)\right)}_{\mathrm{P}(\boldsymbol{x},\Omega_\lambda)} \\ &= \underset{\lambda}{\operatorname{argmax}} \frac{p_\lambda \cdot \mathrm{P}(\boldsymbol{x}|\Omega_\lambda)}{\mathrm{P}(\boldsymbol{x})} \end{aligned}$$



Die theoretisch fundierte Mustererkennung
„lebt“ von der **BAYESREGEL**

- ⇧ minimale Fehlerrate
- ⇩ unbekannte Wahrscheinlichkeitsdichten

Statistische Klassifikatoren

- VERTEILUNGSANNAHME:
  $\mathrm{P}(\boldsymbol{x}|\Omega_\kappa)$ gehört zu einer parametrischen Familie
  $$\{\mathrm{P}(\boldsymbol{x}|\boldsymbol{\theta}) \mid \boldsymbol{\theta} \in \mathcal{M}\}$$
  multivariater Wahrscheinlichkeits(dichte)funktionen.
- Geeignete klassenbedingte Verteilungsparameter $\boldsymbol{\theta}_\kappa$ werden aus einer Lernstichprobe ω_κ geschätzt.

Verteilungsfreie Klassifikatoren

- TRENNFLÄCHEN- oder PRÜFGRÖSSENSTRUKTUR:
  Die Prüfgrößen des optimalen Klassifikators werden gemäß
  $$u_\kappa(\boldsymbol{x}) \approx u(\boldsymbol{x}|\boldsymbol{\theta}_\kappa)\ , \quad \kappa = 1, \ldots, K$$
  durch parametrische Funktionenfamilien modelliert.
- Geeignete klassenbedingte Funktionsparameter $\boldsymbol{\theta}_\kappa$ werden aus einer Lernstichprobe ω_κ geschätzt.

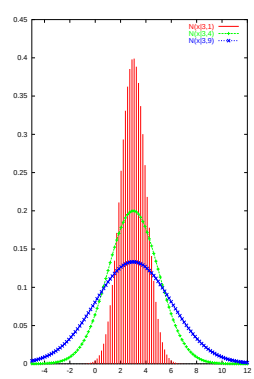
Nichtparametrische Klassifikatoren

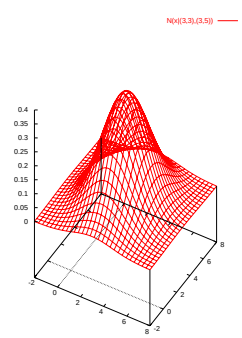
- Keine Kondensierung der Stichprobeninformation in statistische oder geometrische Parameter!
- Für jede Musterklassifikation muß die gesamte Lernstichprobe $\{\omega_\kappa\}$ herangezogen werden.

Der Normalverteilungsklassifikator

Definition 4.2
Eine stetige Zufallsvariable $\mathbb{X}$ heißt (univariat) **normalverteilt** *mit Mittelwert $\mu \in \mathbb{R}$ und Varianz $\sigma^2 \neq 0$, wenn gilt:*

$$\mathrm{f}_{\mathbb{X}}(x) = \mathcal{N}(x \mid \mu, \sigma^2) \stackrel{\text{def}}{=} \frac{1}{\sigma\sqrt{2\pi}} \cdot \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$





Definition 4.3
Ein Vektor $\mathfrak{X}$ stetiger Zufallsvariablen $\mathbb{X}_1, \ldots, \mathbb{X}_D$ heißt **multivariat normalverteilt** *mit Mittelwertvektor $\boldsymbol{\mu} \in \mathbb{R}^D$ und Kovarianzmatrix $\boldsymbol{S}$, wenn*

$$\begin{aligned} \mathrm{f}_{\mathfrak{X}}(\boldsymbol{x}) &= \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}, \boldsymbol{S}) \\ &\stackrel{\text{def}}{=} \frac{1}{\sqrt{|2\pi\boldsymbol{S}|}} \cdot \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top \boldsymbol{S}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right) \end{aligned}$$

gilt und $\boldsymbol{S}$ symmetrisch und positiv-definit ist.

Definition 4.4 *Einen Klassifikator mit den Prüfgrößen*

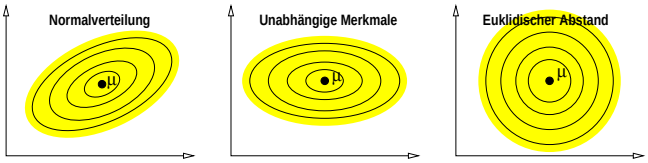
$$u_\kappa(\boldsymbol{x}) \;=\; \mathrm{P}(\boldsymbol{x}, \Omega_\kappa) \;=\; p_\kappa \cdot \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_\kappa, \boldsymbol{S}_\kappa)$$

für $\kappa = 1, \ldots, K$ *bezeichnet man als* **Normalverteilungsklassifikator** *mit den Verteilungsparametern* $[p_\kappa, \boldsymbol{\mu}_\kappa, \boldsymbol{S}_\kappa]_\kappa$.

In der Praxis verwendet man die modifizierten Prüfgrößen

$$u_\kappa(\boldsymbol{x}) \;=\; -2\log\left(\mathrm{P}(\boldsymbol{x}, \Omega_\kappa)\right)$$

die wegen des Minuszeichens allerdings zu *minimieren* sind.



Prüfgrößen

Volle Kovarianzmatrizen
(paarweise Merkmalabhängigkeiten)

$$u_\kappa(\boldsymbol{x}) \;=\; \underbrace{-2\log p_\kappa + \log|2\pi\boldsymbol{S}_\kappa|}_{\gamma_\kappa} + (\boldsymbol{x}-\boldsymbol{\mu}_\kappa)^\top \boldsymbol{S}_\kappa^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_\kappa)$$

Diagonale Kovarianzmatrizen
(paarweise Merkmalunkorreliertheit)

$$u_\kappa(\boldsymbol{x}) \;=\; \gamma_\kappa + \sum_{d=1}^{D}\left(\frac{x_d - \mu_{\kappa,d}}{\sigma_{\kappa,d}}\right)^2$$

Einheitskovarianzmatrizen
(Merkmale unkorreliert und normiert)

$$u_\kappa(\boldsymbol{x}) \;=\; \gamma_\kappa + \|\boldsymbol{x}-\boldsymbol{\mu}_\kappa\|^2$$

VEREINFACHTE KOVARIANZMATRIZEN

Mahalanobis-Klassifikator ($\boldsymbol{S}_\kappa \equiv \boldsymbol{S}$)
(alle Kovarianzmatrizen sind gleich)

$$u_\kappa(\boldsymbol{x}) \;=\; \gamma_\kappa + (\boldsymbol{x}-\boldsymbol{\mu}_\kappa)^\top \boldsymbol{S}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_\kappa)$$

Richter-Klassifikator ($\boldsymbol{S}_\kappa = \alpha_\kappa \boldsymbol{S}$)
(mit $\alpha_\kappa \in \mathbb{R}$ skalierte Kovarianzmatrizen)

$$u_\kappa(\boldsymbol{x}) \;=\; \gamma_\kappa + \alpha_\kappa^{-1} \cdot (\boldsymbol{x}-\boldsymbol{\mu}_\kappa)^\top \boldsymbol{S}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_\kappa)$$

Eigenraum-Klassifikator ($\boldsymbol{S}_\kappa = \boldsymbol{U}\boldsymbol{D}_\kappa\boldsymbol{U}^\top$)
(Hauptachsenskalierung durch Diagonalmatrizen $\boldsymbol{D}_\kappa$)

$$u_\kappa(\boldsymbol{x}) \;=\; \gamma_\kappa + (\boldsymbol{U}^\top(\boldsymbol{x}-\boldsymbol{\mu}_\kappa))^\top \boldsymbol{D}_\kappa^{-1}(\boldsymbol{U}^\top(\boldsymbol{x}-\boldsymbol{\mu}_\kappa))$$

Dyadischer Klassifikator ($\boldsymbol{S}_\kappa = \boldsymbol{D}_\kappa\boldsymbol{S}\boldsymbol{D}_\kappa$)
(Skalierung = komponentenweises Produkt von $\boldsymbol{S}$ und $\boldsymbol{d}_\kappa\boldsymbol{d}_\kappa^\top$)

$$u_\kappa(\boldsymbol{x}) \;=\; \gamma_\kappa + (\boldsymbol{x}-\boldsymbol{\mu}_\kappa)^\top \boldsymbol{D}_\kappa^{-1}\boldsymbol{S}^{-1}\boldsymbol{D}_\kappa^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_\kappa)$$

Hauptachsenklassifikator ($\boldsymbol{S}_\kappa = \sum_{d=1}^{D'} \lambda_{\kappa,d}\boldsymbol{v}_{\kappa,d}\boldsymbol{v}_{\kappa,d}^\top$)
(unvollständige Entwicklung der $\boldsymbol{S}_\kappa$)

$$u_\kappa(\boldsymbol{x}) \;=\; \gamma_\kappa + \sum_{d=1}^{D'} \lambda_{\kappa,d}^{-1} \cdot (\boldsymbol{v}_{\kappa,d}^\top(\boldsymbol{x}-\boldsymbol{\mu}_\kappa))^2$$

Maximum-Likelihood-Schätzung (ML)

Parametrische Verteilungsdichtefamilie

$$\{\mathrm{f}(x|\boldsymbol{\theta}) \mid \boldsymbol{\theta} \in \mathcal{M}\}$$

(logarithmierte) Stichprobenwahrscheinlichkeit

$$\ell_{\boldsymbol{\theta}}(\omega) \;=\; \log\prod_{x\in\omega}\mathrm{f}(x|\boldsymbol{\theta}) \;=\; \sum_{x\in\omega}\log\mathrm{f}(x|\boldsymbol{\theta})$$

Unabhängig und identisch verteilte Stichprobenelemente)

Definition 4.5
Die **Maximum-Likelihood-Schätzung** *der Parameter einer Dichtefamilie* $[\mathrm{f}(x|\boldsymbol{\theta})]$ *maximiert die parameterbedingte Stichprobenwahrscheinlichkeit, d.h. es gilt*

$$\boxed{\hat{\boldsymbol{\theta}}_{\mathrm{ML}} \;=\; \operatorname*{argmax}_{\boldsymbol{\theta}} \prod_{x\in\omega}\mathrm{f}(x|\boldsymbol{\theta})}$$

oder gleichwertig

$$\hat{\boldsymbol{\theta}}_{\mathrm{ML}} \;=\; \operatorname*{argmax}_{\boldsymbol{\theta}} \sum_{x\in\omega}\log\mathrm{f}(x|\boldsymbol{\theta})$$

Satz 4.4
Der ML-Schätzer ist **erwartungstreu**, *d.h.: ist eine Zufallsvariable* $\mathbb{X}$ *gemäß* $\mathrm{f}(x|\boldsymbol{\theta}^\star)$ *verteilt, so ist der Erwartungswert des ML-Schätzers für eine Stichprobe unabhängiger Realisierungen von* $\mathbb{X}$ *gleich* $\boldsymbol{\theta}^\star$.

Für eine repräsentative Lernstichprobe zunehmenden Umfangs strebt der ML-Schätzwert gegen den wahren Parametervektor.

ML-Schätzung für den NV-Klassifikator

Wahrscheinlichkeit einer etikettierten Stichprobe:

$$\mathrm{P}\Big(\bigcup_\kappa \omega_\kappa\Big) \;=\; \prod_{\kappa=1}^{K}\mathrm{P}(\omega_\kappa) \;=\; \prod_{\kappa=1}^{K}\prod_{x\in\omega_\kappa}\mathrm{P}(\Omega_\kappa)\cdot\mathrm{P}(\boldsymbol{x}|\Omega_\kappa)$$

Logarithmierte ML-Zielgröße ($N_\kappa = |\omega_\kappa|$):

$$\begin{aligned}\ell_{\boldsymbol{\theta}}\Big(\bigcup_\kappa \omega_\kappa\Big) &= \log\prod_{\kappa=1}^{K}\prod_{x\in\omega_\kappa} p_\kappa \cdot \mathrm{f}(\boldsymbol{x}|\boldsymbol{\theta}_\kappa)\\ &= \sum_{\kappa=1}^{K} N_\kappa \log p_\kappa + \sum_{\kappa=1}^{K}\left(\sum_{x\in\omega_\kappa}\log\mathrm{f}(\boldsymbol{x}|\boldsymbol{\theta}_\kappa)\right)\end{aligned}$$

➪ $K+1$ voneinander unabhängige Optimierungsprobleme

Satz 4.5 *Die Maximum-Likelihood-Parameter eines Normalverteilungsklassifikators bezüglich einer etikettierten Stichprobe* $[\omega_\kappa]$ *lauten*

$$\begin{aligned}\hat{p}_\kappa &= N_\kappa \Big/ \sum_{\lambda=1}^{K} N_\lambda\\ \hat{\boldsymbol{\mu}}_\kappa &= \frac{1}{N_\kappa}\sum_{x\in\omega_\kappa}\boldsymbol{x}\\ \hat{\boldsymbol{S}}_\kappa &= \frac{1}{N_\kappa}\sum_{x\in\omega_\kappa}(\boldsymbol{x}-\hat{\boldsymbol{\mu}}_\kappa)(\boldsymbol{x}-\hat{\boldsymbol{\mu}}_\kappa)^\top\\ &= \frac{1}{N_\kappa}\sum_{x\in\omega_\kappa}\boldsymbol{x}\boldsymbol{x}^\top - \hat{\boldsymbol{\mu}}_\kappa\hat{\boldsymbol{\mu}}_\kappa^\top\end{aligned}$$

Diagonale Kovarianzen:

$$\tilde{\sigma}^2_{\kappa,d} \;=\; \frac{1}{N_\kappa}\sum_{x\in\omega_\kappa}(x_d-\mu_{\kappa,d})^2$$

Bei klassenübergreifenden Kovarianzstatistiken zerfällt $\ell_\theta(\cdot)$ nicht mehr vollständig in klassenspezifische Optimierungsausdrücke!

Mahalanobis-Klassifikator:

$$\tilde{S} \;=\; S_W([\omega_\kappa]) \;=\; \frac{1}{N}\sum_{\kappa=1}^{K}\sum_{x\in\omega_\kappa}(\boldsymbol{x}-\boldsymbol{\mu}_\kappa)(\boldsymbol{x}-\boldsymbol{\mu}_\kappa)^\top$$

Einphasige Berechnung, nur *eine* Kovarianzstatistik ($\boldsymbol{S}_W = \boldsymbol{S}-\boldsymbol{S}_B$)

Richter-Klassifikator: (iterativ)

- Iterationsanfang:

$$\hat{p}_\kappa = \frac{N_\kappa}{N} \qquad \hat{\boldsymbol{\mu}}_\kappa = \frac{1}{N_\kappa}\sum_{x\in\omega_\kappa}\boldsymbol{x}$$

$$\hat{\alpha}^{(0)}_\kappa = 1 \qquad \hat{\boldsymbol{S}}_\kappa = \frac{1}{N_\kappa}\sum_{x\in\omega_\kappa}\boldsymbol{x}\boldsymbol{x}^\top - \hat{\boldsymbol{\mu}}_\kappa\hat{\boldsymbol{\mu}}_\kappa^\top$$

- Iterationsschritt:

$$\hat{\boldsymbol{S}}^{(i)} = \sum_{\kappa=1}^{K}\hat{p}_\kappa\cdot(\hat{\alpha}^{(i-1)}_\kappa)^{-1}\cdot\hat{\boldsymbol{S}}_\kappa$$

$$\hat{\alpha}^{(i)}_\kappa = \frac{1}{D}\cdot\mathrm{spur}\left(\hat{\boldsymbol{S}}_\kappa\cdot(\hat{\boldsymbol{S}}^{(i)})^{-1}\right)$$

Maximum-a posteriori-Schätzung (MAP)

Verteilungsparameter als Zufallsgröße $\Theta \in \mathcal{M}$

A priori-Verteilungsdichte für die Parameter

$$\mathrm{f}_\Theta(\boldsymbol{\theta}) \;=\; ?$$

A posteriori Wahrscheinlichkeit der Parameter

$$\mathrm{P}(\boldsymbol{\theta}|\omega) \;=\; \frac{\mathrm{P}(\boldsymbol{\theta},\omega)}{\mathrm{P}(\omega)} \;=\; \frac{\mathrm{P}(\boldsymbol{\theta})\cdot\mathrm{P}(\omega|\boldsymbol{\theta})}{\mathrm{P}(\omega)} \;=\; \frac{\mathrm{P}(\boldsymbol{\theta})\cdot\prod_{x\in\omega}\mathrm{P}(x|\boldsymbol{\theta})}{\mathrm{P}(\omega)}$$

Definition 4.6
Die **Maximum-a posteriori-Schätzung** *der Parameter einer Dichtefamilie* $[\mathrm{f}(x|\boldsymbol{\theta})]$ *unter Annahme der* **a priori**-*Verteilungsdichte* $\mathrm{f}_\Theta(\boldsymbol{\theta})$ *für* $\boldsymbol{\theta}$ *maximiert die stichprobenbedingte Wahrscheinlichkeit des gesuchten Parameterfeldes, d.h. es gilt:*

$$\hat{\boldsymbol{\theta}}_{\mathrm{MAP}} \;=\; \underset{\boldsymbol{\theta}}{\mathrm{argmax}}\left(\mathrm{f}_\Theta(\boldsymbol{\theta})\cdot\prod_{x\in\omega}\mathrm{f}_{\mathbb{X}}(x|\boldsymbol{\theta})\right)$$

- $\mathrm{f}_\Theta(\boldsymbol{\theta})$ wird als Gleichverteilung angesetzt oder orientiert sich an „Erfahrungswerten".
- Für große Stichproben strebt $\hat{\boldsymbol{\theta}}_{\mathrm{MAP}}$ gegen $\hat{\boldsymbol{\theta}}_{\mathrm{ML}}$.
- Die analytische Optimierung der MAP-Zielfunktion erfordert eine geeignete Form der a priori-Dichte ($\rightsquigarrow$ *konjugierte* Familien).

Satz 4.6
Die Mustervektoren $\boldsymbol{x}\in\mathbb{R}^D$ *eines numerischen Klassifikationsproblems seien klassenweise normalverteilt mit den unbekannten Parametern* $(p_\kappa,\boldsymbol{\mu}_\kappa,\boldsymbol{S}_\kappa)$, $\kappa=1,\ldots,K$. *Dabei folgen die Klassenwahrscheinlichkeiten* p_κ *einer* **Dirichlet**-*Verteilung*

$$\mathcal{D}(\boldsymbol{p}|\boldsymbol{q}) \;\propto\; \prod_{\kappa=1}^{K}p_\kappa^{(q_\kappa-1)}$$

mit den Hyperparametern $q_\kappa>0$, *die Mittelwertvektoren* $\boldsymbol{\mu}_\kappa$ *einer Normalverteilung*

$$\mathcal{N}(\boldsymbol{\mu}_\kappa \mid \boldsymbol{m}_\kappa,\tau_\kappa^{-1}\boldsymbol{S}_\kappa)$$

mit den Hyperparametern $\boldsymbol{m}_\kappa\in\mathbb{R}^D$ *und* $\tau_\kappa>0$, *und die inversen Kovarianzmatrizen* $\boldsymbol{S}_\kappa^{-1}$ *einer* **Wishart**-*Verteilung*

$$\mathcal{W}(\boldsymbol{S}_\kappa^{-1}\mid\alpha_\kappa,\boldsymbol{W}_\kappa) \;\propto\; |\boldsymbol{S}_\kappa|^{(D-\alpha_\kappa)/2}\cdot\exp\left(-\tfrac{1}{2}\mathrm{spur}\left(\boldsymbol{W}_\kappa\cdot\boldsymbol{S}_\kappa^{-1}\right)\right)$$

mit den Hyperparametern $\alpha_\kappa>D-1$ *und* $\boldsymbol{W}_\kappa\in\mathbb{R}^{D\times D}$; *die Matrizen* $\boldsymbol{W}_\kappa$ *seien alle positiv-definit.*

Die Maximum-a posteriori-Parameter des Normalverteilungsklassifikators bezüglich einer etikettierrten Lernstichprobe $[\omega_\kappa]$ *lauten dann:*

$$\hat{p}_\kappa = \frac{q_\kappa-1+N_\kappa}{Q-K+N}\,, \qquad Q=\sum_\kappa q_\kappa$$

$$\hat{\boldsymbol{\mu}}_\kappa = \frac{1}{\tau_\kappa+N_\kappa}\left(\tau_\kappa\boldsymbol{m}_\kappa+\sum_{x\in\omega_\kappa}\boldsymbol{x}\right)$$

$$\hat{\boldsymbol{S}}_\kappa = \frac{\boldsymbol{W}_\kappa+\tau_\kappa(\hat{\boldsymbol{\mu}}_\kappa-\boldsymbol{m}_\kappa)(\hat{\boldsymbol{\mu}}_\kappa-\boldsymbol{m}_\kappa)^\top+\sum_{x\in\omega_\kappa}\boldsymbol{x}\boldsymbol{x}^\top-N_\kappa\hat{\boldsymbol{\mu}}_\kappa\hat{\boldsymbol{\mu}}_\kappa^\top}{\alpha_\kappa-D+N_\kappa}$$

PLUG-IN-Prinzip:

- Verteilungsannahme

$$\mathrm{f}_{\mathbb{X}}(x) \;\approx\; \mathrm{f}(x|\boldsymbol{\theta})\,, \qquad \boldsymbol{\theta} = \text{„wahre" Parameter}$$

- Einsetzen der Schätzwerte

$$\mathrm{f}_{\mathbb{X}}(x) \;\approx\; \mathrm{f}(x|\hat{\boldsymbol{\theta}}_{\mathrm{ML}}) \quad \text{oder} \quad \mathrm{f}_{\mathbb{X}}(x) \;\approx\; \mathrm{f}(x|\hat{\boldsymbol{\theta}}_{\mathrm{MAP}})$$

BAYES-Schätzung

$$\begin{aligned}\mathrm{P}(x|\omega) &= \int_{\mathcal{M}}\mathrm{P}(x,\boldsymbol{\theta}\mid\omega)\,d\boldsymbol{\theta}\\ &= \int_{\mathcal{M}}\mathrm{P}(x\mid\boldsymbol{\theta},\omega)\cdot\mathrm{P}(\boldsymbol{\theta}\mid\omega)\,d\boldsymbol{\theta}\\ &= \int_{\mathcal{M}}\underbrace{\mathrm{P}(x\mid\boldsymbol{\theta})}_{\mathrm{f}(x|\boldsymbol{\theta})}\cdot\underbrace{\frac{\mathrm{P}(\omega|\boldsymbol{\theta})\cdot\mathrm{P}(\boldsymbol{\theta})}{\mathrm{P}(\omega)}}_{\text{a posteriori}}\,d\boldsymbol{\theta}\end{aligned}$$

d.h. $\mathrm{P}(x|\omega)$ ist eine Linearkombination von (unendlich vielen) konkurrierenden Wahrscheinlichkeitsmodellen

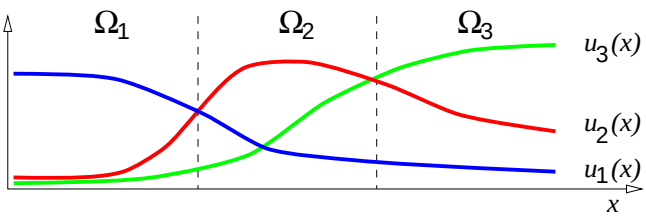
BAYES-Approximation

$\mathrm{P}(\boldsymbol{\theta})$ sei unbekannt $\rightsquigarrow$ Gleichverteilungsannahme

$$\mathrm{P}(x|\omega) \;=\; \frac{\mathrm{P}(\boldsymbol{\theta})}{\mathrm{P}(\omega)}\int_{\mathcal{M}}\mathrm{P}(x|\boldsymbol{\theta})\cdot\mathrm{P}(\omega|\boldsymbol{\theta})\,d\boldsymbol{\theta} \;=\; \frac{\int_{\mathcal{M}}\mathrm{P}(\omega,x|\boldsymbol{\theta})\,d\boldsymbol{\theta}}{\int_{\mathcal{M}}\mathrm{P}(\omega|\boldsymbol{\theta})\,d\boldsymbol{\theta}}$$

$$\approx\; \frac{\max_{\boldsymbol{\theta}\in\mathcal{M}}\mathrm{P}(\omega,x|\boldsymbol{\theta})}{\max_{\boldsymbol{\theta}\in\mathcal{M}}\mathrm{P}(\omega|\boldsymbol{\theta})} \;\approx\; \frac{\prod_{y\in\omega,x}\mathrm{f}(y\mid\hat{\boldsymbol{\theta}}_{\mathrm{ML}}(\omega,x))}{\prod_{y\in\omega}\mathrm{f}(y\mid\hat{\boldsymbol{\theta}}_{\mathrm{ML}}(\omega))}$$

Verteilungsfreie Klassifikatoren



KLASSIFIKATIONSREGEL

$$\delta(\boldsymbol{x}) \;=\; \operatorname*{argmax}_{\kappa} u_\kappa(\boldsymbol{x})$$

Statistisches Modell mit Parametrisierung

$$u_\kappa(\boldsymbol{x}) \;=\; \mathsf{P}(\Omega_\kappa|\boldsymbol{x}) \;\propto\; p_\kappa \cdot \mathsf{f}(\boldsymbol{x}|\boldsymbol{\theta}_\kappa)$$

Direkte Parametrisierung der Trennfunktionen

$$u_\kappa(\boldsymbol{x}) \;=\; h(\boldsymbol{x}|\boldsymbol{\theta}_\kappa)$$

Funktionale Form der parametr. Trennfunktionen

- LINEAR in den D Merkmalen

$$u_\kappa(\boldsymbol{x}) \;=\; \boldsymbol{a}_\kappa^\top \boldsymbol{x} \;=\; \sum_{d=1}^{D} a_{\kappa,d} \cdot x_d$$

- LINEAR in I Merkmaltermen

$$u_\kappa(\boldsymbol{x}) \;=\; \boldsymbol{a}_\kappa^\top \boldsymbol{\phi}(\boldsymbol{x}) \;=\; \sum_{i=1}^{I} a_{\kappa,i} \cdot \phi_i(\boldsymbol{x})$$

- NICHTLINEAR (z.B. Mehrschichtenperzeptron)

Linearer, verteilungsfreier Klassifikator:

$$\boldsymbol{x} \in \mathbb{R}^D \;\Rightarrow\; \boldsymbol{y} = \boldsymbol{\phi}(\boldsymbol{x}) = \begin{pmatrix} \phi_1(\boldsymbol{x}) \\ \phi_2(\boldsymbol{x}) \\ \dots \\ \phi_I(\boldsymbol{x}) \end{pmatrix}$$

$$\Rightarrow\; \boldsymbol{u} = \boldsymbol{A}^\top \boldsymbol{y} = \begin{pmatrix} u_1(\boldsymbol{y}) & = & \boldsymbol{a}_1^\top \boldsymbol{y} \\ u_2(\boldsymbol{y}) & = & \boldsymbol{a}_2^\top \boldsymbol{y} \\ \dots & = & \dots \\ u_K(\boldsymbol{y}) & = & \boldsymbol{a}_K^\top \boldsymbol{y} \end{pmatrix}$$

$$\Rightarrow\; \operatorname{argmax}_\kappa u_\kappa \in \{1, \dots, K\}$$

Nach welchem Gütekriterium optimieren wir die freien Parameter

$$\boldsymbol{A} \;=\; (\boldsymbol{a}_1, \dots, \boldsymbol{a}_K) \;\in\; \mathbb{R}^{I \times K}$$

des Klassifikators ?

Definition 4.7 *Es sei $\Omega_1, \dots, \Omega_K \subseteq \mathbb{R}^D$ ein Klassifikationsproblem. Die Abbildungen*

$$d_\kappa : \begin{cases} \mathbb{R}^D & \to \quad \mathbb{R} \\ \boldsymbol{x} & \mapsto \quad \begin{cases} 1 & \boldsymbol{x} \in \Omega_\kappa \\ 0 & \boldsymbol{x} \notin \Omega_\kappa \end{cases} \end{cases}$$

heißen **ideale Trennfunktionen** *des Problems.*

BEMERKUNGEN

- Die ideale Trennfunktion identifiziert korrekt die Klasse eines Musters.
- Genaugenommen ist bestenfalls $d_\kappa(\boldsymbol{x}) = \mathsf{P}(\Omega_\kappa|\boldsymbol{x})$ möglich.
- Analog sei die **empirische** ideale Trennfunktion einer klassifizierten Stichprobe definiert.

Der Quadratmittelklassifikator

Satz 4.7
Der Klassifikator mit den linearen Trennfunktionen

$$u_\kappa(\boldsymbol{x}) \;=\; \boldsymbol{a}_\kappa^\top \boldsymbol{\phi}(\boldsymbol{x}) \;, \quad \kappa = 1, \dots, K$$

in den Merkmaltermen

$$\phi_i : \mathbb{R}^D \;\to\; \mathbb{R} \;, \quad i = 1, \dots, I$$

welcher den erwarteten quadratischen Fehler

$$\varepsilon(\boldsymbol{A}) \;\stackrel{\text{def}}{=}\; \mathcal{E}[\|\boldsymbol{A}^\top \boldsymbol{\phi}(\mathbb{X}) - \boldsymbol{d}(\mathbb{X})\|^2]$$

zwischen parametrisierter und idealer Trennfunktion minimiert, heißt **Quadratmittelklassifikator**.

Die fehlerminimale Parametermatrix ist

$$\boldsymbol{A} \;=\; \boldsymbol{R}^{-1} \boldsymbol{M}$$

mit den Erwartungswertmatrizen

$$\begin{aligned} \boldsymbol{R} &= \mathcal{E}[\boldsymbol{\phi}(\mathbb{X})\boldsymbol{\phi}(\mathbb{X})^\top] \;, & \boldsymbol{R} \in \mathbb{R}^{I \times I} \\ \boldsymbol{M} &= \mathcal{E}[\boldsymbol{\phi}(\mathbb{X})\boldsymbol{d}(\mathbb{X})^\top] \;, & \boldsymbol{M} \in \mathbb{R}^{I \times K} \end{aligned}$$

(sofern $\boldsymbol{R}$ invertierbar ist).

Liegt eine etikettierte Lernstichprobe $\omega_1, \dots, \omega_K$ vor, so können die Erwartungswerte durch empirische Mittelwerte ersetzt werden; für die Matrizen $\boldsymbol{R}$, $\boldsymbol{M}$ ergeben sich die Schätzwerte

$$\begin{aligned} \hat{\boldsymbol{R}} &= \hat{\boldsymbol{S}} + \hat{\boldsymbol{\mu}}\hat{\boldsymbol{\mu}}^\top \\ \hat{\boldsymbol{M}} &= (\hat{p}_1 \hat{\boldsymbol{\mu}}_1, \dots, \hat{p}_K \hat{\boldsymbol{\mu}}_K) \end{aligned}$$

aus den Statistiken der ϕ-transformierten Stichprobe.

Nichtparametrische Klassifikatoren

- Schätze $\mathsf{P}(\boldsymbol{x}|\Omega_\kappa)$ aufgrund *lokaler* Häufungseigenschaften der Lernstichprobe ω_κ

$$\mathsf{f}^{\text{emp}}(\boldsymbol{x}|\omega_\kappa) \;\propto\; \begin{cases} 1 & \boldsymbol{x} \in \omega_\kappa \\ 0 & \boldsymbol{x} \notin \omega_\kappa \end{cases}$$

(zum Beispiel durch Glättung der *empirischen* Dichtefunktion)

⇧ keine Annahmen über Verteilungen/Trennfunktionen

⇩ gewaltiger Rechen- und Speicheraufwand

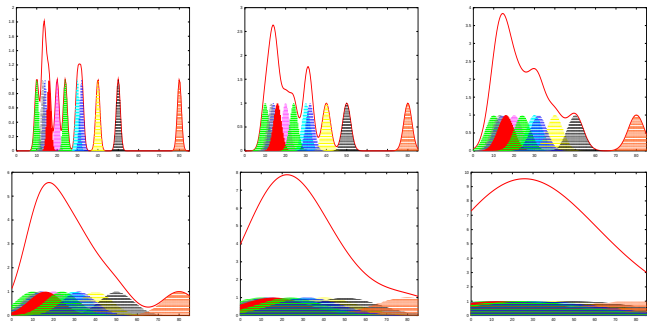
PARZEN-Schätzung

$$\hat{\mathsf{P}}(\boldsymbol{x}|\Omega_\kappa) \;=\; \frac{1}{N_\kappa} \sum_{\boldsymbol{z} \in \omega_\kappa} g(\boldsymbol{x} - \boldsymbol{z})$$

Die **Potentialfunktion** $g : \mathbb{R}^D \to \mathbb{R}_0^+$ ist normiert

$$\int_{\mathbb{R}^D} g(\boldsymbol{\xi})\, d\boldsymbol{\xi} \;=\; 1$$

und hat ihre Masse um den Ursprung $\boldsymbol{0} \in \mathbb{R}^D$ konzentriert.



Satz 4.8
Die multivariate Zufallsvariable $\mathbb{X} \in \mathbb{R}^D$ sei gemäß der stetigen Dichte $\mathsf{f}_{\mathbb{X}}$ verteilt und $\omega = \{z_n | n \in \mathbb{N}\}$ sei eine Folge zufällig und unabhängig gezogener Stichprobenelemente.

Genügt die normierte und beschränkte Potentialfunktion $g : \mathbb{R}^D \to \mathbb{R}_0^+$ der Bedingung

$$\lim_{|x|\to\infty} g(x) \cdot \prod_{d=1}^{D} x_d \;=\; 0$$

und ist $[h_\nu]$ eine positive reelle Zahlenfolge mit den Eigenschaften

$$\lim_{\nu\to\infty} h_\nu^q = 0 \quad \textit{und} \quad \lim_{\nu\to\infty} \nu h_\nu^q = \infty\,, \qquad (\forall q \in \mathbb{N})$$

dann konvergiert die Parzen-Schätzung

$$\hat{\mathsf{P}}_N(x) \;=\; \frac{1}{N}\sum_{n=1}^{N} g\left(\frac{x - z_n}{h_N}\right)$$

im quadratischen Mittel gegen die wahre Dichte $\mathsf{f}_{\mathbb{X}}$.

Parzen-Schätzung mit Gaußkern

$$g(x) \;=\; \mathcal{N}(x \mid 0, \sigma^2 E)$$

σ^2 bestimmt den Grad der Konzentration von $g(\cdot)$ um den Ursprung.

Uniforme Parzen-Schätzung

$$g(x) \;=\; \begin{cases} 1/V & x \in \mathcal{U} \\ 0 & x \notin \mathcal{U} \end{cases}$$

$\mathcal{U} \subseteq \mathbb{R}^D$ Nullpunktumgebung mit $vol(\mathcal{U}) = V$.

Die Schätzung vereinfacht sich zu

$$\hat{\mathsf{P}}(x|\Omega_\kappa) \;=\; \frac{n_\kappa(x)}{N_\kappa \cdot V}$$

und $n_\kappa(x)$ ist die Anzahl der $z \in \omega_\kappa$ mit $x - z \in \mathcal{U}$.

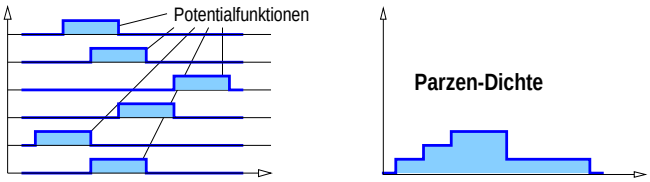
PRÜFGRÖSSE:

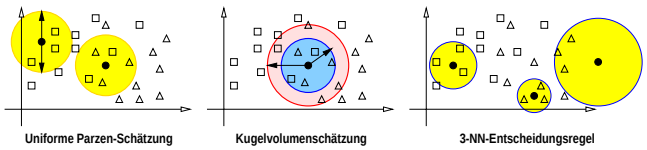
$$u_\kappa(x) \;=\; \hat{p}_\kappa \cdot \hat{\mathsf{P}}(x|\Omega_\kappa) \;=\; \frac{N_\kappa}{N} \cdot \frac{n_\kappa(x)}{N_\kappa \cdot V} \;=\; \frac{n_\kappa(x)}{N \cdot V}$$

ENTSCHEIDUNGSREGEL:

$$\delta(x) \;=\; \operatorname*{argmax}_\kappa u_\kappa(x) \;=\; \operatorname*{argmax}_\kappa n_\kappa(x)$$

(maximale Anzahl der κ-Befunde in $\mathcal{U}(x) = \{x + z \mid z \in \mathcal{U}\}$)





Volumenschätzung (Hyperkugelschätzung)

Halte $n_\kappa(x) \overset{!}{=} k$ fest — variiere V:

$$\hat{\mathsf{P}}(x|\Omega_\kappa) \;=\; \frac{k}{N_\kappa \cdot V_\kappa(x)}$$

$V_\kappa(x)$ ist das minimale Volumen einer x-Umgebung mit mindestes k Befunden aus ω_κ, z.B.

$$V_\kappa(x) \;=\; \min_\rho \{vol(\mathcal{U}_\rho(x) \mid |\mathcal{U}_\rho(x) \cap \omega_\kappa| \geq k\}$$

PRÜFGRÖSSE:

$$u_\kappa(x) \;=\; \hat{p}_\kappa \cdot \hat{\mathsf{P}}(x|\Omega_\kappa) \;=\; \frac{N_\kappa}{N} \cdot \frac{k}{N_\kappa \cdot V_\kappa(x)} \;=\; \frac{k}{N \cdot V_\kappa(x)}$$

ENTSCHEIDUNGSREGEL:

$$\delta(x) \;=\; \operatorname*{argmax}_\kappa u_\kappa(x) \;=\; \operatorname*{argmin}_\kappa V_\kappa(x)$$

(ω_κ versammelt k Muster auf kleinstem Raum um x)

Eine geeignete Wahl für die Konstante ist $k = k(\kappa) = \sqrt{|\omega_\kappa|}$.

kNN-Entscheidungsregel

Uniforme Parzen-Schätzung der a posteriori W'keiten

$$\mathsf{P}(\Omega_\kappa|x) \;=\; \frac{\mathsf{P}(\Omega_\kappa) \cdot \mathsf{P}(x|\Omega_\kappa)}{\mathsf{P}(x)}$$

Sind unter den k nächsten Nachbarn von x je $n_\kappa(x)$ Muster der Klasse κ, so läßt sich schätzen:

$$\hat{\mathsf{P}}(\Omega_\kappa|x) \;=\; \frac{\dfrac{N_\kappa}{N} \cdot \dfrac{n_\kappa(x)}{N_\kappa \cdot V}}{\dfrac{k}{N \cdot V}} \;=\; \frac{n_\kappa(x)}{k}$$

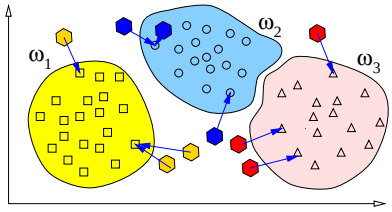
ENTSCHEIDUNGSREGEL:

$$\delta(x) \;=\; \operatorname*{argmax}_\kappa \frac{n_\kappa(x)}{k} \;=\; \operatorname*{argmax}_\kappa n_\kappa(x)$$

Nächster-Nachbar-Regel (NN) $(k = 1)$

PRÜFGRÖSSE:

$$u_\kappa(x) \;=\; \min \{\|x - z\|_r \mid z \in \omega_\kappa\}$$

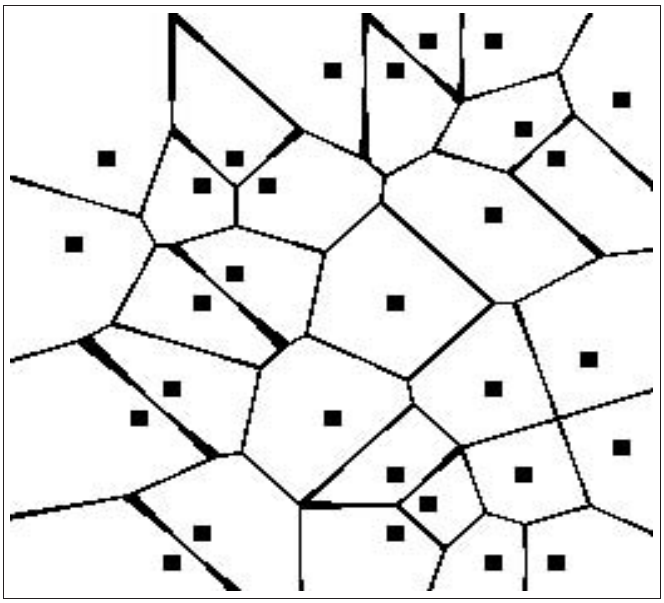


Metriken/Normen

$$\|\boldsymbol{x}-\boldsymbol{z}\|_r \stackrel{\text{def}}{=} \left(\sum_{d=1}^{D} |x_d - z_d|^r\right)^{1/r}$$

Cityblockmetrik (1)
Euklidischer Abstand (2)
Maximumnorm (∞)

VORONOI-Zerlegung



Die Fehlerrate

„Welcher Anteil an Fehlklassifikationen ist für die Entscheidungsregel δ in einer realistischen Anwendung zu erwarten?“

Definition 4.8
Es sei $\{\Omega_\kappa \mid \kappa = 1,\ldots,K\}$ ein Klassifikationsproblem, $[\omega_\kappa^{\text{lern}}]$ eine etikettierte Lernstichprobe und $\delta = \delta([\omega_\kappa^{\text{lern}}])$ eine mit diesen Daten angelernte Entscheidungsregel. Dann heißt

$$\varepsilon(\delta \mid [\omega_\kappa]) \stackrel{\text{def}}{=} \frac{1}{N}\sum_{\kappa=1}^{K}\sum_{\boldsymbol{x}\in\omega_\kappa}(1-\delta_\kappa(\boldsymbol{x}))$$

$$= \frac{\text{„Anzahl falsch klassifizierter Muster“}}{\text{„Anzahl aller Muster in } [\omega_\kappa]\text{“}}$$

die **Klassifikationsfehlerrate** *von δ auf der Stichprobe $[\omega_\kappa]$. Insbesondere heißt*

$$\varepsilon_{\text{lern}}(\delta) \stackrel{\text{def}}{=} \varepsilon(\delta|[\omega_\kappa^{\text{lern}}])$$

die **Reklassifikationsfehlerrate** *von δ.*
Das Risiko

$$\varepsilon(\delta) = \varepsilon(\delta \mid [\Omega\kappa]) = \Re(\delta)$$

aus Satz 4.1 hinsichtlich der 0/1-Kostenmatrix $[r_{\kappa\lambda}]$ bezeichnen wir als die **wahre Fehlerrate** *des vorliegenden Klasssifikationsproblems, und ihren kleinstmöglichen Wert*

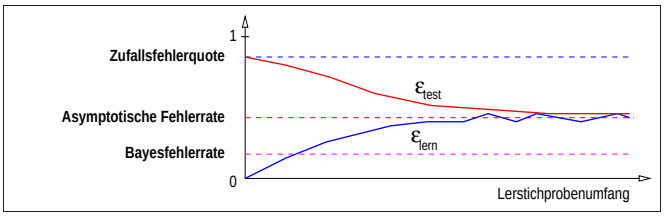
$$\varepsilon_{\text{Bayes}}([\Omega\kappa]) \stackrel{\text{def}}{=} \min_{\delta} \varepsilon(\delta \mid [\Omega\kappa])$$

als **Bayesfehlerrate**.

BEMERKUNGEN:

- Die wahre Fehlerrate liegt immer oberhalb der Bayesfehlerrate.
- Empirische Fehlerraten $\varepsilon(\delta|\omega)$ sind **Schätzungen** für $\varepsilon(\delta)$.
- **Überanpassung**: die Reklassifikationsfehlerrate $\varepsilon_{\text{lern}}(\delta)$ ist i.a. eine zu *optimistische* Schätzung von $\varepsilon(\delta)$
- Deshalb verwendet man eine **unabhängige Teststichprobe** $[\omega_\kappa^{\text{test}}]$ und schätzt

$$\widehat{\varepsilon}(\delta) = \varepsilon_{\text{test}}(\delta) = \varepsilon(\delta|[\omega_\kappa^{\text{test}}])$$



FAUSTREGEL

Der Stichprobenumfang $|\omega^{\text{lern}}|$ ist verantwortlich für die Güte des angelernten Klassifikators.

Der Stichprobenumfang $|\omega^{\text{test}}|$ ist verantwortlich für die Genauigkeit seiner geschätzten Fehlerrate.

Schätzfehler der Fehlerschätzung

Für einen Klassifikator δ mit wahrer Fehlerrate ε_δ gehorcht die diskrete Zufallsvariable

$$\mathbb{M} = \text{„Anzahl falschklassifizierter unter } N \text{ Mustern“}$$

einer **Binomialverteilung**:

$$\mathrm{P}(\mathbb{M} = n_f) = \mathcal{B}(n_f \mid \varepsilon_\delta, N) = \binom{N}{n_f}\cdot\varepsilon_\delta^{n_f}\cdot(1-\varepsilon_\delta)^{N-n_f}$$

Satz 4.9 (Binomialer Grenzwertsatz)
Ist die Zufallsvariable $\mathbb{X}_N$ für jedes $N \in \mathbb{N}$ binomialverteilt gemäß $\mathcal{B}(\cdot|N,p)$, so gilt die asymptotische Näherungsformel

$$\lim_{N\to\infty} \mathrm{P}\left(\frac{\mathbb{X}_N - \mathcal{E}[\mathbb{X}_N]}{\sqrt{\mathrm{Var}[\mathbb{X}_N]}} \le x\right) = \frac{1}{\sqrt{2\pi}}\cdot\int_{-\infty}^{x} e^{-\frac{\xi^2}{2}}\,d\xi$$

mit den Mittelwerten

$$\mu = \mathcal{E}[\mathbb{X}_N] = N\cdot p$$

und Varianzen

$$\sigma^2 = \mathrm{Var}[\mathbb{X}_N] = N\cdot p\cdot(1-p)$$

der Binomialverteilung.

Satz 4.10 (Fehlerrate der NN-Regel)
Ist $\varepsilon_{\text{Bayes}}$ die Bayesfehlerrate eines Klassifikationsproblems $[\Omega_\kappa]$ und ε_{NN} die asymptotische Fehlerrate ($|\omega| \to \infty$) des Nächster-Nachbar-Klassifikators, so gilt

$$\varepsilon_{\text{Bayes}} \le \varepsilon_{\text{NN}} \le \varepsilon_{\text{Bayes}}\cdot\left(2 - \frac{K}{K-1}\varepsilon_{\text{Bayes}}\right)$$

Für hinreichend kleine Fehlerraten folgt daraus

$$\varepsilon_{\text{NN}}/2 \le \varepsilon_{\text{Bayes}} \le \varepsilon_{\text{NN}}$$

Klassifikatorentwurfstechniken

Reklassifikation

$$\omega \;=\; \omega^{\text{lern}} \;=\; \omega^{\text{test}}$$

⇨ extrem optimistische Fehlerschätzung!

Einseitige Aufspaltung („*held-out*“)

$$\omega \;=\; \omega^{\text{lern}} \;\uplus\; \omega^{\text{test}}$$

- LERNEN — $\boldsymbol{\delta} = \boldsymbol{\delta}(\omega^{\text{lern}})$
- TESTEN — $\hat{\varepsilon} = \varepsilon(\boldsymbol{\delta}|\omega^{\text{test}})$

⇨ Verschwendung von Datenmaterial in der Lern- und Testphase

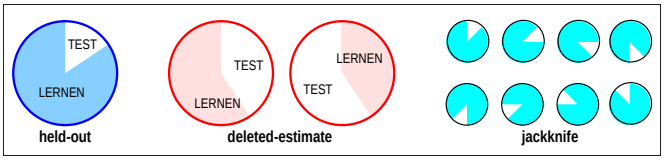
Zweiseitige Aufspaltung („*deleted-estimate*“)

$$\omega \;=\; \omega^{A} \;\uplus\; \omega^{B}$$

- LERNEN — $\boldsymbol{\delta}^{A} = \boldsymbol{\delta}(\omega^{A})$, $\boldsymbol{\delta}^{B} = \boldsymbol{\delta}(\omega^{B})$
- TESTEN — $\hat{\varepsilon} = \text{Mittelwert}(\varepsilon(\boldsymbol{\delta}^{A}|\omega^{B}), \varepsilon(\boldsymbol{\delta}^{B}|\omega^{A}))$

⇨ relativ kleine Lernstichprobe

Klassifikatorentwurfstechniken (Forts.)



Rotationsverfahren („*jackknife*“)

$$\omega \;=\; \omega^{1} \;\uplus\; \omega^{2} \;\uplus\; \ldots \;\uplus\; \omega^{R}$$

- LERNEN — für alle r: $\boldsymbol{\delta}^{r} = \boldsymbol{\delta}(\omega \backslash \omega^{r})$
- TESTEN — $\hat{\varepsilon} = {}^{1}\!/_{R} \sum \varepsilon(\boldsymbol{\delta}^{r}|\omega^{r})$

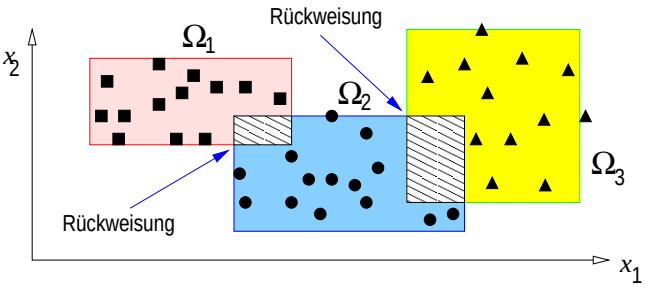
Eliminationsverfahren („*leave-one-out*“)

$$\omega \;=\; \bigcup_{\boldsymbol{x}\in\omega} \{\boldsymbol{x}\}$$

- LERNEN — Bayesapproximation
- TESTEN — analytische Berechnung falls möglich

Parallelepiped-Klassifikator

Klassengebiete ⇔ Hyperquader des $\mathbb{R}^{D}$



PRÜFGRÖSSE:

$$u_{\kappa}(\boldsymbol{x}) \;=\; \begin{cases} 1 & a^{L}_{\kappa,d} \le x_d \le a^{H}_{\kappa,d} \;\; (\forall d) \\ 0 & \text{sonst} \end{cases}$$

ENTSCHEIDUNGSREGEL

$$\delta(\boldsymbol{x}) \;=\; \begin{cases} \Omega_{\kappa} & \boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{e}_{\kappa} \text{ für ein } \kappa \\ \Omega_{0} & \|\boldsymbol{u}\| = 0 \text{ oder } \|\boldsymbol{u}\| > 1 \end{cases}$$

LERNEN der Begrenzungsparameter $\boldsymbol{a}^{L}_{\kappa}$, $\boldsymbol{a}^{H}_{\kappa}$

$$a^{L}_{\kappa,d} \;=\; \min\{x_d \mid \boldsymbol{x} \in \omega_{\kappa}\} \qquad (\forall\kappa)$$
$$a^{H}_{\kappa,d} \;=\; \max\{x_d \mid \boldsymbol{x} \in \omega_{\kappa}\} \qquad (\forall\kappa)$$

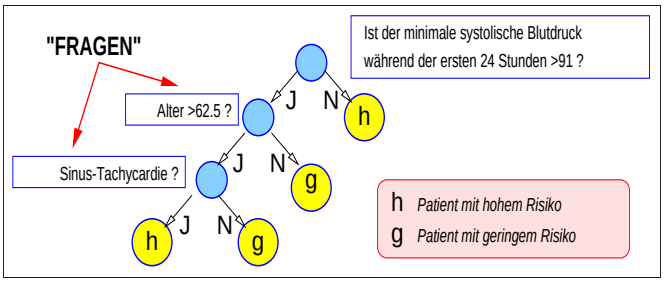
VORTEILE	NACHTEILE
extrem schnelle Lernphase effiziente Klassifikation intuitiv interpretierbare Klassengebiete	Merkmalunabhängigkeit, achsenparallele Trennflächen, unimodale Verteilung, klassenfreie Gebiete, keine Schätzung der $P(\Omega_{\kappa}\|\boldsymbol{x})$

Binärer Entscheidungsbaum

Untersuchung von Herzinfarktpatienten

19 Variablen gemessen, 30 Tahe beobachtet

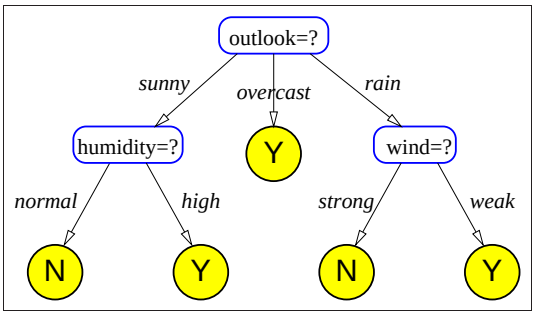
Ist ein zweiter, diesmal tödlicher Infarkt innerhalb des nächsten Monats zu befürchten?



Nichtbinärer Entscheidungsbaum

Freizeitorientierte Wetterbeobachtungen

Ist das Wetter zum Tennisspielen geeignet?

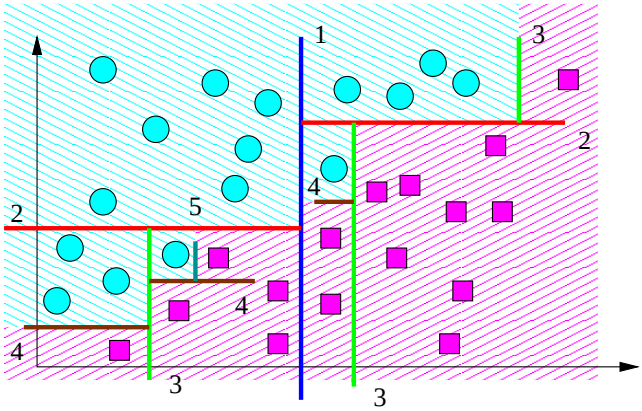


Binärer Entscheidungsbaum

- Alle Merkmale $x_1, \ldots, x_D$ sind *reellwertig*
- Rekursive Halbraumbildung durch *„Fragen“*

$$x_d \;\leq\; \theta$$

- Elementare Zellen tragen eine *Klassenkennzeichnung*



Vom Entscheidungsbaum induzierte Klassengebiete

$$\bar{\Omega}_\kappa \;=\; \bigcup_{m=1}^{M_\kappa} \bar{\Omega}_{\kappa,m} \qquad (\forall \kappa)$$

$$\bar{\Omega}_{\kappa,m} \;=\; \bigcap_{l=1}^{M_{\kappa,m}} H_{\kappa,m,l} \;=\; \text{„Halbraum } x_d \leq \theta \text{ oder } x_d > \theta \text{ “}$$

Notation

Ist $\mathcal{B}$ die Menge der **Knoten** eines Entscheidungsbaumes, so bezeichne $\beta_\triangle(\mathcal{B})$ den **Wurzelknoten** und $\mathcal{B}_\ell$ die Menge der **Blattknoten**.

Für den Knoten $\beta \in \mathcal{B}$ mit $\beta \neq \beta_\triangle$ sei $\beta^\uparrow$ der **Vorgängerknoten**, d.h. es gilt die Graphrelation $\beta^\uparrow \prec \beta$.

Ist $\mathcal{B}$ ein binärer Entscheidungsbaum und $\beta \in \mathcal{B} \backslash \mathcal{B}_\ell$ ein **innerer** Knoten, so bezeichnen β° und $\beta^\bullet$ die beiden unmittelbaren **Nachfolger** von β.

Mit jedem Knoten β eines Entscheidungsbaumes zum Problemkreis Ω ist eine **Entscheidungsfrage** (*„Dichotomie“*) $Q(\beta)$ verknüpft — für Musterräume $\Omega = \mathbb{R}^D$ betrachten wir ausschließlich Fragen der Gestalt

$$Q(\beta) : \begin{cases} \mathbb{R}^D & \to \quad \{0,1\} \\ \boldsymbol{x} & \mapsto \quad \begin{cases} 1 & x_{d(\beta)} \leq \theta_\beta \\ 0 & \text{sonst} \end{cases} \end{cases}$$

Mit jedem Blattknoten $\beta \in \mathcal{B}_\ell$ ist eine **Klassenmarkierung** $\delta_\ell(\beta)$ verknüpft:

$$\delta_\ell : \mathcal{B}_\ell \longrightarrow \{1, \ldots, K\}$$

Klassifikationsphase

Definition 4.9
Ist $(\mathcal{B}, Q, \delta_\ell)$ ein binärer Entscheidungsbaum über $\Omega = \mathbb{R}^D$ und $\omega \subseteq \Omega$, so definieren wir die **assoziierten Mustermengen** *ω_β durch:*

$$\begin{aligned} \omega(\beta_\triangle) &\stackrel{\text{def}}{=} \omega \\ \omega(\beta^\circ) &\stackrel{\text{def}}{=} \{\boldsymbol{x} \in \omega_\beta \mid x_{d(\beta)} \leq \theta_\beta\} \\ \omega(\beta^\bullet) &\stackrel{\text{def}}{=} \{\boldsymbol{x} \in \omega_\beta \mid x_{d(\beta)} \not\leq \theta_\beta\} \end{aligned}$$

Satz 4.11
Ist $(\mathcal{B}, Q, \delta_\ell)$ ein binärer Entscheidungsbaum über $\Omega = \mathbb{R}^D$, so gilt:

$$\Omega \;=\; \biguplus_{\beta \in \mathcal{B}_\ell} \Omega(\beta)$$

Der Entscheidungsbaum definiert ferner eine vollständige Zerlegung von Ω in die Klassengebiete

$$\Omega_\kappa \stackrel{\text{def}}{=} \bigcup_{\delta_\ell(\beta) = \kappa} \Omega(\beta)$$

Definition 4.10
Ein binärer Entscheidungsbaum $(\mathcal{B}, Q, \delta_\ell)$ über $\Omega = \mathbb{R}^D$ definiert eine assoziierte **Entscheidungsfunktion**

$$\delta : \begin{cases} \mathbb{R}^D & \to \quad \{1, \ldots, K\} \\ \boldsymbol{x} & \mapsto \quad \delta_\ell(\beta(\boldsymbol{x})) \end{cases}$$

Dabei bezeichne $\beta(\boldsymbol{x})$ denjenigen Blattknoten $\beta \in \mathcal{B}_\ell$ mit $\boldsymbol{x} \in \Omega(\beta)$ (das „Blatt“ von $\boldsymbol{x}$).

Klassifikation von x mittels „Durchschleusen“

1 INITIALISIERUNG: Setze $\beta = \beta_\triangle(\mathcal{B})$.

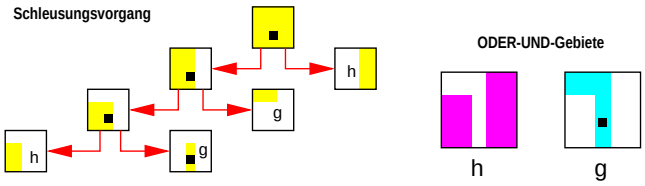
2 BEFRAGUNG:
Reiche $\boldsymbol{x}$ gemäß $Q(\beta)$ an einen Kindknoten weiter:

$$\beta \Leftarrow \begin{cases} \beta^\circ & x_d \leq \theta \\ \beta^\bullet & x_d \not\leq \theta \end{cases}$$

3 TERMINIERUNG:
Ist β ein Blattknoten, so lautet das Resultat

$$\delta(\boldsymbol{x}) \;=\; \delta_\ell(\beta)$$

Andernfalls gehe $\rightsquigarrow$ Schritt (2).



Intuitiv interpretierbare Klassenentscheidungen:

$$\begin{aligned} \Omega_h &= (\{x_b \not\leq 91\} \wedge \{x_a \not\leq 62.5\} \wedge \{x_s \not\leq 0\}) \vee (\{x_b \leq 91\}) \\ \Omega_g &= (\{x_b \not\leq 91\} \wedge \{x_a \not\leq 62.5\} \wedge \{x_s \leq 0\}) \vee (\{x_b \not\leq 91\} \wedge \{x_a \leq 62.5\}) \end{aligned}$$

mit den Variablen (Merkmalen)

$$\begin{aligned} x_b &= \text{„minimaler systolischer Blutdruck“} \\ x_a &= \text{„Alter des Patienten“} \\ x_s &= \text{„Sinus-Tachycardie? (0 oder 1)“} \end{aligned}$$

„Züchten“ eines Entscheidungsbaumes

aus einer etikettierten Lernstichprobe $\omega_1, \ldots, \omega_K$

Gierige top-down Induktion

1 INITIALISIERUNG:
Erzeuge den Wurzelknoten β mit den assoziierten Stichproben $[\omega_\kappa]$

2 STOPPTEST:
Ist β hinreichend **reinklassig**, so beende die lokale Konstruktion mit einer Blattmarkierung $\delta_\ell(\beta) = \kappa$

3 FRAGEAUSWAHL:
Wähle eine Frage $Q(\beta)$ mit maximaler Reduktion der **Entscheidungsunsicherheit**.

4 EXPANSION:
Bilde die Nachfolgerknoten β°, $\beta^\bullet$ und ihre assoziierten Stichproben $[\omega_\kappa(\beta^\circ)]$ und $[\omega_\kappa(\beta^\bullet)]$ bezüglich $Q(\beta)$.

5 REKURSION:
Fahre mit β° und $\beta^\bullet$ bei $\rightsquigarrow$ Schritt (2) fort.

- *Auswahlregel* — wie definieren wir Entscheidungsunsicherheit? Auf welche Weise finden wir (effizient) die optimale Frage?
- *Stoppkriterium* — wann ist ein Knoten reinklassig?
- *Überadaption* — wie vereiteln wir die Zersplitterung von Ω?
- *Suchproblem* — welches ist der global optimale Baum? Finden wir ihn mittels TDI? (NEIN!)

Entscheidungsunsicherheit

Lokale a posteriori Klassenwahrscheinlichkeit in Knoten β

$$p_\kappa(\beta) \;=\; \mathrm{P}(\mathbb{X} \in \Omega_\kappa \mid \mathbb{X} \in \boldsymbol{\Omega}(\beta))$$

Maximum-Likelihood-Schätzwert, Lernstichprobe $[\omega_\kappa]$

$$\hat{p}_\kappa(\beta) \;=\; \frac{\text{„Anzahl der } \Omega_\kappa\text{-Muster in } \beta\text{“}}{\text{„Anzahl aller Muster in } \beta\text{“}} \;=\; \frac{|\omega_\kappa(\beta)|}{\sum_{\lambda=1}^{K} |\omega_\lambda(\beta)|}$$

Definition 4.11
Es sei $K \in \mathbb{N}$ und $\{p_\kappa \mid \kappa = 1, \ldots, K\}$ eine diskrete Wahrscheinlichkeitsverteilung. Eine Abbildung

$$\Im : \{p_1, \ldots, p_K\} \mapsto u \in \mathbb{R}$$

heißt Maß für die **Entscheidungsunsicherheit** *(Inhomogenität, „impurity“), falls gilt:*

- *Die Größe $\Im(\cdot)$ ist nichtnegativ.*
- *$\Im(\cdot)$ ist maximal für die Gleichverteilung $p_\kappa \equiv 1/K$*
- *$\Im(\cdot)$ ist minimal für die definiten Verteilungen*

$$\boldsymbol{e}_\lambda \;=\; (0, \ldots, 0, p_\lambda = 1, 0, \ldots, 0)\;, \quad \lambda \in \{1, \ldots, K\}$$

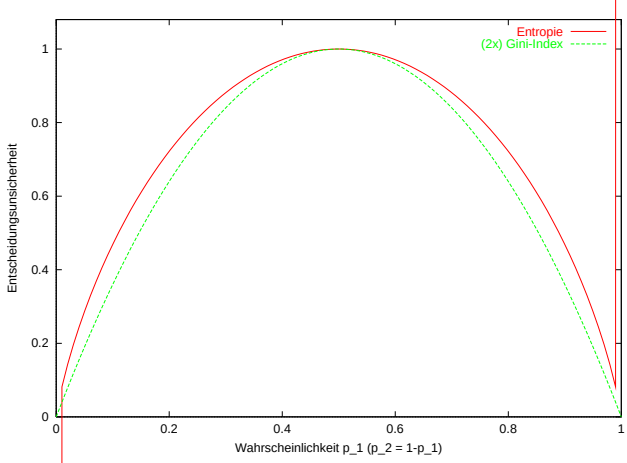
Die Fragen in den inneren Knoten des Entscheidungsbaumes sind so zu wählen, daß die Entscheidungsunsicherheit möglichst wirkungsvoll reduziert wird ohne den Merkmalraum zu stark zu fragmentieren.

Gini-Index

$$\Im([p_\kappa]) \;=\; 1 - \sum_{\kappa=1}^{K} p_\kappa^2 \;=\; 1 - \|\boldsymbol{p}\|^2$$

Entropie

$$\Im([p_\kappa]) \;=\; \mathcal{H}([p_\kappa]) \;=\; \mathcal{E}[-\log_2 \mathrm{P}(\mathbb{K})] \;=\; -\sum_{\kappa=1}^{K} p_\kappa \log_2 p_\kappa$$



Der Maximalwert des Gini-Index ist $(K-1)/K$, die maximale Entropie ist $\log_2 K$.

Reduktion der Entscheidungsunsicherheit

Entscheidungsunsicherheit im Knoten β

$$\Im(\beta) \;=\; \Im(p_1(\beta), \ldots, p_K(\beta))$$

Lokale **Verzweigungswahrscheinlichkeiten** in Knoten β

$$p(\beta'|\beta) \;=\; \mathrm{P}(\mathbb{X} \in \boldsymbol{\Omega}(\beta') \mid \mathbb{X} \in \boldsymbol{\Omega}(\beta))\;, \quad \beta \prec \beta'$$

Erwartete E.U. durch Stellen der Frage Q in β

$$\Im(\beta|Q) \;=\; \sum_i p(\beta_i|\beta) \cdot \Im(\beta_i)$$

Erwartete **Reduktion der E.U.** hinsichtlich Q

$$\Delta_Q(\beta) \;=\; \Im(\beta) - \Im(\beta|Q)$$

Für das Entropie-Kriterium ergibt sich folgende **empirische** Entscheidungsunsicherheit:

$$\hat{\Delta}_Q(\beta) \;=\; -\sum_{\kappa=1}^{K} \frac{N_\kappa(\beta)}{N(\beta)} \log_2 \frac{N_\kappa(\beta)}{N(\beta)} \;+\; \sum_i \frac{N(\beta_i)}{N(\beta)} \sum_{\kappa=1}^{K} \frac{N_\kappa(\beta_i)}{N(\beta_i)} \log_2 \frac{N_\kappa(\beta_i)}{N(\beta_i)}$$

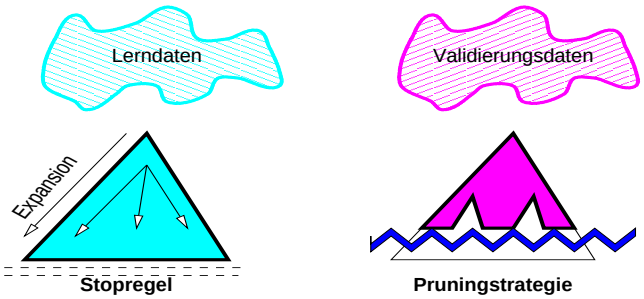
Dabei bezeichnen

$$N(\beta) \;=\; |\bigcup_\lambda \omega_\lambda(\beta)|\;, \quad N_\kappa(\beta) \;=\; |\omega_\kappa(\beta)|$$

die Kardinalitäten der assoziierten Stichproben; die β_i laufen über die beiden durch Q induzierten Nachfolger β°, $\beta^\bullet$ von β.

Überanpassung an die Lernstichprobe

- große Stichprobe $\rightsquigarrow$ großer Entscheidungsbaum
- unzuverlässige Entscheidung in den Blättern
- unsignifikante Fragen in den unteren Zweigen



1 ZÜCHTEN:
Expandiere initialen Baum mittels Lernstichprobe $[\omega_\kappa]$ unter Einhaltung des *Reinheitsgebotes*.

2 ZURÜCKSCHNEIDEN:
Eliminiere sukzessiv (von unten) alle Knoten, die keine *Verminderung der Fehlerrate* auf der Validierungsstichprobe bewirken.

Wahre und empirische Fehlerraten

Lokale Fehlerrate im Knoten β

$$\varepsilon(\beta) \;=\; 1 - \max_\kappa p_\kappa(\beta)$$

Gesamtfehlerrate des Entscheidungsbaumes

$$\varepsilon(\mathcal{B}) \;=\; \sum_{\beta\in\mathcal{B}_\ell} \mathsf{P}(\mathbb{X}\in\boldsymbol{\Omega}(\beta))\cdot\varepsilon(\beta)$$

Kumulative Fehlerrate im Knoten β

$$\varepsilon^\star(\mathcal{B}) \;=\; \frac{1}{\mathsf{P}(\mathbb{X}\in\boldsymbol{\Omega}(\beta))}\cdot\sum_{\beta'\in\mathcal{B}_\ell^\beta} \mathsf{P}(\mathbb{X}\in\boldsymbol{\Omega}(\beta'))\cdot\varepsilon(\beta')$$

(es sei $\mathcal{B}^\beta$ der Teilbaum von $\mathcal{B}$ unterhalb β)

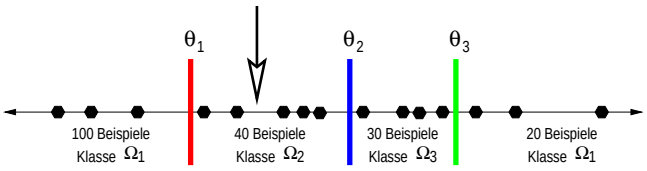
Satz 4.12
Sei $(\mathcal{B}, Q, \delta_\ell)$ ein binärer Entscheidungsbaum über $\boldsymbol{\Omega} = \mathbb{R}^D$. Die **globale Fehlerrate** *des Teilbaums $\mathcal{B}^\beta$ — das ist die minimale kumulative Fehlerrate aller seiner gestutzten Varianten — berechnet sich durch folgende Rekursion:*

$$\varepsilon^\forall(\beta) \;=\; \begin{cases} \varepsilon(\beta) & \beta\in\mathcal{B}_\ell \\ \min\left(\begin{array}{c}\varepsilon(\beta)\\ \sum_{\beta\prec\beta'} p(\beta'|\beta)\cdot\varepsilon^\forall(\beta')\end{array}\right) & \beta\notin\mathcal{B}_\ell \end{cases}$$

Die empirischen Fehlerraten (Lernstichprobe, Validierungsstichprobe, Teststichprobe) werden in analoger Weise definiert und berechnet.

Auswahl einer optimalen Frage

- Nur $N(\beta) = |\omega(\beta)|$ Schwellen sind zu überprüfen
- Nur die klassentrennenden Schwellen besitzen u.U. maximales $\Delta_Q(\beta)$
- Es gibt eine Rekursionsformel für $\Delta_Q(\beta)$



Nichtendliches Optimierungsproblem für β

$$Q(\beta) \;=\; \{x_{d^\star} \le \theta^\star\}$$

$$(d^\star, \theta^\star) \;=\; \operatorname*{argmax}_{1\le d\le D}\ \operatorname*{argmax}_{\theta\in\mathbb{R}} \Delta_{\{x_d\le\theta\}}(\beta)$$

Aufsteigende Sortierung der d-Merkmalwerte $\{x_d \mid \boldsymbol{x}\in\omega(\beta)\}$

$$a_1 < a_2 < a_3 \ldots < a_n < \ldots < a_{N(\beta)}$$

Potentielle Schwellenwerte für $Q_{d,\theta}$ sind ausschließlich

$$\theta_n \;=\; \frac{a_{n+1}-a_n}{2}\,, \qquad n = 1, 2, \ldots, N(\beta)-1$$

$\rightsquigarrow$ „**Mittelpunktschwellen**“

Separierende Schwellenwerte

Definition 4.12
Eine Mittelpunktschwelle θ_n von $\{x_d \mid \boldsymbol{x}\in\omega(\beta)\}$ heißt **innere** *Schwelle, falls alle Muster $\boldsymbol{x}\in\omega(\beta)$ mit $x_d = a_n$ oder $x_d = a_{n+1}$ zu einundderselben Klasse Ω_κ gehören.*

Andernfalls heißt θ_n **separierende** *Schwelle oder* **Klassengrenze**.

Satz 4.13 (Fayyad & Irani, 1992)
Sind $[\theta_n]$ die Mittelpunktschwellen zur assoziierten Stichprobe $[\omega_\kappa(\beta)]$ von β zum Merkmalindex d, und gilt

$$\theta_{n^\star} \;=\; \operatorname*{argmax}_{\theta_n} \Delta_{\{x_d\le\theta_n\}}(\beta)$$

für die entropiebezogene Entscheidungsunsicherheit, so ist $\theta_{n^\star}$ notwendigerweise eine Klassengrenze.

Satz 4.14 *Es seien $\theta_1 < \theta_2$ zwei benachbarte Klassengrenzen für die x_d in $[\omega_\kappa(\beta)]$, zwischen denen genau m Muster der Klasse Ω_κ liegen. Dann gilt die Rekursionsformel*

$$\Delta_{\{x_d\le\theta_2\}}(\beta) \;=\; \Delta_{\{x_d\le\theta_1\}}(\beta) + \frac{h(l,r)-h(l+m,r+m)+h(l_\kappa+m,r_\kappa+m)-h(l_\kappa,r_\kappa)}{N}$$

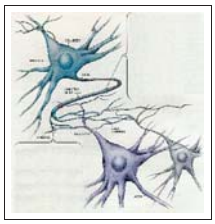
mit den Abkürzungen

$$h(p,q) \;=\; p\log_2 p - q\log_2 q$$

und den Zählwerten

$$l_\kappa \;=\; |\{x_d < \theta_1 \mid \boldsymbol{x}\in\omega_\kappa\}| \qquad l \;=\; \textstyle\sum_\kappa l_\kappa$$
$$r_\kappa \;=\; |\{x_d > \theta_1 \mid \boldsymbol{x}\in\omega_\kappa\}| \qquad r \;=\; \textstyle\sum_\kappa r_\kappa$$

Das menschliche Zentralnervensystem

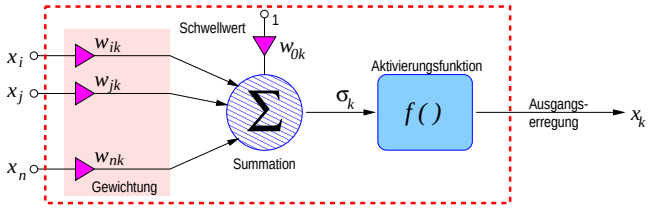


Ca. 10^{12} Nervenzellen
Ca. 10^{14} Verbindungen

Das **Neuron** — elementare Verarbeitungseinheit

- Sein **Axon** liefert eine Grundaktivität in Form elektrischer Impulse
- Weitergabe an andere Neuronen über **synaptische** Verbindungen unterschiedlicher Stärke
- Überschreitet die gesammelte **Eingangserregung** einen bestimmten Betrag, führt dies zu einer Aktivitätsänderung *(erhöhte Impulsfrequenz)*

MODELL-NEURON



Ein **Künstliches Neuronales Netz** (KNN) ist ein Arrangement von *Modellneuronen* und ihren *Verbindungen*.

Sein **dynamisches Verhalten** wird bestimmt durch:

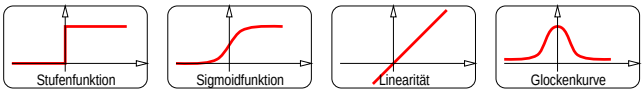
A. GEWICHTMATRIX

$$\boldsymbol{W} \;=\; [w_{jk}]_{1\le j,k\le N}$$

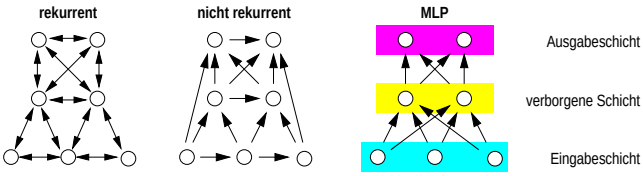
B. DEFINITION DER EINGANGSERREGUNG

$$\sigma_k \;=\; \begin{cases} \sum_j w_{jk}x_j & \text{(Skalarprodukt)} \\ \sum_j (w_{jk}-x_j)^2 & \text{(Abstandsmaß)} \end{cases}$$

C. AKTIVIERUNGSFUNKTION



D. VERBINDUNGSTOPOLOGIE



CAM — Content-Addressable Memory

- CAM realisiert ein *assoziatives* Gedächtnis
  Zugriffsschlüssel $\subseteq$ Datensatz
- Neuronen mit *diskreten* Zuständen
  $$x_j \in \{-1,+1\}$$
- Eingangserregung, Aktivierung $\rightsquigarrow$ Systemdynamik
  $$x_k(t+1) \;=\; \text{sign}\left(\sum_j w_{jk}x_j(t)\right)$$
- Der **Systemzustand** in t ist charakterisiert durch
  $$\boldsymbol{x}(t) \;=\; (x_1(t), x_2(t), \ldots, x_N(t))^\top$$
- EINGABEMUSTER: Startzustand $\boldsymbol{x}(0)$
- AUSGABEMUSTER: **Gleichgewichtszustand** $\boldsymbol{y}$ mit
  $$\boldsymbol{y} \;=\; \boldsymbol{x}(t) \quad \text{für alle } t > t_0$$
  („*equilibrium*“, Attraktionszonen = Klassengebiete)
- Aufgabengebiete
  Bildrestaurierung, Gesichterkennung, Retrieval unvollständiger Muster, Tabellenzugriff
- *Autoassoziatives CAM* — Hopfield-Netze
  Heteroassoziatives CAM — Kanerva-Modell

Hopfield-Netze

Volle Verbindungstopologie; symmetrische Gewichte

Definition 4.13
Das dynamische System mit dem Zustandsübergang

$$x_k(t+1) \;=\; \text{sign}\left(\sum_{j=1}^{N} w_{jk}x_j(t)\right) \;, \quad t\in\mathbb{N},\; k=1,\ldots,N$$

mit der symmetrischen Gewichtmatrix $\boldsymbol{W}$ *heißt* **Hopfield-Netz**. *Es definiert eine (partielle) Abbildung*

$$\boldsymbol{u} : \begin{cases} \{-1,+1\}^N & \to & \{-1,+1\}^N \\ \boldsymbol{z} & \rightharpoonup & \lim\limits_{t\to\infty} \boldsymbol{x}(t) \;\; \textit{mit dem Start } \boldsymbol{x}(0)=\boldsymbol{z} \end{cases}$$

im Raum der N*-dimensionalen Binärmuster.*

Hebb'sche Lernregel:
(Gewichte proportional zum Aktivierungsratenprodukt der beiden vernetzten Neuronen)

$$w_{jk} \;=\; \sum_{\lambda=1}^{K} y_{\lambda,j}\cdot y_{\lambda,k}$$

Bewirkt die Speicherung der K „Prototypen“ $\boldsymbol{y}_1,\ldots,\boldsymbol{y}_K$

Für Gleichgewichtspunkte $\boldsymbol{x} = \text{sign}(\boldsymbol{W}\boldsymbol{x})$ gilt:

$$\sum_j w_{jk}x_j \;=\; \sum_j\sum_\lambda y_{\lambda,j}y_{\lambda,k}x_j$$

Für einen Prototypen $\boldsymbol{x}=\boldsymbol{y}_\nu$ gilt insbesondere

$$\sum_j w_{jk}y_{\nu,j} \;=\; \sum_j\sum_\lambda y_{\lambda,j}y_{\lambda,k}y_{\nu,j} \;=\; N\cdot y_{\nu,k} + \sum_j\sum_{\lambda\neq\nu} y_{\lambda,j}y_{\lambda,k}y_{\nu,j}$$

$\boldsymbol{y}_\nu$ ist Gleichgewichtspunkt, wenn der rechte Term betragsmäßig kleiner als N ist.

KAPAZITÄT: $\quad K \;\le\; 0.138\cdot N$

BEISPIEL: *(John S. Denker, 1986)*

Telefonbucheinträge (25 Zeichen/Zeile, 5 Bit/Z)

Prototypen

John Stewart Denker	8128
Lawrence David Jackel	7773
Richard Edward Howard	5952
Wayne P. Hubbard	7707
Brian W. Straughn	3126
John Henry Scofield	8109

Erfolgreicher Zugriff

t	$E = -\boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{x}$	CAM-Zustand	
0.0	-0.0	john s	
0.2	-0.0784	john sdewirubneoimv	8109
0.4	-0.8426	john sdewirtbnenimv	8129
0.6	-0.8451	john sdewirtbnenimv	8129
0.8	-0.8581	john sdewirt nenkmv	8128
1.0	-0.9099	john sdewart denker	8128
1.2	-0.9824	john stewart denker	8128

Erfolgloser Zugriff

t	$E = -\boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{x}$	CAM-Zustand	
0.0	-0.0	garbage	
0.2	-0.0244	garbagee lafj naabd	5173
0.4	-0.6280	garbaged derjd naabd	7173
0.6	-0.6904	garbaged derjd naabd	7173
0.8	-0.7345	gasbafed derjd naabd	7173
1.0	-0.7595	gasbabed derjd naabd	7173
1.2	-0.7709	fasjebad derjd naabd	7173
1.4	-0.8276	fasjebad derjd naabd	7173
1.6	-0.8282	fasjeb d derjd naabd	7173

Verbindungstopologien

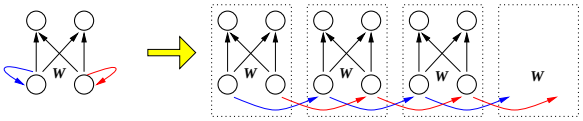
Die Einträge w_{jk} der Gewichtmatrix $\boldsymbol{W}$ definieren eine **neuronale Verknüpfungsrelation**

$$j \prec k \quad \overset{\text{def}}{\Leftrightarrow} \quad w_{jk} \neq 0$$

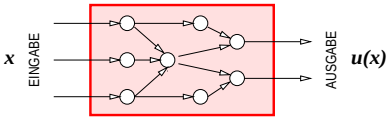
auf der Menge $\mathcal{N} = \{1, \ldots, N\}$ der Modellneuronen eines KNN.

- **Rekurrente Netzwerke**
  Neuronale Zyklen $i_1 \prec i_2 \prec \ldots \prec i_n \prec i_1$ — Expansion ZEIT / RAUM

- **Zeitverzögerungsnetzwerke** (TDNN)
  Nur einfache Zyklen $i \prec i$ — kartesisches Produktnetz $\mathcal{N} \times [1:T]$



- **Nichtrekurrente Netzwerke**
  Keine Zyklen — es erübrigt sich die Zeitvariable t



- **Mehrschichten-Netzwerke** (MLP)
  Es gibt eine Zerlegung in Neuronenschichten

$$\mathcal{N} = \mathcal{N}_0 \uplus \mathcal{N}_1 \uplus \ldots \uplus \mathcal{N}_m$$

Vollständige Vernetzung von Schicht zu Schicht

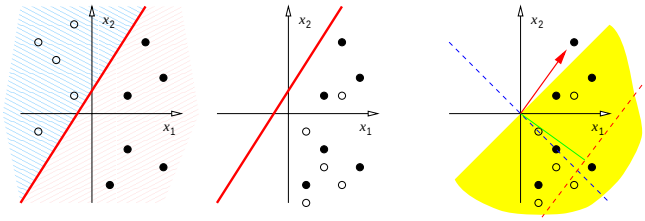
$$j \prec k \quad \Leftrightarrow \quad (\exists \nu)\ j \in \mathcal{N}_{\nu-1} \wedge k \in \mathcal{N}_\nu$$

Klassisches Perzeptron (Rosenblatt 1962)

Definition 4.14 *Ein einschichtiges KNN mit N Eingabeknoten, K Ausgabeknoten, der Gewichtmatrix $\boldsymbol{W} \in \mathbb{R}^{K \times N}$ und dem Systemverhalten*

$$y_k(\boldsymbol{x}) \;=\; \text{sign}\left(\sum_{j=1}^{N} w_{jk} \cdot x_j\right) \quad , \quad k = 1, \ldots, K$$

heißt **Perzeptron**.



2-Klassen-Problem Punktspiegelung Korrekturregel

Perzeptron-Lernregel

$$\boldsymbol{w} \;=\; \begin{cases} \boldsymbol{w} & \boldsymbol{w}^\top \boldsymbol{z} > 0 \\ \boldsymbol{w} - \eta \cdot (\boldsymbol{w}^\top \boldsymbol{z}) \cdot \boldsymbol{z} & \text{sonst} \end{cases}$$

- Korrektur von $\boldsymbol{w}$ proportional zur *negativen Projektion* von $\boldsymbol{w}$ auf $\boldsymbol{z}$. Der Koeffizient η heißt **Lernrate**.

- Prinzip des **Gradientenabstiegs**: $w'_{jk} = w_{jk} - \eta \Delta w_{jk}$ mit

$$\Delta w_{jk} \;=\; \partial \varepsilon(\boldsymbol{z}) \;/\; \partial w_{jk}$$

- Stapelweise Auffrischung der Gewichte

$$\Delta w_{jk} \;=\; \sum_{\boldsymbol{z} \in \omega} \partial \varepsilon(\boldsymbol{z}) \;/\; \partial w_{jk}$$

Das Mehrschichtenperzeptron (MLP)

(multi-layer perceptron)

Definition 4.15
Ein KNN mit Gewichtmatrix $\boldsymbol{W} \in \mathbb{R}^{N \times N}$ und zyklenfreier neuronaler Verknüpfungsrelation '$\prec$' heißt **nichtrekurrentes KNN** *oder* **Vorwärtsnetz**.
Die Menge $\mathcal{N}$ aller Netzknoten zerfällt in die Teilmengen

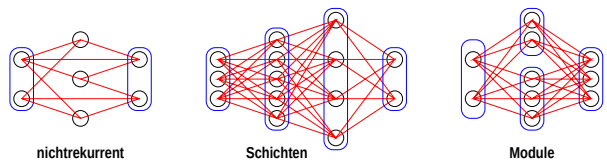
$$\begin{aligned} \mathcal{N}_E &= \{k \in \mathcal{N} \mid \neg \exists j : j \prec k\} \\ \mathcal{N}_A &= \{j \in \mathcal{N} \mid \neg \exists k : j \prec k\} \\ \mathcal{N}_H &= \mathcal{N} \setminus (\mathcal{N}_E \cup \mathcal{N}_A) \end{aligned}$$

der **Eingabe-**, *der* **Ausgabe-** *bzw. der* **verborgenen** *Neuronen.*
Zerfällt $\mathcal{N}$ darüberhinaus in disjunkte Schichten $\mathcal{N}_0, \ldots, \mathcal{N}_L$ mit der Eigenschaft

$$w_{jk} \neq 0 \quad \Leftrightarrow \quad (\exists \ell)\ j \in \mathcal{N}_{\ell-1} \wedge k \in \mathcal{N}_\ell$$

für alle $1 \leq j, k \leq N$, so sprechen wir von einem **Mehrschichtenperzeptron**.

($L+1$ Schichten von *Neuronen* — L Schichten von *Gewichten*)



nichtrekurrent
keine Anregungszyklen

Schichten
volle Verdrahtung zwischen benachbarten Schichten

Module
zyklenfreier Modulgraph
Metakanten voll verdrahtet

Satz 4.15 *Die Ausgangserregungen $x_k(t)$ der Neuronen eines Mehrschichtenperzeptrons sind nicht von der Zeit, sondern nur von den Zuständen $\boldsymbol{x}_E = (x_k | k \in \mathcal{N}_E)^\top$ abhängig und berechnen sich durch die Vorwärtsrekursion*

$$x_k \;=\; f\left(\sum_{j \in \mathcal{N}_{\ell-1}} w_{jk} x_j\right) \qquad (\forall \ell = 1, \ldots, L) \text{ und } (\forall k \in \mathcal{N}_\ell)$$

mit der Aktivierungsfunktion $f(\cdot)$. Das Netzwerk definiert folglich eine Abbildung

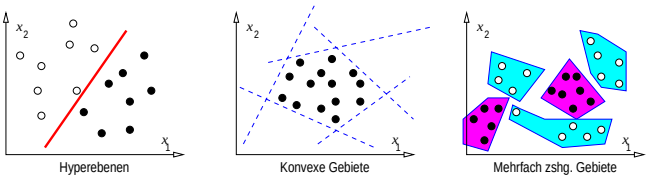
$$\boldsymbol{u} : \begin{cases} \mathbb{R}^{|\mathcal{N}_E|} & \to \quad \mathbb{R}^{|\mathcal{N}_A|} \\ \boldsymbol{x}_E & \mapsto \quad \boldsymbol{u}(\boldsymbol{x}_E) = \boldsymbol{x}_A = (x_k | k \in \mathcal{N}_A)^\top \end{cases}$$

BEMERKUNG:
Für beliebige nichtrekurrente Netze gilt die Rekursionsformel

$$x_k \;=\; f\left(\sum_{j \prec k} w_{jk} x_j\right)$$

wobei die Berechnungsreihenfolge verträglich mit '$\prec$' sein muß.

Ein Klassifikationssystem für K Klassen des $\boldsymbol{\Omega} = \mathbb{R}^D$ kann durch ein Vorwärtsnetz mit D Eingabe- und K Ausgabeneuronen realisiert werden.



Satz 4.16 *Mehrschichtennetzwerke mit nichtlinearer Aktivierungsfunktion (z.B. Signum oder Sigmoid) und mindestens zwei verborgenen Neuronenschichten ($L \geq 3$) können — asymptotisch mit wachsender Neuronenzahl — beliebige Klassengebiete bzw. Funktionen approximieren.*

Fehlerrückführungsverfahren

(„error backpropagation")

Minimiere den **quadratischen Fehler**

$$\varepsilon(\omega | \boldsymbol{W}) \;=\; \sum_{\boldsymbol{x} \in \omega} \|d(\boldsymbol{x}) - \boldsymbol{u}(\boldsymbol{x})\|^2$$

zwischer *idealer* und *neuronaler* Prüfgröße

Stapelweise Auffrischung (*„batch learning"*)

$$\boldsymbol{W}' \;=\; \boldsymbol{W} - \eta \nabla_{\boldsymbol{W}} \varepsilon(\omega | \boldsymbol{W})$$

mit der Gradientenmatrix $\nabla_{\boldsymbol{W}} \varepsilon = [\partial \varepsilon / \partial w_{jk}]$

Satz 4.17 (Werbos 1972)
Gegeben sei ein nichtrekurrentes KNN mit Neuronen $\mathcal{N}$, Gewichtmatrix $\boldsymbol{W}$ und sigmoider Aktivierungsfunktion

$$f(\sigma) \;=\; \frac{1}{1 + e^{\sigma}}$$

Sei ferner $\boldsymbol{x}_E \in \mathbb{R}^{|\mathcal{N}_E|}$ ein Eingabevektor und $\boldsymbol{d} \in \mathbb{R}^{|\mathcal{N}_A|}$ die zu lernende ideale Ausgabe dazu. Dann gilt für die partiellen Ableitungen des quadratischen Fehlers

$$\varepsilon \;=\; \frac{1}{2} \cdot \|\boldsymbol{d} - \boldsymbol{x}_A\|^2$$

— $\boldsymbol{x}$, $\boldsymbol{x}_A$ bezeichnen die Zustände der (Ausgangs)neuronen bei Eingabe von $\boldsymbol{x}$ — die Aussage

$$\partial \varepsilon / \partial w_{jk} \;=\; x_j x_k (1 - x_k) \beta_k$$

mit den rekursiv berechenbaren Größen

$$\beta_j \;=\; \frac{\partial \varepsilon}{\partial x_j} \;=\; \begin{cases} (x_j - d_j) & j \in \mathcal{N}_A \\ \displaystyle\sum_{k | j \prec k} \beta_k x_k (1 - x_k) w_{jk} & j \notin \mathcal{N}_A \end{cases}$$

Fehlerrückführungsalgorithmus

GEGEBEN: ein nichtrekurrentes KNN $(\mathcal{N}, \prec)$ mit
$|\mathcal{N}|_E = D$ und $|\mathcal{N}|_A = K$
und eine klassifizierte Lernstichprobe
$\{\omega_\kappa \mid \kappa = 1, \ldots, K\}$

1 Initialisiere Gewichtmatrix $\boldsymbol{W}$ und setze $\nu = 0$.

2 Initialisiere Akkumulatoren $\Delta^{(\nu)} w_{ij} = 0$.

3 Für alle $\kappa = 1, \ldots, K$, für alle $\boldsymbol{x}_E \in \omega_\kappa$:

- **a** Berechne Eingangserregungen $\boldsymbol{\sigma}(\boldsymbol{x}_E)$.
  Berechne Ausgangserregungen $\boldsymbol{x}(\boldsymbol{x}_E)$.
- **b** Für alle Ausgabeneuronen $j \in \mathcal{N}_A$:

$$\beta_j \;=\; x_j - \begin{cases} 1 & \text{Neuron } j \text{ codiert } \Omega_\kappa \\ 0 & \text{sonst} \end{cases}$$

- **c** Für alle anderen Neuronen $j \in \mathcal{N} \backslash \mathcal{N}_A$:

$$\beta_j \;=\; \sum_{k | j \prec k} \beta_k \cdot f'(\sigma_k) \cdot w_{jk}$$

- **d** Für alle Neuronen $i, j \in \mathcal{N}$ mit $i \prec j$:

$$\Delta^{(\nu)} w_{ij} \Leftarrow \Delta^{(\nu)} w_{ij} + \beta_j \cdot f'(\sigma_j) \cdot x_i$$

4 Berechnen der neuen Gewichte

$$\boldsymbol{W} \Leftarrow \boldsymbol{W} + \eta \cdot \Delta^{(\nu)} \boldsymbol{W}$$

5 Abbruch — oder $\nu \Leftarrow \nu + 1$ und weiter bei $\rightsquigarrow$ $\boxed{2}$

SUPPORTVEKTORMASCHINEN

(Vapnik, Burgess, Schölkopf u.v.a.)

- **Zweiklassenproblem** $\Leftarrow$ Mehrklassenproblem

$$\begin{aligned} \omega \;&=\; \omega_1 \cup \omega_2 \\ &=\; \{(\boldsymbol{x}_t, y_t) \mid t = 1, \ldots, T\} \\ &\subseteq\; \mathbb{R}^D \times \{-1, +1\} \end{aligned}$$

- **Lineare** $\Leftarrow$ nichtlineare Trennfunktionen

$$u(\boldsymbol{x}) \;=\; \boldsymbol{x}^\top \boldsymbol{w} + b \;=\; \sum_{i=1}^{D} w_i x_i + b$$

Hyperebene H, Normalenvektor $\boldsymbol{w}$, Abstand $b / \|\boldsymbol{w}\|$ von $\boldsymbol{0} \in \mathbb{R}^D$

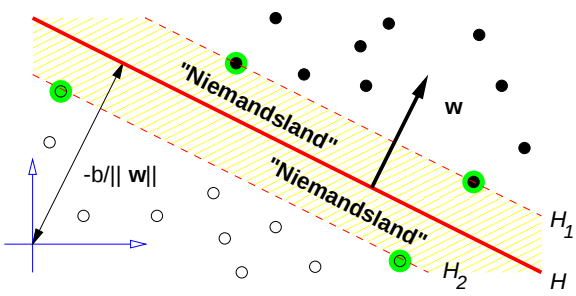
- **Separable** / nichtseparable Lerndaten

$$\exists (\boldsymbol{w}, b) : \quad \begin{cases} \boldsymbol{x}^\top \boldsymbol{w} + b > 0 & \forall t : y_t = +1 \\ \boldsymbol{x}^\top \boldsymbol{w} + b < 0 & \forall t : y_t = -1 \end{cases}$$

Nach Reskalierung folgt:

$$y_t \cdot (\boldsymbol{x}^\top \boldsymbol{w} + b) - 1 \;\geq\; 0 \qquad (\forall t)$$

Separable Lernstichproben



Lemma 4.18 *Die Hyperebene $H : \boldsymbol{x}^\top \boldsymbol{w} + b$, welche die Lernbeispiele*

$$\omega \;=\; \{(\boldsymbol{x}_t, y_t) \mid t = 1, \ldots, T\}$$

mit einem Niemandsland maximaler Breite separiert, gehorcht den Restriktionen

$$y_t \cdot (\boldsymbol{x}_t^\top \boldsymbol{w} + b) - 1 \;\geq\; 0 \qquad (t = 1, \ldots, T)$$

und besitzt dabei die kleinstmögliche Norm $\|\boldsymbol{w}\|$ des Normalenvektors $\boldsymbol{w}$.

Primäres Optimierungsproblem: (Lagrange)

Minimiere das Funktional

$$\ell_p(\omega) \;=\; \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_t \alpha_t y_t \cdot (\boldsymbol{x}_t^\top \boldsymbol{w} + b) + \sum_t \alpha_t$$

bezüglich $\boldsymbol{w}$, b unter den Restriktionen

$$\forall t : \quad \begin{cases} \alpha_t & \geq\; 0 \\ \partial \ell_p / \partial \alpha_t & =\; 0 \end{cases}$$

für die Lagrangemultiplikatoren α_t.

Duales Optimierungsproblem: (Wolfe)

Maximiere das Funktional

$$\ell_d(\omega) \;=\; \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_t \alpha_t y_t \cdot (\boldsymbol{x}_t^\top \boldsymbol{w} + b) + \sum_t \alpha_t$$

bezüglich $\boldsymbol{\alpha}$ unter den Restriktionen

$$\forall t : \quad \begin{cases} \alpha_t & \geq\; 0 \\ \partial \ell_p / \partial w_t & =\; 0 \\ \partial \ell_p / \partial b & =\; 0 \end{cases}$$

und

$$\sum_t \alpha_t y_t \;=\; 0$$

⇨ *Konvexe Quadratische Optimierung*

Lemma 4.19
Für den gesuchten Normalenvektor $\boldsymbol{w}$ der optimalen Hyperebene (Lemma 4.18) für eine separable Stichprobe finden wir nach Maximierung der **Wolfe***'schen dualen Zielfunktion*

$$\ell_d(\omega) \;=\; \sum_t \alpha_t - \frac{1}{2} \cdot \sum_{s,t} \alpha_s \alpha_t \cdot y_s y_t \cdot \boldsymbol{x}_s^\top \boldsymbol{x}_t$$

unter den Restriktionen

$$\forall t : \; \alpha_t \;\geq\; 0 \quad \text{und} \quad \sum_t \alpha_t y_t \;=\; 0$$

die Darstellung

$$\boldsymbol{w} \;=\; \sum_t \alpha_t \cdot y_t \cdot \boldsymbol{x}_t \;.$$

Die infolge $\alpha_t \neq 0$ explizit in dieser Linearkombination auftretenden Stichprobenvektoren $\boldsymbol{x}_t$ heißen **Supportvektoren** *des Klassifikationsproblems.*

Karush-Kuhn-Tucker Bedingungen (KKT):

$$\frac{\partial \ell}{\partial w_i} \;=\; w_i - \sum_t \alpha_t y_t \boldsymbol{x}_{t,i} \;=\; 0 \quad (\forall i)$$

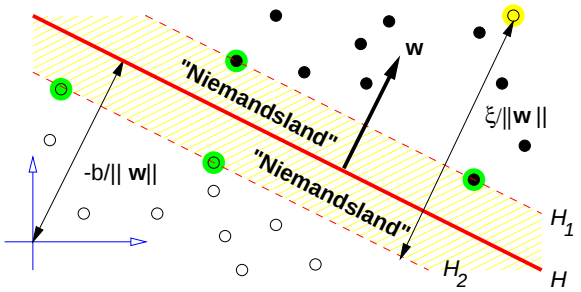
$$\frac{\partial \ell}{\partial b} \;=\; -\sum_t \alpha_t y_t \;=\; 0$$

$$y_t \cdot (\boldsymbol{x}_t^\top \boldsymbol{w} + b) - 1 \;\geq\; 0 \quad (\forall t)$$

$$\alpha_t \cdot (y_t \cdot (\boldsymbol{x}_t^\top \boldsymbol{w} + b) - 1) \;=\; 0 \quad (\forall t)$$

⇨ Es gilt $b \;=\; y_t^{-1} - \boldsymbol{x}_t^\top \boldsymbol{w}$ für alle t mit $\alpha_t \neq 0$.

Nichtseparable Lernstichproben



Schlupfvariable ξ_t, $t = 1, \ldots, T$

Berücksichtigung zusätzlicher Kosten für „*Grenzüberschreitungen*“

$$\boldsymbol{x}_t^\top \boldsymbol{w} + b \;\geq\; +1 - \xi_t \qquad (\forall t : y_t = +1)$$

$$\boldsymbol{x}_t^\top \boldsymbol{w} + b \;\leq\; -1 + \xi_t \qquad (\forall t : y_t = -1)$$

$$\xi_t \;\geq\; 0 \qquad (\forall t)$$

Neue Zielfunktion:

$$\frac{1}{2}\|\boldsymbol{w}\|^2 + C \cdot \sum_t \xi_t \stackrel{!}{=} \text{MIN}$$

Primäres nichtseparables OP: (Lagrange)

Minimiere das Funktional

$$\ell_p(\omega) \;=\; \frac{1}{2}\|\boldsymbol{w}\|^2 + C\cdot\sum_t \xi_t - \sum_t \alpha_t(y_t(\boldsymbol{x}_t^\top \boldsymbol{w}+b)-1+\xi_t) - \sum_t \mu_t\xi_t$$

bezüglich $\boldsymbol{w}$, b unter den Restriktionen

$$\forall t: \quad \begin{cases} \alpha_t & \geq 0 \\ \mu_t & \geq 0 \\ \partial\ell_p/\partial\alpha_t & = 0 \\ \partial\ell_p/\partial\mu_t & = 0 \end{cases}$$

für die Lagrangemultiplikatoren α_t.

Duales nichtseparables OP: (Wolfe)

Maximiere das Funktional

$$\ell_p(\omega) \;=\; \frac{1}{2}\|\boldsymbol{w}\|^2 + C\cdot\sum_t \xi_t - \sum_t \alpha_t(y_t(\boldsymbol{x}_t^\top \boldsymbol{w}+b)-1+\xi_t) - \sum_t \mu_t\xi_t$$

bezüglich $\boldsymbol{\alpha}$ unter den Restriktionen

$$\forall t: \quad \begin{cases} \alpha_t & \geq 0 \\ \mu_t & \geq 0 \\ \partial\ell_p/\partial w_t & = 0 \\ \partial\ell_p/\partial b & = 0 \end{cases}$$

und

$$\sum_t \alpha_t y_t \;=\; 0$$

Lemma 4.20
Für den gesuchten Normalenvektor $\boldsymbol{w}$ der optimalen Hyperebene für eine nicht notwendig separable Stichprobe finden wir nach Maximierung der **Wolfe***'schen dualen Zielfunktion*

$$\ell_d(\omega) \;=\; \sum_t \alpha_t - \frac{1}{2}\cdot\sum_{s,t}\alpha_s\alpha_t\cdot y_s y_t\cdot \boldsymbol{x}_s^\top \boldsymbol{x}_t$$

unter den Restriktionen

$$\forall t: \; 0 \;\leq\; \alpha_t \;\leq\; C \quad \text{und} \quad \sum_t \alpha_t y_t \;=\; 0$$

die Darstellung

$$\boldsymbol{w} \;=\; \sum_t \alpha_t\cdot y_t\cdot \boldsymbol{x}_t \;.$$

Karush-Kuhn-Tucker Bedingungen (KKT):

$$\begin{aligned}
w_i - \textstyle\sum_t \alpha_t y_t x_{t,i} &= \tfrac{\partial\ell}{\partial w_i} &&= 0 & \xi_t &\geq 0 \\
-\textstyle\sum_t \alpha_t y_t &= \tfrac{\partial\ell}{\partial b} &&= 0 & \alpha_t &\geq 0 \\
C - \alpha_t - \mu_t &= \tfrac{\partial\ell}{\partial \xi_i} &&= 0^* & \mu_t &\geq 0 \\
y_t(\boldsymbol{x}_t^\top \boldsymbol{w} + b) - 1 + \xi_t &\geq 0 & & & \mu_t\xi_t &= 0^{**} \\
\alpha_t(y_t(\boldsymbol{x}_t^\top \boldsymbol{w} + b) - 1 + \xi_t) &= 0^{***} & & &&
\end{aligned}$$

⇨ Es gilt $b \;=\; y_t^{-1} - \boldsymbol{x}_t^\top \boldsymbol{w}$ für alle t mit $0 \neq \alpha_t \neq C$.

Nichtlineare Einbettungen

$$\phi: \begin{cases} \mathbb{R}^D & \to \quad \mathbb{H} \\ \boldsymbol{x} & \mapsto \quad \tilde{\boldsymbol{x}} = \phi(\boldsymbol{x}) \end{cases}$$

- **SVM-Prüfgröße**

$$\begin{aligned} u(\boldsymbol{x}) &= \boldsymbol{x}^\top \boldsymbol{w} + b \\ &= \boldsymbol{x}^\top \left(\sum_t \alpha_t y_t \boldsymbol{x}_t\right) + b \\ &= \sum_t \alpha_t y_t \langle \boldsymbol{x}, \boldsymbol{x}_t\rangle + b \end{aligned}$$

- **MAK-Prüfgröße**

$$\begin{aligned} u_\kappa(\boldsymbol{x}) &= \gamma_\kappa + \|\boldsymbol{x} - \boldsymbol{\mu}_\kappa\|^2 \\ &= \gamma'_\kappa + \boldsymbol{x}^\top \boldsymbol{x} - 2\boldsymbol{x}^\top \boldsymbol{\mu}_\kappa \\ &= \gamma'_\kappa + \langle \boldsymbol{x}, \boldsymbol{x}\rangle - 2\cdot\sum_{\boldsymbol{z}\in\omega_\kappa}\frac{1}{N_\kappa}\langle \boldsymbol{x}, \boldsymbol{z}\rangle \end{aligned}$$

- **QMK-Prüfgröße**

$$\begin{aligned} u_\kappa(\boldsymbol{x}) &= (\boldsymbol{a}^{(\kappa)})^\top \boldsymbol{x} \\ &= \left(\sum_t \alpha_t^{(\kappa)} \boldsymbol{x}_t\right)^\top \boldsymbol{x} \\ &= \sum_t \alpha_t^{(\kappa)} \langle \boldsymbol{x}_t, \boldsymbol{x}\rangle \end{aligned}$$

Die Prüfgrößen vieler Klassifikatoren nutzen Eingabe und Lernstichprobe ausschließlich in Form innerer Produkte!

SVM für expandierte Merkmalvektoren

$$\begin{aligned} u(\phi\boldsymbol{x}) &= \langle \phi\boldsymbol{x}, \boldsymbol{w}\rangle + b \\ &= \left\langle \phi\boldsymbol{x}, \sum_t \alpha_t y_t\cdot\phi\boldsymbol{x}_t \right\rangle + b \\ &= \sum_t \alpha_t y_t \underbrace{\langle \phi\boldsymbol{x}, \phi\boldsymbol{x}_t\rangle}_{\mathsf{K}(\boldsymbol{x},\boldsymbol{x}_t)} + b \end{aligned}$$

Beispiel 4.1
Betrachte Datenvektoren im $\mathbb{R}^2$ und den Kernoperator $\mathsf{K}(\boldsymbol{x},\boldsymbol{y}) = (\boldsymbol{x}^\top\boldsymbol{y})^2$. Dann gilt für die Abbildung

$$\phi: \begin{cases} \mathbb{R}^2 & \to \quad \mathbb{H} = \mathbb{R}^3 \\ (x_1, x_2)^\top & \mapsto \quad (x_1^2, \sqrt{2}x_1x_2, x_2^2)^\top \end{cases}$$

die gewünschte Beziehung $\langle \boldsymbol{x},\boldsymbol{y}\rangle = \mathsf{K}(\boldsymbol{x},\boldsymbol{y})$. Gleiches gilt aber auch für

$$\phi'\boldsymbol{x} \stackrel{\text{def}}{=} \begin{pmatrix} (x_1^2 - x_2^2) \\ 2x_1x_2 \\ (x_1^2 + x_2^2) \end{pmatrix} \qquad \text{oder} \qquad \phi''\boldsymbol{x} \stackrel{\text{def}}{=} \begin{pmatrix} x_1^2 \\ x_1x_2 \\ x_1x_2 \\ x_2^2 \end{pmatrix}$$

Satz 4.21 (Mercer-Bedingung, 1953)
Gegeben sei der Kernoperator $\mathsf{K}: \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}$. Dann sind die folgenden Aussagen äquivalent:

- *Für alle Abbildungen $g: \mathbb{R}^D \to \mathbb{R}$ mit $\int g^2(\boldsymbol{x})\,d\boldsymbol{x} < \infty$ gilt*

$$\int \mathsf{K}(\boldsymbol{x},\boldsymbol{y})g(\boldsymbol{x})g(\boldsymbol{y})\,d\boldsymbol{x}\,d\boldsymbol{y} \;\geq\; 0 \;.$$

- *Es gibt einen (un)endlichdimensionalen Vektorraum $\mathbb{H}$ und eine „Injektion" $\phi: \mathbb{R}^D \to \mathbb{H}$ mit*

$$\mathsf{K}(\boldsymbol{x},\boldsymbol{y}) \;=\; \langle \phi\boldsymbol{x}, \phi\boldsymbol{y}\rangle \;=\; \sum_i (\phi\boldsymbol{x})_i(\phi\boldsymbol{y})_i \qquad (\forall \boldsymbol{x},\boldsymbol{y} \in \mathbb{R}^D)$$

Lemma 4.22
Jeder Kernoperator der Gestalt

$$\mathsf{K} : \begin{cases} \mathbb{R}^D \times \mathbb{R}^D & \to & \mathbb{R} \\ (\boldsymbol{x}, \boldsymbol{y}) & \mapsto & (\boldsymbol{x}^\top \boldsymbol{y})^p \end{cases} , \qquad p \in \mathbb{N}$$

erfüllt die Mercer-Bedingung.

Lemma 4.23
Erfüllen die Kernoperatoren $\mathsf{K}_n : \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}$, $n \in \mathbb{N}$ *alle die Mercer-Bedingung, so auch*

$$\mathsf{K}(\boldsymbol{x}, \boldsymbol{y}) \stackrel{\text{def}}{=} \sum_{n=0}^{\infty} c_n \cdot \mathsf{K}_n(\boldsymbol{x}, \boldsymbol{y}) , \qquad c_n > 0 (\forall n)$$

falls $\mathsf{K}(\cdot, \cdot)$ *uniform konvergent ist.*

Beispiel 4.2

- *Jeder* **Polynomialkern**

  $$\mathsf{K}(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x}^\top \boldsymbol{y} + 1)^p , \qquad p \in \mathbb{N}$$

  erfüllt die MB. Die Prüfgröße ist ein Polynom vom Grade p *in den Daten* $(x_1, \ldots, x_D) \in \mathbb{R}^D$.

- *Jeder* **Gaußkern**

  $$\mathsf{K}(\boldsymbol{x}, \boldsymbol{y}) = e^{-\|\boldsymbol{x}-\boldsymbol{y}\|^2/2\sigma^2} , \qquad \sigma > 0$$

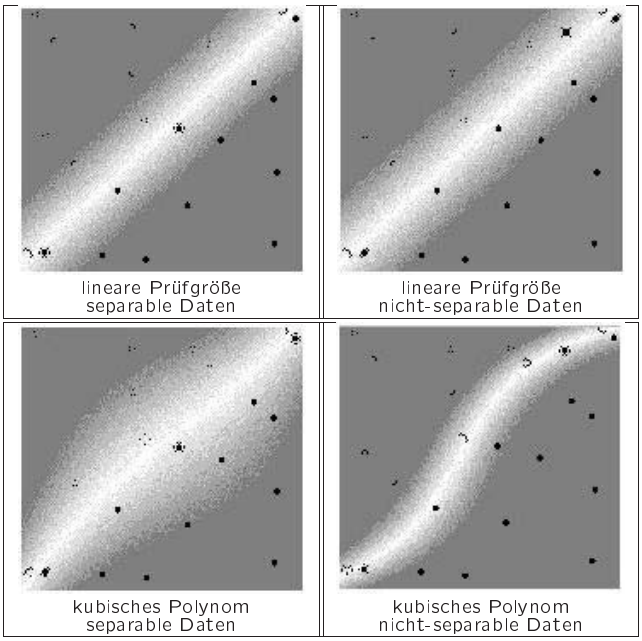
  erfüllt die MB. Die Prüfgröße entspricht einem **Radialbasisfunktion**-*Klassifikator (RBF).*

- *Der* **Tangens-Hyperbolicus-Kern**

  $$\mathsf{K}(\boldsymbol{x}, \boldsymbol{y}) = \tanh(\kappa \boldsymbol{x}^\top \boldsymbol{y} - \delta) , \qquad \kappa, \delta \in \mathbb{R}$$

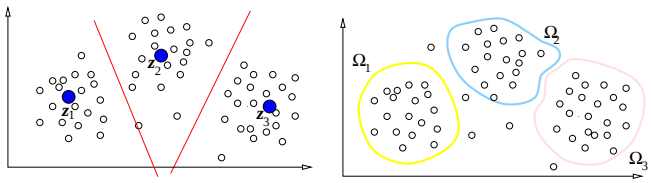
  erfüllt nur für ausgewählte Parameterkombinationen κ, δ *die MB. Die Prüfgröße entspricht dem Ausgang eines zweischichtigen (dreistufigen) MLP mit sigmoider Aktivierungsfunktion.*

Zweiklassenprobleme im $\mathbb{R}^2$



UNÜBERWACHTES LERNEN

Die Lernstichprobe $\omega \subseteq \Omega$ ist *nicht* etikettiert.



Vektorquantisierung Clustering

- **Vektorquantisierung**
  Charakteristische Prototypenvektoren $\{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_K\}$ mit

  $$\varepsilon(\boldsymbol{x}) \stackrel{\text{def}}{=} \min_{\kappa} \|\boldsymbol{x} - \boldsymbol{z}_\kappa\|^2 \quad \rightsquigarrow \quad \text{„kleiner Wert“}$$

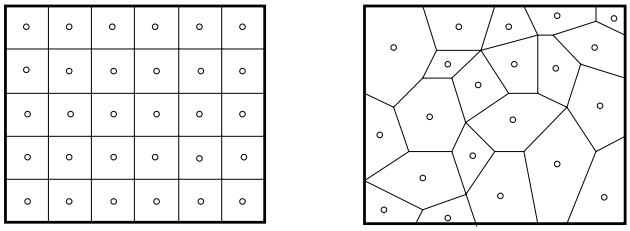
- **Häufungsanalyse** (*„clustering“*)
  Trennscharfe Zerlegung in kompakte Teilmengen

  $$\omega = \omega_1 \uplus \omega_2 \uplus \ldots \uplus \omega_K$$

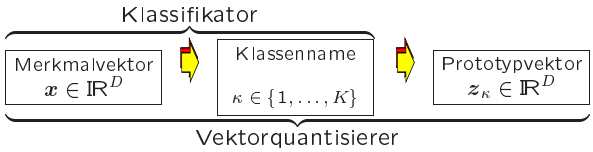
- **Mischungsidentifikation**
  Modelliere die Daten ω mit der Mischverteilung

  $$\mathsf{P}(\boldsymbol{x}) = \sum_{\kappa=1}^{K} c_\kappa \cdot \mathsf{f}(\boldsymbol{x}|\boldsymbol{\theta}_\kappa) , \qquad \textstyle\sum_\kappa c_\kappa = 1$$

Vektorquantisierung



Uniforme Quantisierung der Ebene Nicht-uniforme Quantisierung



Definition 4.16
Ist $\mathcal{Z} = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_K\}$ *eine Menge von* **Prototypenvektoren** *des* $\mathbb{R}^D$, *so heißt die Abbildung*

$$q : \begin{cases} \mathbb{R}^D & \to & \mathcal{Z} \\ \boldsymbol{x} & \mapsto & q(\boldsymbol{x}) \end{cases}$$

Vektorquantisierer *über* $\mathbb{R}^D$ *mit dem* **Codebuch** $\mathcal{Z}$.

Ein Vektorquantisierer mit Codebuchgröße K codiert D-dimensionale Vektoren mit $\lceil \log_2 K \rceil$ bit/Vektor.

Satz 4.24
Der Vektorquantisierer q mit dem Codebuch $\{z_1, \ldots, z_K\}$ definiert eine disjunkte Zerlegung des Raumes $\mathbb{R}^D$ in **Quantisiererzellen**

$$\Omega_\kappa(q) \stackrel{\text{def}}{=} \{\boldsymbol{x} \mid q(\boldsymbol{x}) = \boldsymbol{z}_\kappa\}$$

Auf ähnliche Weise wird auch eine gegebene Stichprobe $\omega \subseteq \mathbb{R}^D$ in disjunkte Teilstichproben $\omega_\kappa(q)$, $\kappa = 1, \ldots, K$ zerlegt.

Um die *Güte* eines Vektorquantisierers zu charakterisieren, setzen wir einen metrischen Abstand $d : \mathbb{R} \times \mathbb{R} \to \mathbb{R}_0^+$ voraus.

Definition 4.17
Es sei $\mathbb{X}$ eine multivariate Zufallsvariable über $\mathbb{R}^D$. Dann heißt

$$\varepsilon_{\mathbb{X}}(\mathcal{Z}, q) \;=\; \mathcal{E}[d(\mathbb{X}, q(\mathbb{X}))] \;=\; \sum_{\kappa=1}^{K} \int_{\boldsymbol{x} \in \Omega_\kappa(q)} d(\boldsymbol{x}, \boldsymbol{z}_\kappa) \mathrm{P}_{\mathbb{X}}(\boldsymbol{x})\, d\boldsymbol{x}$$

die **erwartete Verzerrung** *(oder der* Quantisierungsfehler*) des Vektorquantisierers $(\mathcal{Z}, q)$. Für eine Stichprobe $\omega \subseteq \mathbb{R}^D$ heißt*

$$\varepsilon_\omega(\mathcal{Z}, q) \;=\; \sum_{\kappa=1}^{K} \sum_{\boldsymbol{x} \mid q(\boldsymbol{x}) = \boldsymbol{z}_\kappa} d(\boldsymbol{x}, \boldsymbol{z}_\kappa)$$

die **empirische Verzerrung** *von $(\mathcal{Z}, q)$ bezüglich ω.*

Ein Vektorquantisierer mit minimaler Verzerrung heißt **optimal** *bezüglich $\mathrm{P}_{\mathbb{X}}(\cdot)$ bzw. ω.*

Für die Konstruktion eines optimalen Vektorquantisierers zur Stichprobe ω gibt es keine geschlossene Formel — wir kennen allerdings zwei *notwendige Bedingungen* für die Optimalität.

Satz 4.25 *Ein optimaler Vektorquantisierer wählt stets einen* nächstliegenden *Prototypen aus, d.h. es gilt für alle Vektoren $\boldsymbol{x} \in \mathbb{R}^D$*

$$q(\boldsymbol{x}) \;=\; \operatorname*{argmin}_{\boldsymbol{z}_\lambda} d(\boldsymbol{x}, \boldsymbol{z}_\lambda)$$

bzw. es gilt für alle $\kappa = 1, \ldots, K$ die Aussage

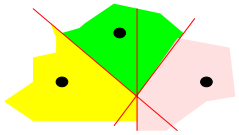
$$\Omega_\kappa(q) \;=\; \left\{ \boldsymbol{x} \mid d(\boldsymbol{x}, \boldsymbol{z}_\kappa) = \min_\lambda d(\boldsymbol{x}, \boldsymbol{z}_\lambda) \right\} .$$

Darüberhinaus ist der Prototypvektor $\boldsymbol{z}_\kappa$ das **Zentroid** *seiner Zelle, d.h. es gilt*

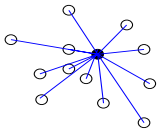
$$\boldsymbol{z}_\kappa \;=\; \operatorname*{argmin}_{\boldsymbol{z}} \mathcal{E}[d(\boldsymbol{z}, \mathbb{X}) \mid \mathbb{X} \in \Omega_\kappa(q)]$$

beziehungsweise

$$\boldsymbol{z}_\kappa \;=\; \operatorname*{argmin}_{\boldsymbol{z}} \sum_{\boldsymbol{x} \in \omega_\kappa(q)} d(\boldsymbol{z}, \boldsymbol{x}) .$$



Voronoi-Partition



Klassenzentroid

BEMERKUNGEN:

- Bei gegebenem Codebuch $\mathcal{Z}$ verursacht der Minimum-Abstand-Quantisierer die geringste Verzerrung.
- Bei gegebener Zellenbildung verursachen die Zentroide als Prototypen die geringste Verzerrung.
- Bei allen elliptisch-symmetrischen Abstandsmaßen
  $$d(\boldsymbol{z}, \boldsymbol{x}) \;=\; (\boldsymbol{z} - \boldsymbol{x})^\top \boldsymbol{S}^{-1} (\boldsymbol{z} - \boldsymbol{x})$$
  entspricht das Zentroid dem Mittel- beziehungsweise Erwartungswert.

LBG-Algorithmus zum Vektorquantisierer-Entwurf

(Iterativer Austauschalgorithmus; Linde/Buzo/Gray 1980)

1 Wähle die Anzahl $K \in \mathbb{N}$ der VQ-Zellen; setze $\nu = 1$.

2 Wähle Startcodebuch / initiale Prototypenvektoren

$$\mathcal{Z}^{(0)} \;=\; \left\{ \boldsymbol{z}^{(0)}{}_1, \ldots, \boldsymbol{z}^{(0)}{}_K \right\}$$

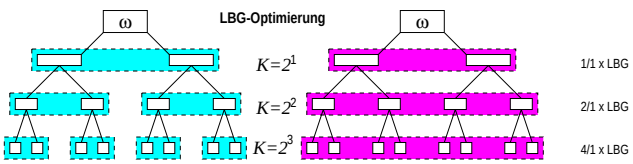
3 Klassifiziere alle Lernmuster und bilde neue Zellen

$$\omega_\kappa^{(\nu)} \;=\; \left\{ \boldsymbol{x} \mid \operatorname*{argmin}_\lambda d(\boldsymbol{x}, \boldsymbol{z}_\lambda^{(\nu-1)}) = \kappa \right\}$$

4 Berechne das neue Codebuch mit den Prototypen

$$\boldsymbol{z}_\kappa^{(\nu)} \;=\; \text{zentroid}(\omega_\kappa^{(\nu)}) \;=\; \frac{1}{|\omega_\kappa^{(\nu)}|} \sum_{\boldsymbol{x} \in \omega_\kappa^{(\nu)}} \boldsymbol{x}$$

5 Abbruch — oder $\nu \Leftarrow \nu + 1$ und weiter bei $\rightsquigarrow$ $\boxed{3}$

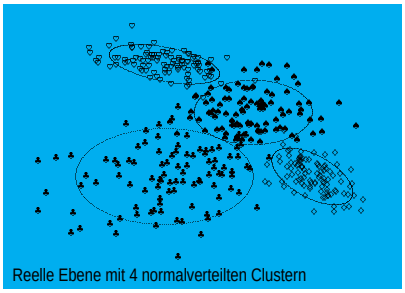


MISCHVERTEILUNGSIDENTIFIKATION

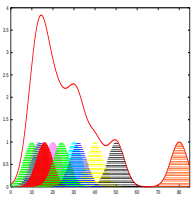
Definition 4.18
Eine (multivariate) Zufallsvariable $\mathbb{X}$ mit der Dichte

$$\mathrm{P}_{\mathbb{X}}(\boldsymbol{x}) \;=\; \sum_{\kappa=1}^{K} c_\kappa \cdot \mathrm{f}(\boldsymbol{x} | \boldsymbol{\theta}_\kappa) \qquad , \; \textstyle\sum_\kappa c_\kappa = 1$$

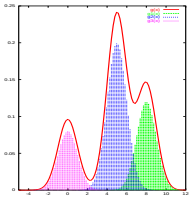
heißt **mischverteilt** *mit der* **Ordnung** *K, den Mischungskoeffizienten $[c_\kappa]$ und Mischungskomponenten aus der parametrischen Verteilungsfamilie $\mathrm{f}(\cdot|\cdot)$.*



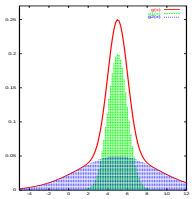
Gemischte Normalverteilungen sind **identifizierbar**, d.h. die Parameterwerte c_κ, $\boldsymbol{\theta}_\kappa$ sind eindeutig bestimmbar, sofern der exakte Funktionsverlauf von $\mathrm{P}_{\mathbb{X}}(\boldsymbol{x})$ bekannt ist!



Parzen-Dichte



Gauß-Mischung



Richter-Mischung

Maximum-Likelihood-Schätzung

Die ML-Zielfunktion einer mischverteilten Stichprobe

$$\begin{aligned}\ell_{c,\theta}(\omega) &= \log \prod_{x\in\omega} \mathrm{P}(x \mid c, \theta) \\ &= \sum_{x\in\omega} \log \left(\sum_{\kappa=1}^{K} c_\kappa \cdot \mathrm{f}(x|\theta_\kappa) \right)\end{aligned}$$

zerfällt nach Nullsetzen der partiellen Ableitungen in ein System *gekoppelter*, *transzendenter* Bestimmungsgleichungen, z.B. für normalverteilte Mischungskomponenten

$$\begin{aligned}\widehat{c}_\kappa &= \frac{1}{|\omega|} \sum_{x\in\omega} \mathrm{P}(\Omega_\kappa \mid x, \widehat{c}, \widehat{\theta}) \\ \widehat{\mu}_\kappa &= \frac{1}{\widehat{c}_\kappa |\omega|} \sum_{x\in\omega} \mathrm{P}(\Omega_\kappa \mid x, \widehat{c}, \widehat{\theta}) \cdot x \\ \widehat{S}_\kappa &= \frac{1}{\widehat{c}_\kappa |\omega|} \sum_{x\in\omega} \mathrm{P}(\Omega_\kappa \mid x, \widehat{c}, \widehat{\theta}) \cdot (x - \widehat{\mu}_\kappa)(x - \widehat{\mu}_\kappa)^\top\end{aligned}$$

HUHN-EI-PROBLEM:
Die Parameter des Schätzproblems berechnen sich aus den *a posteriori* Wahrscheinlichkeiten $\mathrm{P}(\Omega_\kappa|x)$ — und umgekehrt!

ITERATIVE LÖSUNGSANSÄTZE:

- **Entscheidungsüberwachtes Lernen**

$$\mathrm{P}(\Omega_\kappa \mid x, \widehat{c}, \widehat{\theta}) \in \{0, 1\}$$

- **Expectation-Maximization-Algorithmus**
  Zufallsprozesse $(\mathbb{X}, \mathbb{U})$ mit *beobachtbaren* und *verborgenen* Zufallsgrößen

Entscheidungsüberwachtes Lernen

der Parameter einer Mischverteilung
aus einer *unklassifizierten* Lernstichprobe ω

$$\theta^{(0)} \rightsquigarrow [\omega_\kappa^{(1)}] \rightsquigarrow \theta^{(1)} \rightsquigarrow [\omega_\kappa^{(2)}] \rightsquigarrow \theta^{(2)} \rightsquigarrow \ldots \rightsquigarrow \theta^{(\nu)} \rightsquigarrow \ldots$$

1 Wähle Ordnung $K \in \mathbb{N}$ und setze $\nu = 1$.

2 Initialisiere die Parameterfelder

$$\theta_\kappa^{(0)} \quad , \ \kappa = 1, \ldots, K$$

3 Klassifiziere alle Stichprobenvektoren

$$\begin{aligned}\delta^{(\nu)}(x) &= \operatorname*{argmax}_\lambda \mathrm{P}(\Omega_\lambda \mid x, \widehat{c}^{(\nu-1)}, \widehat{\theta}^{(\nu-1)}) \\ \omega_\kappa^{(\nu)} &= \left\{ x \in \omega \mid \delta^{(\nu)}(x) = \kappa \right\}\end{aligned}$$

4 Berechne neue ML-Schätzwerte für die Parameter

$$\begin{aligned}\widehat{c}^{(\nu)} &= |\omega_\kappa^{(\nu)}| \ / \ |\omega| \\ \widehat{\theta}^{(\nu)} &= \operatorname*{argmax}_\theta \ell_\theta(\omega_\kappa^{(\nu)})\end{aligned}$$

5 Abbruch — oder $\nu \Leftarrow \nu + 1$ und weiter bei $\rightsquigarrow$ $\boxed{3}$

Das EM-Prinzip („*expectation-maximization*")

Definition 4.19
Es seien $\mathbb{X}$, $\mathbb{U}$ zwei (Vektoren von) Zufallsvariablen mit der gemeinsamen parametrischen Verteilungsdichte $\mathrm{P}(x, u \mid \theta)$ und der Randverteilungsdichte

$$\mathrm{P}(x|\theta) = \int \mathrm{P}(x, u \mid \theta) \, du \ .$$

Für zwei Parameterfelder θ, θ' heißt die Größe

$$\begin{aligned}\mathrm{Q}(\theta, \theta') &= \mathcal{E}[\log \mathrm{P}(x, u \mid \theta') \mid x, \theta] \\ &= \int \mathrm{P}(u \mid x, \theta) \cdot \log \mathrm{P}(x, u \mid \theta') \, du\end{aligned}$$

die **Kullback-Leibler-Statistik** *von θ und θ'.*

Satz 4.26 (Dempster/Laird/Rubin 1977)
Für die Maximum-Likelihood-Zielfunktion

$$\ell(\theta) = \log \mathrm{P}(x|\theta) = \log \int \mathrm{P}(x, u \mid \theta) \, du$$

der Randverteilungsdichte von $(\mathbb{X}, \mathbb{U})$ gilt die Aussage:

$$\mathrm{Q}(\theta, \theta') \geq \mathrm{Q}(\theta, \theta) \quad \Rightarrow \quad \ell(\theta') \geq \ell(\theta)$$

Insbesondere gilt die Gleichheit der ML-Zielgrößen genau im Fall der Gleichheit der KL-Statistiken.

EM-Algorithmus (allgemeine Form)

1 Setze $\nu = 1$; wähle Startparameter $\theta^{(0)}$.

2 **Erwartungswerte**: $\mathrm{Q}(\theta^{(\nu-1)}, \theta)$

3 **Maximierung**: $\theta^{(\nu)} = \operatorname{argmax}_\theta \mathrm{Q}(\theta^{(\nu-1)}, \theta)$

4 Abbruch — oder $\nu \Leftarrow \nu + 1$ und weiter bei $\rightsquigarrow$ $\boxed{2}$

Satz 4.27 (Mischungsidentifikation)
Der kombinierte Schritt $\boxed{2}+\boxed{3}$ des EM-Algorithmus zur Identifikation der Mischverteilung

$$\mathrm{P}(x) = \sum_{\kappa=1}^{K} c_\kappa \cdot \mathcal{N}(x \mid \mu_\kappa, S_\kappa)$$

auf Grundlage der unklassifizierten Stichprobe ω lautet

$$\begin{aligned}c'_\kappa &= \frac{1}{|\omega|} \sum_{x\in\omega} \gamma_\kappa(x) \\ \mu'_\kappa &= \frac{1}{\sum_{x\in\omega} \gamma_\kappa(x)} \sum_{x\in\omega} \gamma_\kappa(x) \cdot x \\ S'_\kappa &= \frac{1}{\sum_{x\in\omega} \gamma_\kappa(x)} \sum_{x\in\omega} \gamma_\kappa(x) \cdot (x - \mu'_\kappa)(x - \mu'_\kappa)^\top\end{aligned}$$

mit den a posteriori Wahrscheinlichkeiten

$$\gamma_\kappa(x) = \mathrm{P}(\Omega_\kappa \mid x, [c_\lambda, \mu_\lambda, S_\lambda])$$

hinsichtlich der vormaligen Verteilungsparameter.

Mischverteilungsklassifikatoren

1 GMVK — Gaußscher MV-Klassifikator

der Ordnung M

$$\mathrm{P}(\boldsymbol{x}|\Omega_\kappa) \;=\; \sum_{m=1}^{M} c_{\kappa m} \cdot \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_{\kappa m}, \boldsymbol{S}_{\kappa m}) \qquad (\forall \kappa)$$

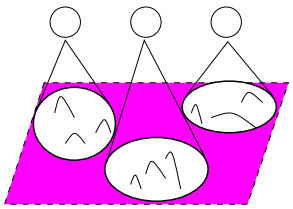
Schätzverfahren: $(\forall \kappa)\ \omega_\kappa \xrightarrow{\text{EM}} [(c_{\kappa m}, \boldsymbol{\mu}_{\kappa m}, \boldsymbol{S}_{\kappa m}) \mid m = 1, \ldots, M]$

2 GKK — Gaußkern-Klassifikator

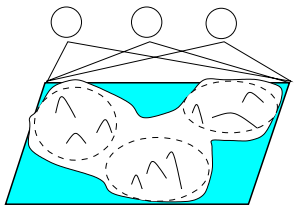
der Ordnung M

$$\mathrm{P}(\boldsymbol{x}|\Omega_\kappa) \;=\; \sum_{m=1}^{M} c_{\kappa m} \cdot \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_{m}, \boldsymbol{S}_{m}) \qquad (\forall \kappa)$$

Einphasenschätzung (EM); Zweiphasenschätzung (ML/QM)



Klassenweise Mischverteilung

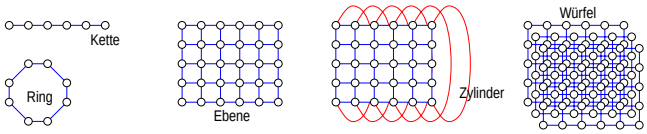


Gemeinsames "Codebuch" von Mischungskomponenten

Merkmalskarten (Teuvi Kohonen, 1988)

SOFM — *„self-organizing feature maps“*

Nichtlineare Abbildung des $\mathbb{R}^D$
auf K regelmäßige Gitterpunkte des $\mathbb{R}^N$



D Eingabeneuronen und
K Ausgabeneuronen mit Gewichtmatrix

$$\boldsymbol{W} \;=\; (\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_K) \;\in\; \mathbb{R}^{D \times K}$$

Neuronale Aktivität

$$u_\kappa(\boldsymbol{x}) \;=\; f(\sigma_\kappa(\boldsymbol{x})) \;=\; \sum_{j=1}^{D} w_{j\kappa} \cdot x_j \;=\; \boldsymbol{w}_\kappa^\top \boldsymbol{x}$$

Unüberwachtes Lernen

$$\max_\kappa \boldsymbol{w}_\kappa^\top \boldsymbol{x} \;\stackrel{!}{=}\; \text{„}\mathit{Maximum}\text{“}$$

$$\min_\kappa \|\boldsymbol{x} - \boldsymbol{w}_\kappa\|^2 \;\stackrel{!}{=}\; \text{„}\mathit{Minimum}\text{“}$$

Gradientenabstieg

$$\boldsymbol{w}_\lambda \Leftarrow \boldsymbol{w}_\lambda + \eta(\boldsymbol{x} - \boldsymbol{w}_\lambda)$$

für alle Nachbarneuronen $\lambda \in \mathcal{U}(\kappa^*)$ des „Siegerneurons“ κ^*

$$\kappa^* \stackrel{\text{def}}{=} \operatorname*{argmin}_\kappa \|\boldsymbol{x} - \boldsymbol{w}_\kappa\|^2$$

STANDARDLERNAUFGABE

- Zufallsvariablenfeld $\mathbb{X}$
- Parameterfeld θ
- Fehlerterm $\varepsilon(\mathbb{X}, \theta)$
- Theoretische Zielgröße

$$\tilde{\ell}(\theta) \;=\; \mathcal{E}[\varepsilon(\mathbb{X}, \theta)] \;\stackrel{!}{=}\; \text{MIN}$$

- Empirische Zielgröße

$$\ell(\theta) \;=\; \sum_{\boldsymbol{x} \in \omega} \varepsilon(\boldsymbol{x}, \theta) \;=\; \sum_{t=1}^{T} \varepsilon(\boldsymbol{x}_t, \theta)$$

LERNSTRATEGIEN

- **Stapelweises Lernen**

$$\hat{\theta} \stackrel{\text{def}}{=} \operatorname*{argmin}_\theta \sum_{\boldsymbol{x} \in \omega} \varepsilon(\boldsymbol{x}, \theta)$$

- **Inkrementelles Lernen**

$$\hat{\theta}_s \stackrel{\text{def}}{=} \operatorname*{argmin}_\theta \sum_{t=1}^{s} \varepsilon(\boldsymbol{x}_t, \theta)$$

- **Adaptives Lernen**

$$\hat{\theta}_s \stackrel{\text{def}}{=} \operatorname*{argmin}_\theta \sum_{t=1}^{s} g(s-t) \cdot \varepsilon(\boldsymbol{x}_t, \theta)$$