

MUSTERERKENNUNG

Vorlesung im Sommersemester 2017

Prof. E.G. Schukat-Talamazzini

Stand: 6. März 2017

Teil X

Unüberwachtes Lernen

Aufgabenstellung

Vektorquantisierung

Mischungsidentifikation

GMM-Klassifikatoren

Selbstorganisierende Karten

Mathematische Hilfsmittel

Unüberwachtes Lernen

Wozu Lernen aus Mustern ohne Klassenetikettierung ?

Klassifikatoren mit preiswerter Lernstichprobe

Unetikettierte Lerndaten sind in den meisten Fällen zu einem Bruchteil der **Kosten** akquirierbar.

Klassen mit multimodalen Musterverteilungen

Die Lerndaten sind nach Musterklassen etikettiert, aber nicht nach den **modusbildenden Faktoren** innerhalb dieser Klassen.

Automatisches Gruppieren von Datenobjekten

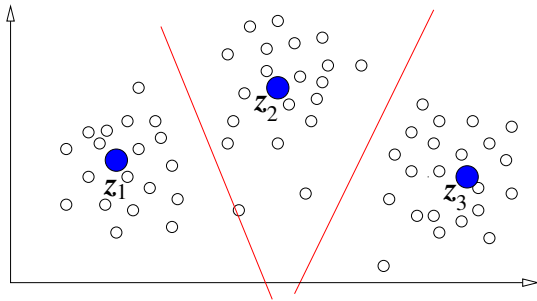
Bildung von **Clustern** (Ballungsgebieten) einander ähnlicher Merkmalvektoren im **Datamining**.

Blockweise Quantisierung von Abtastwerten

Effektivere **Datenkompression** mittels Kodierung von Datenvektoren durch (den Index ihres) **Zellenprototypen**.

Vektorquantisierung

Unüberwachtes Lernen von Zellenprototypen des \mathbb{R}^D



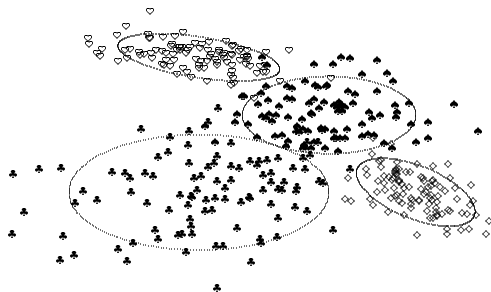
Charakteristische Prototypenvektoren $\{z_1, \dots, z_K\}$ mit

$$\varepsilon(x) \stackrel{\text{def}}{=} \min_{\kappa} \|x - z_{\kappa}\|^2 \rightsquigarrow \text{kleine Verzerrung}$$

🔑 Intensionale Klassenbildung

Mischungsidentifikation

Unüberwachtes Lernen von Mischverteilungsdichtekomponenten



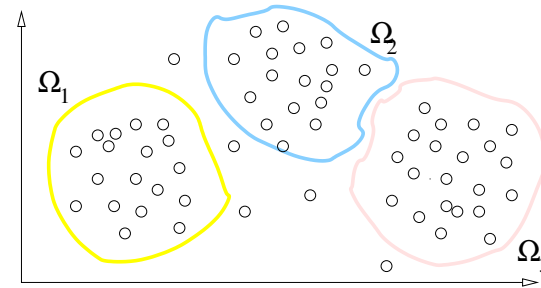
Modelliere die Daten ω mit einer **Mischverteilungsdichte**

$$P(x) = \sum_{\kappa=1}^K c_{\kappa} \cdot f(x|\theta_{\kappa}), \quad \sum_{\kappa} c_{\kappa} = 1$$

🔑 Unscharfe Klassenbildung

Clustering, Häufungsanalyse

Unüberwachtes Lernen von Gruppenzugehörigkeiten



Trennscharfe Zerlegung in kompakte Teilmengen

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

🔑 Extensionale Klassenbildung

Aufgabenstellung

Vektorquantisierung

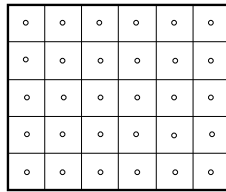
Mischungsidentifikation

GMM-Klassifikatoren

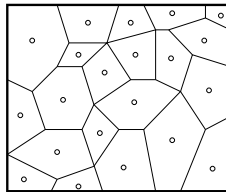
Selbstorganisierende Karten

Mathematische Hilfsmittel

Vektorquantisierung



Uniforme Quantisierung der Ebene



Nicht-uniforme Quantisierung

Definition

Ist $\mathcal{Z} = \{z_1, \dots, z_K\}$ eine Menge von **Prototypenvektoren** des \mathbb{R}^D , so heißt die Abbildung

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

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

Bemerkung

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

Verzerrung eines Quantisierers

Definition

Es sei \mathbb{X} eine multivariate Zufallsvariable über \mathbb{R}^D und $d: \mathbb{R}^D \times \mathbb{R}^D \rightarrow \mathbb{R}$ ein Unähnlichkeitsmaß. Dann heißt

$$\varepsilon_{\mathbb{X}}(\mathcal{Z}, q) = \mathcal{E}[d(\mathbb{X}, q(\mathbb{X}))] = \sum_{\kappa=1}^K \int_{\Omega_{\kappa}(q)} d(\mathbf{x}, z_{\kappa}) \cdot f_{\mathbb{X}}(\mathbf{x}) d\mathbf{x}$$

die **erwartete Verzerrung** (oder *Quantisierungsfehler*) des Vektorquantisierers (\mathcal{Z}, q) .

Für eine Stichprobe $\omega \subseteq \mathbb{R}^D$ heißt

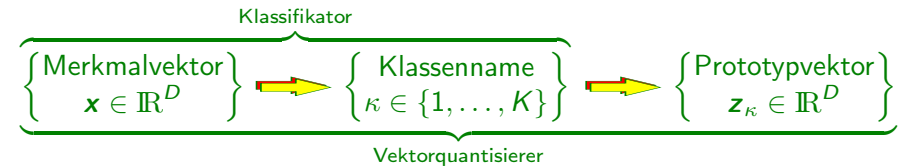
$$\varepsilon_{\omega}(\mathcal{Z}, q) = \sum_{\kappa=1}^K \sum_{\mathbf{x} | q(\mathbf{x}) = z_{\kappa}} d(\mathbf{x}, z_{\kappa})$$

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

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

Vektorquantisierung

Zellen \triangleq Klassen \triangleq Gruppen



Lemma

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

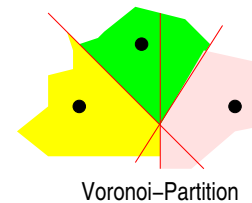
$$\Omega_{\kappa}(q) \stackrel{\text{def}}{=} \{\mathbf{x} \mid q(\mathbf{x}) = z_{\kappa}\}.$$

Bemerkung

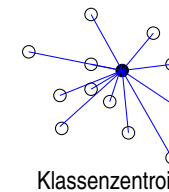
Insbesondere wird der **Lerndatensatz** $\omega \subseteq \mathbb{R}^D$ in disjunkte **Gruppen** $\omega_1(q), \omega_2(q), \dots, \omega_K(q)$ von Datenobjekten zerlegt.

Zwei notwendige Bedingungen

... aber keine geschlossene Lösung für den optimalen Quantisierer (q^*, \mathcal{Z}^*)



Voronoi-Partition



Klassenzentroid

Bei gegebenem Codebuch \mathcal{Z} verursacht der Minimum-Abstand-Quantisierer die geringste Verzerrung.

Bei gegebener Zellenbildung verursachen die Zentroide als Prototypen die geringste Verzerrung.

Satz

Ein optimaler Vektorquantisierer wählt stets den **nächstliegenden** Prototypen aus:

$$q(\mathbf{x}) = \underset{z_{\lambda} \in \mathcal{Z}}{\operatorname{argmin}} d(\mathbf{x}, z_{\lambda})$$

Darüberhinaus ist jeder Prototypvektor z_{κ} ein **Zentroid** seiner Zelle:

$$z_{\kappa} = \underset{\mathbf{z}}{\operatorname{argmin}} \sum_{\mathbf{x} \in \omega_{\kappa}(q)} d(\mathbf{z}, \mathbf{x})$$

Zentroid, Medoid & Mittelwert

Definition

Sei Ω eine Menge mit Unähnlichkeitsmaß $d : \Omega \times \Omega \rightarrow \mathbb{R}$ und $\omega \subset \Omega$ eine (endliche) Teilmenge. Dann heißt

$$\mu^{\text{cnt}}(\omega) = \operatorname{argmin}_{x \in \Omega} \sum_{z \in \omega} d(x, z)$$

Zentroid der Menge ω und es heißt

$$\mu^{\text{med}}(\omega) = \operatorname{argmin}_{x \in \omega} \sum_{z \in \omega} d(x, z)$$

Medoid der Menge ω .

Bemerkungen

1. Das Medoid von ω ist mit Aufwand $O(|\omega|^2)$ zu berechnen.
2. In $\Omega = \mathbb{R}$ mit Betragsmetrik gilt Medoid gleich Median.
3. In $\Omega = \mathbb{R}^D$ mit $d(x, y) = \|x - y\|_2^2$ gilt Zentroid gleich Mittelwert.

K-means Algorithmus

Iterativer Abstieg mit instantaner Auffrischung des Codebuchs

(Algorithmus)

- 1 INITIALISIERUNG
Wähle zufällige Startprototypen $\{z_1, \dots, z_K\}$ aus und setze $t \leftarrow 1$.
- 2 KLASSIFIKATION
Wähle $y = x_{t \bmod T}$ und bestimme die Gewinnerzelle:

$$\kappa = \operatorname{argmin}_{\lambda} \|y - z_{\lambda}\|$$

- 3 REPRÄSENTATION
Frische nun den κ -ten Zellenprototypen auf:

$$z_{\kappa} \leftarrow \alpha_t \cdot y + (1 - \alpha_t) \cdot z_{\kappa}$$

- 4 TERMINIERUNG
Wenn $\varepsilon(\cdot) \leq \theta$ dann ENDE; sonst $t \leftarrow t + 1$ und \rightsquigarrow 2.

(zumfliegA)

Lloyd-Algorithmus

Iterativer Abstieg mit stapelweiser Auffrischung des Codebuchs

(Algorithmus)

- 1 INITIALISIERUNG
Wähle eine zufällige Startpartition $\omega_1 \uplus \omega_2 \uplus \dots \uplus \omega_K = \omega$ aus.
- 2 REPRÄSENTATION
Berechne alle neuen Prototypen:

$$z_{\kappa} = \mu^{\text{cnt}}(\omega_{\kappa}) = \frac{1}{|\omega_{\kappa}|} \cdot \sum_{x \in \omega_{\kappa}} x$$

- 3 KLASSIFIKATION
Berechne alle neuen Gruppen:

$$\omega_{\kappa} = \left\{ x_t \in \omega \mid \operatorname{argmin}_{\lambda} \|x_t - z_{\lambda}\| = \kappa \right\}$$

- 4 TERMINIERUNG
Wenn $\varepsilon_{\omega}(\mathcal{Z}, q) \leq \theta$ dann ENDE; sonst \rightsquigarrow 2.

(zumfliegA)

Qualität des berechneten Codebuchs

Startkonfiguration

- Die Lloyd-Iteration kann mit initialen Prototypen **oder** mit initialen Zellen gestartet werden.
- Initiale Prototypen** werden zufällig aus ω gezogen. $z_{\kappa} \in \omega$
Initiale Zellen werden musterweise ausgewürfelt. $\kappa_t \in \{1..K\}$
Die Codebuchgröße K **ist vorzugeben!**
- Das **Ergebniscodebuch** hängt systematisch von der Wahl der **Startparameter** ab. $O(I \cdot T \cdot K)$

Konvergenzverhalten

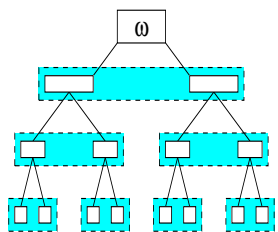
- Instantan aufgefrischte Codebücher **konvergieren schneller** als stapelweise aufgefrischte Codebücher.
- Instantanes Lernen provoziert oft **Parameterszillation**.
- Instantane Iteration bedient sich als Abbruchkriterium einer **näherungsweisen Codebuchverzerrung**.
- Im Iterationsverlauf kommt es u.U. zu **irreversiblen Zellentleerungen**.

LBG-Rekursion

Linde, Buzo, Gray (1980) — Teile-und-Herrsche-Verfahren

Codebuchgröße

$K = 2^B$, $B \in \mathbb{N}$ Zellen



Rechenaufwand

$O(2TI)$ 2-means
 $O(2TI)$ Ebene b
 $O(B \cdot 2TI)$ 2^B -LBG b

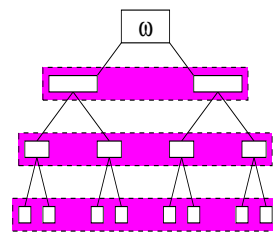
- 1 TERMINIERUNG
Falls $B = 0$ liefere $\mu^{\text{ent}}(\omega)$ zurück.
- 2 REDUKTION I
Berechne Codebuch $\{z_1, z_2\}$ zu ω mittels 2-means oder 2-Lloyd.
- 3 REDUKTION II
Gruppier ω in $\omega_1 \uplus \omega_2$
- 4 REKURSION
Rufe $\text{LBG}(\omega_1, B - 1)$ und $\text{LBG}(\omega_2, B - 1)$
- 5 REKOMBINATION
Vereinige beide Codebücher.

LBG-Iteration

Schrittweise Verfeinerung des Codebuchs

Codebuchgröße

$K = 2^B$, $B \in \mathbb{N}$ Zellen



Rechenaufwand

$O(2TI)$ 2-means
 $O(2^b TI^*)$ Ebene b
 $O(2^B \cdot TI^*)$ 2^B -LBG b

- 1 INITIALISIERUNG
Setze $b = 0$.
- 2 REPRODUKTION
Berechne die 2^b Codebücher $\{z_1^b, z_2^b\}$ zu ω zu den Zellen ω_λ (2-Lloyd).
- 3 REKOMBINATION
Vereinige alle Codebücher zu \mathcal{Z} ; setze $b \leftarrow b + 1$.
- 4 VERZERRUNGSABBAU
Iteriere Codebuch \mathcal{Z} via 2^b -Lloyd.
- 5 ZELLENBILDUNG
Gruppier ω nach Codebuch \mathcal{Z} .
- 6 TERMINIERUNG
Falls $b = B$, dann ENDE; sonst \rightsquigarrow 2.

Aufgabenstellung

Vektorquantisierung

Mischungsidentifikation

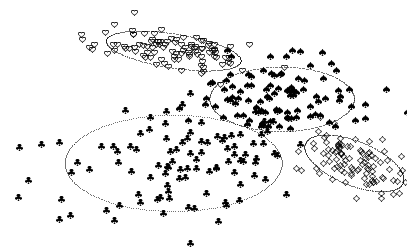
GMM-Klassifikatoren

Selbstorganisierende Karten

Mathematische Hilfsmittel

Mischverteilungsdichten

Dichtegebirge $\hat{=}$ Konvexkombination mehrerer (unimodaler) Dichten



Beispiel

$K = 4$ MV-Komponenten
 $D = 2$ Merkmale
 Normalverteilungsannahme

$$f(\mathbf{x} | \theta_\kappa) = \mathcal{N}(\mathbf{x} | \mu_\kappa, \mathbf{S}_\kappa)$$

für alle $\kappa \in \{\clubsuit, \spadesuit, \diamondsuit, \heartsuit\}$

Definition

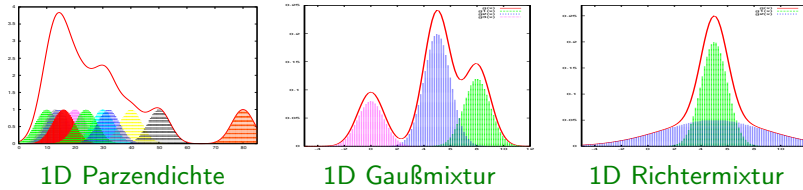
Eine (multivariate) Zufallsvariable \mathbb{X} mit der Dichte

$$f_{\mathbb{X}}(\mathbf{x}) = \sum_{\kappa=1}^K \pi_\kappa \cdot f(\mathbf{x} | \theta_\kappa), \quad \sum_{\kappa} \pi_\kappa = 1$$

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

Mischverteilungsdichten

Identifizierbarkeit — der Schluß von der Summe auf die Summanden



Satz (Yakowitz, 1970^[7])

Gemischte Normalverteilungen sind **identifizierbar**, d.h., die Parameterwerte $\pi_\kappa, \theta_\kappa$ sind eindeutig bestimmbar, sofern der exakte Funktionsverlauf von $f_{\mathbb{X}}(\mathbf{x})$ bekannt ist.

Bemerkungen

1. Beweisidee: Die Familie der NV-Dichten bildet eine Orthogonalbasis.
2. Der Funktionsverlauf von $f_{\mathbb{X}}(\cdot)$ ist selbstverständlich **nicht** bekannt.
3. Alle elliptisch-symmetrischen Dichten $f(\mathbf{x}) = C \cdot \varphi(\|\mathbf{x} - \boldsymbol{\mu}\|_S)$ lassen sich durch Richtermixturen approximieren.^[7]

Entscheidungsüberwachtes Lernen

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

(Algorithmus)

1 INITIALISIERUNG

Wähle Ordnung $K \in \mathbb{N}$, setze $\nu = 1$ und wähle Startparameter

$$\boldsymbol{\theta}_\kappa^{(0)}, \quad \kappa = 1, \dots, K$$

2 NEUKLASSIFIKATION

Klassifiziere auf Grundlage der aktuellen Verteilungsparameter

$$\delta^{(\nu)}(\mathbf{x}) = \underset{\lambda}{\operatorname{argmax}} P(\Omega_\lambda | \mathbf{x}, \boldsymbol{\pi}^{(\nu-1)}, \boldsymbol{\theta}^{(\nu-1)})$$

$$\omega_\kappa^{(\nu)} = \{ \mathbf{x} \in \omega \mid \delta^{(\nu)}(\mathbf{x}) = \kappa \}$$

3 NEUSCHÄTZUNG

Bestimme ML-Schätzwerte auf Grundlage der aktuellen Gruppierung

$$\pi_\kappa^{(\nu)} = |\omega_\kappa^{(\nu)}| / |\omega|$$

$$\boldsymbol{\theta}_\kappa^{(\nu)} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \ell_{\boldsymbol{\theta}}(\omega_\kappa^{(\nu)})$$

4 TERMINIERUNG

Abbruch — oder $\nu \leftarrow \nu + 1$ und weiter bei 2

(zumf3hog1A)

Empirische Mischverteilungsidentifikation

Unetikettierte Lerndaten $\{\mathbf{x}_1, \dots, \mathbf{x}_T\}$ statt Dichtefunktionsverlauf $f_{\mathbb{X}}(\cdot)$

Maximum-Likelihood-Zielfunktion mischverteilter Daten

$$\ell_{\boldsymbol{\pi}, \boldsymbol{\theta}}(\omega) = \log \prod_{\mathbf{x} \in \omega} P(\mathbf{x} | \boldsymbol{\pi}, \boldsymbol{\theta}) = \sum_{\mathbf{x} \in \omega} \log \left(\sum_{\kappa} \pi_\kappa \cdot f(\mathbf{x} | \boldsymbol{\theta}_\kappa) \right)$$

Nullsetzen der partiellen Ableitungen:

$$\hat{\pi}_\kappa = \frac{1}{|\omega|} \cdot \sum_{\mathbf{x} \in \omega} P(\Omega_\kappa | \mathbf{x}, \hat{\boldsymbol{\pi}}, \hat{\boldsymbol{\theta}}) \cdot 1$$

$$\hat{\boldsymbol{\mu}}_\kappa = \frac{1}{\hat{\pi}_\kappa \cdot |\omega|} \cdot \sum_{\mathbf{x} \in \omega} P(\Omega_\kappa | \mathbf{x}, \hat{\boldsymbol{\pi}}, \hat{\boldsymbol{\theta}}) \cdot \mathbf{x}$$

$$\hat{\mathbf{S}}_\kappa = \frac{1}{\hat{\pi}_\kappa \cdot |\omega|} \cdot \sum_{\mathbf{x} \in \omega} P(\Omega_\kappa | \mathbf{x}, \hat{\boldsymbol{\pi}}, \hat{\boldsymbol{\theta}}) \cdot (\mathbf{x} - \hat{\boldsymbol{\mu}}_\kappa)(\mathbf{x} - \hat{\boldsymbol{\mu}}_\kappa)^\top$$

System $\left\{ \begin{array}{l} \text{gekoppelter} \\ \text{transzender} \end{array} \right\}$ Bestimmungsgleichungen \Rightarrow „Huhn-Ei-Problem“

Entscheidungsüberwachtes Lernen

Unüberwachtes Lernen \leftrightarrow Überwachtes Lernen & Iteration

Lemma (EM*-Algorithmus)

Der entscheidungsüberwachte Lernalgorithmus bewirkt in jedem Iterationsschritt eine monotone Verbesserung der **überwachten Likelihoodfunktion**

$$\ell_{\boldsymbol{\pi}, \boldsymbol{\theta}}^*(\omega) = \sum_{\mathbf{x} \in \omega_\kappa} \max_{\kappa=1..K} \log (\pi_\kappa \cdot f(\mathbf{x} | \boldsymbol{\theta}_\kappa)) .$$

Bemerkungen

1. EM* findet i.a. nur ein **lokales** Optimum.
2. EM* konvergiert ohne Oszillationen (s.o.) in $\ell_{\boldsymbol{\pi}, \boldsymbol{\theta}}^*(\omega)$
3. Wichtiger Spezialfall: $f(\cdot | \boldsymbol{\theta}) = \mathcal{N}(\cdot | \boldsymbol{\mu}, \mathbf{S})$
4. Gaußsche Mischungsidentifikation $\hat{=}$ 'ill-posed problem'

$$\omega_1 = \{\mathbf{x}^*\} \rightsquigarrow \hat{\boldsymbol{\mu}}_1 = \mathbf{x}^*, \hat{\mathbf{S}}_1 = \mathbf{0} \rightsquigarrow f(\mathbf{x}^* | \hat{\boldsymbol{\theta}}_1) = \infty \rightsquigarrow \text{„Gotcha!“}$$

ABHILFE: keine Varianzen (VQ) · fixierte/verklebte Varianzen · Regularisierung (MAP) · EM-Prinzip

Identifikation nach dem EM-Prinzip

Unabhängiges, identisches & zweistufiges Auswürfeln

$$P(\omega) = \prod_{t=1}^T f_{\mathbb{X}}(\mathbf{x}_t) = \prod_{t=1}^T \sum_{\kappa=1}^K \pi_{\kappa} \cdot f(\mathbf{x}_t | \theta_{\kappa})$$

- **Beobachtbarer Anteil der Daten** ('observable')

$$\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots, \mathbf{x}_T)^{\top} \in \mathbb{R}^{T \times D}$$

- **Verborgener Anteil der Daten** ('latent')

$$\kappa = (\kappa_1, \kappa_2, \kappa_3, \dots, \kappa_T)^{\top} \in \{1, \dots, K\}^T$$

Maximum-Likelihood-Schätzung


$$\hat{\theta} = \operatorname{argmax}_{\theta} P(\mathbf{X} | \theta) = \operatorname{argmax}_{\theta} \sum_{\kappa_1=1}^K \sum_{\kappa_2=1}^K \sum_{\kappa_3=1}^K \dots \sum_{\kappa_T=1}^K P(\kappa, \mathbf{X} | \theta)$$

Beweis.

Wir haben zu zeigen, daß im Schritt #3 des Algorithmus die Kullback-Leibler-Statistik maximiert wird.

$$\begin{aligned} Q(\theta, \theta') &= \mathcal{E}[\log P([\mathbf{x}_n, \kappa_n] | \theta') | [\mathbf{x}_n], \theta] \\ &= \sum_{\kappa_1=1}^K \dots \sum_{\kappa_N=1}^K (P([\kappa_n] | [\mathbf{x}_n], \theta) \cdot \log P([\mathbf{x}_n, \kappa_n] | \theta')) \\ &= \sum_{\kappa_1=1}^K \dots \sum_{\kappa_N=1}^K \left\{ \prod_{n=1}^N P(\kappa_n | \mathbf{x}_n, \theta) \cdot \sum_{n=1}^N \log (\pi'_{\kappa_n} \cdot P(\mathbf{x}_n | \theta'_{\kappa_n})) \right\} \\ &= \sum_{\kappa=1}^K \sum_{n=1}^N P(\kappa | \mathbf{x}_n, \theta) \cdot (\log \pi'_{\kappa} + \log P(\mathbf{x}_n | \theta'_{\kappa})) \\ &= \sum_{\kappa=1}^K \sum_{n=1}^N \underbrace{\left(\frac{\pi_{\kappa} \cdot P(\mathbf{x}_n | \theta_{\kappa})}{\sum_{\lambda=1}^K \pi_{\lambda} \cdot P(\mathbf{x}_n | \theta_{\lambda})} \right)}_{\gamma_{n\kappa}} (\log \pi'_{\kappa} + \log P(\mathbf{x}_n | \theta'_{\kappa})) \\ &= \sum_{\kappa=1}^K \left(\sum_{n=1}^N \gamma_{n\kappa} \right) \log \pi'_{\kappa} + \sum_{\kappa=1}^K \left(\sum_{n=1}^N \gamma_{n\kappa} \cdot \log \mathcal{N}(\mathbf{x}_n | \mu'_{\kappa}, \mathbf{S}'_{\kappa}) \right) \end{aligned}$$

Die letzte Zeile zerfällt in eine Optimierungsgleichung für die Mischungsgewichte π'_1, \dots, π'_K und in je eine Optimierungsgleichung für die Parameter $\mu'_{\kappa}, \mathbf{S}'_{\kappa}$ der κ -ten Normalverteilungsdichte.

Die Schätzung verläuft praktisch wie im Abschnitt über den NVK beschrieben, nur daß dort die a posteriori Wahrscheinlichkeiten $\gamma_{n\kappa}$ — dafür, daß der Stichprobenvektor \mathbf{x}_n zu Ω_{κ} gehört — „harte“ Zuordnungen trafen, d.h. es galt $\gamma_{n\kappa} \in \{0, 1\}$. 

EM-Algorithmus

Identifikation gaußscher Mischungsverteilungen

(Algorithmus)

1 INITIALISIERUNG

Wähle zufällige Startparameter

$$(\pi_{\kappa}, \mu_{\kappa}, \mathbf{S}_{\kappa}), \quad \kappa \in \{1, \dots, K\}$$

2 A POSTERIORI ERWARTUNGSWERTE

Berechne für alle $\kappa = 1, \dots, K$ und $t = 1, \dots, T$ die Werte

$$\gamma_{\kappa,t} \propto \pi_{\kappa} \cdot \mathcal{N}(\mathbf{x}_t | \mu_{\kappa}, \mathbf{S}_{\kappa}), \quad \sum_{\lambda=1}^K \gamma_{\lambda,t} = 1$$

3 MAXIMIERUNG DER PARAMETER

$$\pi_{\kappa} \leftarrow \frac{\sum_t \gamma_{\kappa,t}}{\sum_{\lambda} \sum_t \gamma_{\lambda,t}}, \quad \mu_{\kappa} \leftarrow \frac{\sum_t \gamma_{\kappa,t} \mathbf{x}_t}{\sum_t \gamma_{\kappa,t}}, \quad \mathbf{S}_{\kappa} \leftarrow \frac{\sum_t \gamma_{\kappa,t} \mathbf{x}_t \mathbf{x}_t^{\top}}{\sum_t \gamma_{\kappa,t}} - \mu_{\kappa} \mu_{\kappa}^{\top}$$

4 TERMINIERUNG

Wenn $\ell(\dots)$ stagniert dann ENDE, sonst \rightsquigarrow 2.

(zumfittingA)

Konvergenzeigenschaften

EM-Identifikation gaußscher Mischungsverteilungen

Ungelöste Probleme

Unbeschränkte Zielgröße

Startwertabhängigkeit

Unproduktive Parameterzyklen

pathologische Lösungen

lokale Optima

Kraterphänomen

Gelöste Probleme

Rangdefizite

Verfälschte Zielgröße

jedes \mathbf{x}_t aktualisiert jedes θ_{λ}

$\ell(\theta) \rightarrow \max$

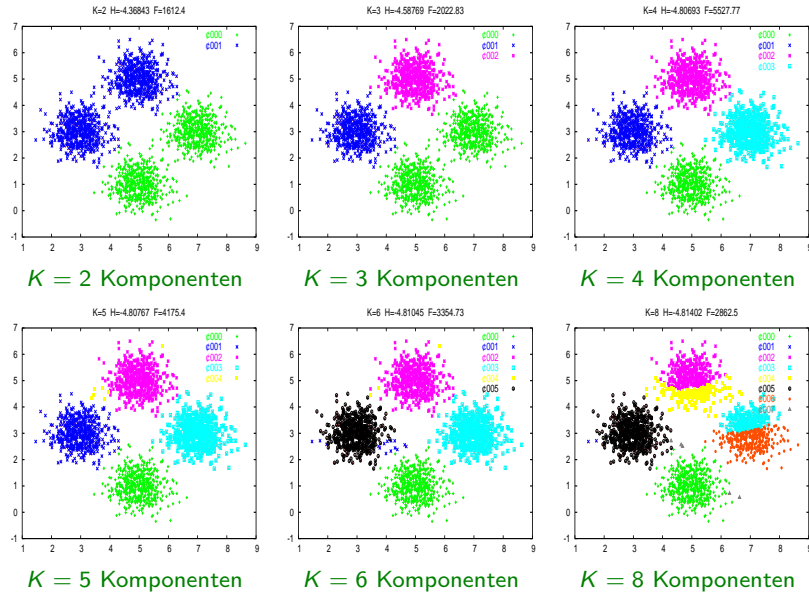
Hintergrundkomponente

Mitführen einer Rückweisungskomponente Ω_0 zur Ausreißerbehandlung:

$$f_0(\cdot) = \mathcal{N}(\cdot | \mu(\omega), \mathbf{S}_0) \quad \text{mit} \quad \mathbf{S}_0 = \begin{cases} \mathbf{S}(\omega) \\ C \cdot \mathbf{E} \end{cases}$$

Beispiel

Clustering einer 4-Mischung isotrop-sphärischer Datenpunkte



Aufgabenstellung

Vektorquantisierung

Mischungsidentifikation

GMM-Klassifikatoren

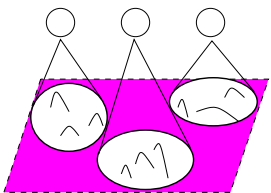
Selbstorganisierende Karten

Mathematische Hilfsmittel

GMK — Gaußscher Mischverteilungsklassifikator

Je ein Verteilungsdichten-Pool der Ordnung M pro Klasse

$$P(\mathbf{x}|\Omega_\kappa) = \sum_{m=1}^M \pi_{\kappa m} \cdot \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_{\kappa m}, \mathbf{S}_{\kappa m})$$



Klassen
MV-Komponenten
Dichteparameter
Dichtekoeffizienten

K
 $K \cdot M$
 $O(K \cdot M \cdot D^2)$
 $O(K \cdot M)$

Klassenweise unabhängiges Parameterschätzverfahren

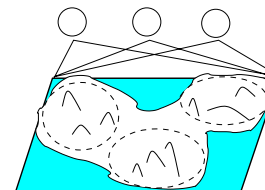
Identifiziere für alle $\kappa = 1..K$ die M -Mischung von ω_κ :

$$\omega_\kappa \xrightarrow{\text{EM}} \{(\pi_{\kappa m}, \boldsymbol{\mu}_{\kappa m}, \mathbf{S}_{\kappa m}) \mid m = 1, \dots, M\}$$

GKK — Gaußkernklassifikator

Ein gemeinsamer Verteilungsdichten-Pool der Ordnung M für alle Klassen

$$P(\mathbf{x}|\Omega_\kappa) = \sum_{m=1}^M \pi_{\kappa m} \cdot \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_m, \mathbf{S}_m)$$



Klassen
MV-Komponenten
Dichteparameter
Dichtekoeffizienten

K
 M
 $O(M \cdot D^2)$
 $O(K \cdot M)$

Einstufiges integriertes Parameterschätzverfahren

Betrachte den dreistufigen Datenerzeugungsprozeß

$$\{p_\kappa\} \rightsquigarrow \mathbb{K} \rightarrow \{\pi_{\kappa m}\} \rightsquigarrow \mathbb{M} \rightarrow \{\boldsymbol{\mu}_m, \mathbf{S}_m\} \rightsquigarrow \mathbb{X} = (\mathbb{X}_1, \dots, \mathbb{X}_D)^\top$$

und wende das EM-Prinzip darauf an.

Gaußkernklassifikator

Zweistufige Parameterschätzung

$$\ell(\omega) = \log \prod_{\kappa=1}^K \prod_{\mathbf{x} \in \omega_{\kappa}} P(\mathbf{x} | \Omega_{\kappa}) = \underbrace{\sum_{\kappa=1}^K \sum_{\mathbf{x} \in \omega_{\kappa}} \log \sum_{m=1}^M \pi_{\kappa m} \cdot \underbrace{\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_m, \mathbf{S}_m)}_{\gamma_m(\mathbf{x})}}_{\ell_{\kappa}(\omega_{\kappa})}$$

(Algorithmus)

1 SCHÄTZUNG DES GLOBALEN DICHTE-POOLS

Berechne das „Codebuch“ mittels EM-Algorithmus:

$$\{(\boldsymbol{\mu}_m, \mathbf{S}_m) \mid m = 1, \dots, M\}$$

2 SCHÄTZUNG DER MISCHUNGSKOEFFIZIENTEN

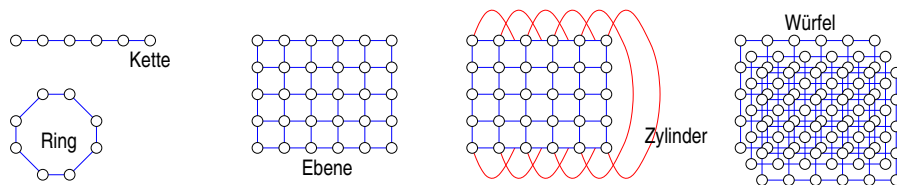
Klassenweise wird ein EM-Algorithmus mit **fixierten** Poolparametern durchgeführt:

$$\pi'_{\kappa m} = \frac{1}{|\omega_{\kappa}|} \sum_{\mathbf{x} \in \omega_{\kappa}} \xi_{\kappa m}(\mathbf{x}) \quad \text{mit} \quad \xi_{\kappa m}(\mathbf{x}) \stackrel{\text{def}}{=} P(\Omega^m | \mathbf{x}) = \frac{\pi_{\kappa m} \gamma_m(\mathbf{x})}{\sum_m \pi_{\kappa m} \gamma_m(\mathbf{x})}$$

(zumfittingA)

Kohonen Mermalkarten

SOFM — 'self-organizing feature map'



$$q^{\text{SOFM}} : \begin{cases} \mathbb{R}^D \rightarrow \{\mathbf{o}_1, \dots, \mathbf{o}_L\} \subset \mathbb{R}^M \\ \mapsto \{\mathbf{w}_1, \dots, \mathbf{w}_L\} \subset \mathbb{R}^D \end{cases}, \quad M = 1, 2, 3$$

Definition

Ein Feld von L Knoten heißt **Selbstorganisierende Karte**, falls jeder Knoten ℓ durch einen **Referenzvektor** $\mathbf{w}_{\ell} \in \mathbb{R}^D$ sowie durch einen **Ortsvektor** $\mathbf{o}_{\ell} \in \mathbb{R}^M$ repräsentiert wird und die Ortsvektoren eine regelmäßige Punktmenge im \mathbb{R}^M bilden.

Aufgabenstellung

Vektorquantisierung

Mischungsidentifikation

GMM-Klassifikatoren

Selbstorganisierende Karten

Mathematische Hilfsmittel

Kompetitives Lernen

Nachbarschaft in \mathbb{R}^D \leftrightarrow Nachbarschaft in \mathbb{R}^M

Neuronale Aktivität ('winner-takes-all')

$$u_0(\mathbf{x}) = \min_{\ell=1..L} u_{\ell}(\mathbf{x}) \quad \text{und für alle } \ell: \quad u_{\ell}(\mathbf{x}) = \|\mathbf{w}_{\ell} - \mathbf{x}\|^2$$

Lernen mit Nebenziel

Minimiere die Verzerrung $\sum_{\mathbf{x} \in \omega} u_0(\mathbf{x})$ unter Wahrung kleinstmöglicher Distanzabweichungen

$$\Delta_{k,\ell} = \|\mathbf{w}_k - \mathbf{w}_{\ell}\|^2 - \|\mathbf{o}_k - \mathbf{o}_{\ell}\|^2, \quad k, \ell \in \{1, \dots, L\}$$

zwischen Merkmal- und Ortsraum.

Gradientenabstiegsverfahren

(Algorithmus)

1 INITIALISIERUNG

Wähle zufällige Punkte $\mathbf{y}_1, \dots, \mathbf{y}_L \in \mathbb{R}^N$ aus.

2 ITERATIONSSCHRITT ($\forall t = 1, \dots, T$)

Berechne den Gewinnerknoten ℓ mit

$$\ell = \underset{1 \leq k \leq L}{\operatorname{argmin}} \|\mathbf{w}_k - \mathbf{x}_t\|^2$$

und aktualisiere alle (?) Prototypen:

$$\mathbf{w}_k \leftarrow \mathbf{w}_k + r_{k\ell} \cdot (\mathbf{x}_t - \mathbf{w}_k)$$

3 ABBRUCHKRITERIUM

Wiederhole Schritt 2 oder \rightsquigarrow ENDE.

(summiert log(A))

Blasenfunktion

$$r_{ij} = \begin{cases} \eta & \|\mathbf{o}_i - \mathbf{o}_j\| < \rho \\ 0 & \text{sonst} \end{cases}$$

ρ Blasenradius, η Lernrate

Gaußglocke

$$r_{ij} = \eta \cdot \exp\left(-\frac{\|\mathbf{o}_i - \mathbf{o}_j\|^2}{2\sigma^2}\right)$$

σ^2 Abklingrate, η Lernrate

Aufgabenstellung

Vektorquantisierung

Mischungsidentifikation

GMM-Klassifikatoren

Selbstorganisierende Karten

Mathematische Hilfsmittel

Entropie, Kreuzentropie und Divergenz^[7]

Stetige Formulierung — für Verteilungsdichtefunktionen

Definition

Es seien $f, g: \Omega \rightarrow \mathbb{R}$ zwei Verteilungsdichtefunktionen desselben Ereignishorizonts. Wir bezeichnen

$$\mathcal{H}(f) \stackrel{\text{def}}{=} - \int f(\mathbf{x}) \cdot \log f(\mathbf{x}) d\mathbf{x}$$

als (differentielle) **Entropie** von f ,

$$\mathcal{H}(f, g) \stackrel{\text{def}}{=} - \int f(\mathbf{x}) \cdot \log g(\mathbf{x}) d\mathbf{x}$$

als (differentielle) **Kreuzentropie** zwischen f und g und

$$\mathcal{D}(f\|g) \stackrel{\text{def}}{=} \mathcal{H}(f, g) - \mathcal{H}(f, f) = \int f(\mathbf{x}) \cdot \log \frac{f(\mathbf{x})}{g(\mathbf{x})} d\mathbf{x}$$

als **Kullback-Leibler-Divergenz** von f zu g .

Jensen-Ungleichung

Divergenz $\hat{=}$ (nicht symmetrisches) Unähnlichkeitsmaß

Lemma

1. Für alle f, g gilt $\mathcal{H}(f) = \mathcal{H}(f, f) \leq \mathcal{H}(f, g)$.
2. Für die Kullback-Leibler-Divergenz gilt stets $\mathcal{D}(f\|g) \geq 0$.
3. Der Fall $\mathcal{D}(f\|g) = 0$ tritt nur für $f \equiv g$ ein.
4. Die Werte $\mathcal{H}(f, g)$ und $\mathcal{D}(f\|g)$ lassen sich als Erwartungswerte bzgl. $f = f_{\mathbb{X}}$ deuten.

Beweis.

Verwende die Konkavität ($\log z \leq z - 1$) des Logarithmus, die für alle $z \neq 1$ strikt ist.

$$\begin{aligned} \mathcal{H}(f, f) - \mathcal{H}(f, g) &= \int f(\mathbf{x}) \cdot \log \frac{g(\mathbf{x})}{f(\mathbf{x})} d\mathbf{x} \\ &\leq \int f(\mathbf{x}) \cdot \left(\frac{g(\mathbf{x})}{f(\mathbf{x})} - 1 \right) d\mathbf{x} \\ &= \int g(\mathbf{x}) d\mathbf{x} - \int f(\mathbf{x}) d\mathbf{x} = 1 - 1 = 0 \end{aligned}$$

Kullback-Leibler-Statistik

W'keitsmodell für Datensätze mit **beobachtbaren** und **verborgenen** Variablen

Definition

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

$$P(\mathbf{x} \mid \theta) = \int P(\mathbf{x}, \mathbf{u} \mid \theta) d\mathbf{u}.$$

Für zwei Parameterfelder θ, θ' heißt der bedingte Erwartungswert

$$Q(\theta, \theta') = \mathcal{E}[\log P(\mathbf{x}, \mathbf{u} \mid \theta') \mid \mathbf{x}, \theta] = \int P(\mathbf{u} \mid \mathbf{x}, \theta) \cdot \log P(\mathbf{x}, \mathbf{u} \mid \theta') d\mathbf{u}$$

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

Beweis.

Wir drücken die ML-Zielfunktion mit Hilfe bedingter Divergenz und bedingter Kreuzentropie aus:

$$\begin{aligned} \ell(\theta') &= \log P(\mathbf{x} \mid \theta') \\ &= \log P(\mathbf{x} \mid \theta') \cdot \int P(\mathbf{u} \mid \mathbf{x}, \theta) d\mathbf{u} \\ &= \int P(\mathbf{u} \mid \mathbf{x}, \theta) \cdot \log P(\mathbf{x} \mid \theta') d\mathbf{u} \\ &= \mathcal{E}[\log P(\mathbf{x} \mid \theta') \mid \mathbf{x}, \theta] \\ &= \mathcal{E}\left[\log \frac{P(\mathbf{x}, \mathbf{u} \mid \theta')}{P(\mathbf{u} \mid \mathbf{x}, \theta')} \mid \mathbf{x}, \theta\right] \\ &= \underbrace{\mathcal{E}[\log P(\mathbf{x}, \mathbf{u} \mid \theta') \mid \mathbf{x}, \theta]}_{Q(\theta, \theta')} + \underbrace{\mathcal{E}[-\log P(\mathbf{u} \mid \mathbf{x}, \theta') \mid \mathbf{x}, \theta]}_{\mathcal{H}(\theta, \theta')} \end{aligned}$$

Unter der Voraussetzung $Q(\theta, \theta') \geq Q(\theta, \theta)$ des Satzes gilt nun aus Grund der Jensen-Ungleichung für Kreuzentropien die Behauptung, denn:

$$\begin{aligned} \ell(\theta') &= Q(\theta, \theta') + \mathcal{H}(\theta, \theta') \\ &\geq Q(\theta, \theta) + \mathcal{H}(\theta, \theta) = \ell(\theta) \end{aligned}$$

□

EM — Expectation-Maximization-Prinzip

Satz (Dempster, Laird, Rubin 1977^[7])

Für die Maximum-Likelihood-Zielfunktion

$$\ell(\theta) = \log P(\mathbf{x} \mid \theta) = \log \int P(\mathbf{x}, \mathbf{u} \mid \theta) d\mathbf{u}$$

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

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

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

Bemerkung

Das EM-Prinzip liefert ein hinreichendes Kriterium, um gegenüber θ überlegene Modellparameter θ' aufzufinden; insbesondere werden wir „glücklich“ mit:

$$\theta^* = \operatorname{argmax}_{\theta'} Q(\theta, \theta')$$

Generischer EM-Algorithmus

Konvergiert gegen ein lokales Optimum im Parameterraum

(Algorithmus)

- 1 INITIALISIERUNG
Setze $\nu = 1$ und wähle Startparameter $\theta^{(0)}$.
- 2 A POSTERIORI ERWARTUNGSWERTE
Eine Formel in den Unbekannten des Feldes θ analytisch oder simuliert
 $Q(\theta^{(\nu-1)}, \theta)$
- 3 MAXIMIERUNG DER PARAMETER
Ableiten, Nullsetzen, Auflösen ... analytisch oder iteriert
 $\theta^{(\nu)} = \operatorname{argmax}_{\theta} Q(\theta^{(\nu-1)}, \theta)$
- 4 TERMINIERUNG
Abbruch — oder $\nu \leftarrow \nu + 1$ und weiter bei 2.

(sumthogIA)

Teleskopsummation

Umformung der Kullback-Leibler-Statistik zu einer gewichteten ML-Zielgröße

Lerndatensatz (T Muster)

Beobachtbare Objekteigenschaften

$$\mathbf{x} = (x_1, \dots, x_T)$$

Verborgene Objekteigenschaften

$$\mathbf{u} = (u_1, \dots, u_T)$$

Vereinfachte Form der der KL-Statistik

$$Q(\theta, \theta') = \dots = \sum_{t=1}^T \sum_u \log P(x_t, u | \theta') \cdot \underbrace{P(u | x_t, \theta)}_{\gamma_t(u)}$$

Bemerkungen

1. $\gamma_t(u)$ ist die a posteriori Verteilung der latenten Mustereigenschaft \mathbb{U} .
2. $Q(\theta, \theta')$ sieht bis auf die Gewichte wie eine gewöhnliche ML-Zielgröße aus.
3. **M-Schritt:** Schätzformeln sind *gewichtete* arithmetische Mittelwerte!

Beweis.

$Q(\theta, \theta')$

$$\begin{aligned} &= \sum_u P(\mathbf{u} | \mathbf{x}, \theta) \cdot \log P(\mathbf{x}, \mathbf{u} | \theta') \\ &= \sum_{u_1} \dots \sum_{u_T} \left\{ \prod_{s=1}^T P(u_s | x_s, \theta) \right\} \cdot \sum_{t=1}^T \log P(x_t, u_t | \theta') \\ &= \sum_{t=1}^T \sum_{u_1} \dots \sum_{u_T} \left\{ \prod_{s=1}^T P(u_s | x_s, \theta) \right\} \cdot \log P(x_t, u_t | \theta') \\ &= \sum_{t=1}^T \sum_{u_t} \log P(x_t, u_t | \theta') \cdot \sum_{u_1} \dots \sum_{u_{t-1}} \sum_{u_{t+1}} \dots \sum_{u_T} \left\{ \prod_{s=1}^T P(u_s | x_s, \theta) \right\} \\ &= \sum_{t=1}^T \sum_{u_t} \log P(x_t, u_t | \theta') \cdot P(u_t | x_t, \theta) \cdot \underbrace{\sum_{u_1} \dots \sum_{u_{t-1}} \sum_{u_{t+1}} \dots \sum_{u_T} \prod_{s \neq t} P(u_s | x_s, \theta)}_{=1} \\ &= \sum_{t=1}^T \sum_u \log P(x_t, u | \theta') \cdot \underbrace{P(u | x_t, \theta)}_{\gamma_t(u)} \end{aligned}$$

□

Zusammenfassung (10)

1. **Unüberwachtes Lernen** dient der Modellierung **multimodaler Verteilungsdichten** oder der **Einsparung etikettierten Datenmaterials**.
2. Der **verzerrungsminimale Vektorquantisierer** besitzt ein Codebuch, dessen Prototypen die **Zellenzentroide** sind; die Quantisierung gehorcht der **Minimum-Abstand-Regel** (Voronoi-partition).
3. Der Codebuchentwurf erfolgt iterativ mit dem **Lloyd-** bzw. dem **K-means**-Austauschalgorithmus; die Zellenzahl K und eine **Anfrangspartition** sind vorzugeben.
4. Die Fälle $K \gg 2$ werden aus **Effizienz-** und **Robustheitsgründen** durch den hierarchischen **Linde-Buzo-Gray**-Topdown-Algorithmus gelöst.
5. Die Identifikation der Komponenten einer **Mischverteilung** ist (in der Theorie) **eindeutig lösbar**, wenn die Dichtefamilie eine **Orthogonalbasis** des Funktionenraums bildet.
6. In praxi werden Mischverteilungen durch eine Inkarnation des **Expectation-Maximization**-Algorithmus identifiziert, einem Iterationsverfahren mit **garantiertem Aufwuchs** der Likelihood-Zielgröße.
7. Die **entscheidungsüberwachte** Variante, der **EM***-Algorithmus, ist weniger robust gegenüber **pathologischen** Bestlösungen.