Strukturelle Modelle in der Bildverarbeitung
Markovsche Ketten, Lernen nach dem Maximum Likelihood Prinzip

D. Schlesinger – TUD/INF/KI/IS

- Maximum Likelihood Prinzip (allgemein)
- ML für Markovsche Ketten (überwacht)
- Unüberwachtes Lernen, EM Algorithmus
- EM für Markovsche Ketten

Das übergeordnete Thema – Lernen

Gegeben sei eine parametrisierte Klasse (Familie) der Wahrscheinlichkeitsverteilungen, d.h. $P(x;\Theta) \in \mathcal{P}$.

Beispiel – die Menge aller Gaussiane im $\mathbb{R}^n$

$$p(x;\mu,\sigma) = \frac{1}{(\sqrt{2\pi}\sigma)^n} \exp\left[-\frac{\|x-\mu\|^2}{2\sigma^2}\right],$$

parametrisiert mit dem Mittelwert $\mu \in \mathbb{R}^n$ und $\sigma \in \mathbb{R}$, d.h. $\Theta = (\mu, \sigma)$

Eine Lernstichprobe steht zur Verfügung: z.B. $L = (x^1, x^2, \ldots, x^{|L|})$ mit $x^l \in \mathbb{R}^n$.

Man entscheide sich für eine Wahrscheinlichkeitsverteilung aus der vorgegebenen Familie, d.h. für einen Parametersatz (z.B. $\Theta^* = (\mu^*, \sigma^*)$ für den Gaussian).

Maximum Likelihood Prinzip

Die Lernstichprobe ist eine Realisierung der unbekannten Wahrscheinlichkeitsverteilung, sie ist entsprechend der Wahrscheinlichkeitsverteilung gewürfelt.

$\Downarrow$

Das, was beobachtet wird, hat hohe Wahrscheinlichkeit

$\Downarrow$

Maximiere die Wahrscheinlichkeit der Lernstichprobe bezüglich der Parameter:

$$p(L;\Theta) \rightarrow \max_{\Theta}$$

Diskrete Wahrscheinlichkeitsverteilungen

Allgemeine diskrete Wahrscheinlichkeitsverteilung für $k \in K$,
d.h. $\Theta = p(k) \in \mathbb{R}^{|K|}$, $p(k) \geq 0$, $\sum_k p(k) = 1$.

Lernstichprobe $L = (k^1, k^2, \ldots, k^{|L|})$, $k^l \in K$.

Annahme (sehr oft):
Die Elemente der Lernstichprobe werden unabhängig von einander generiert.

ML:

$$p(L;\Theta) = \prod_l p(k^l) = \prod_k \prod_{l:k^l=k} p(k) = \prod_k p(k)^{n(k)}$$

mit den Häufigkeiten $n(k)$ der Werte k in der Lernstichprobe.

$$\ln p(L;\Theta) = \sum_k n(k) \ln p(k) \to \max_p$$

oder (bei einer unendlichen Lernstichprobe)

$$\ln p(L;\Theta) = \sum_k p^*(k) \ln p(k) \to \max_p$$

Das Schannonsche Lemma

$$\sum_i a_i \ln x_i \to \max_x, \quad \text{s.t.} \quad x_i \geq 0 \;\; \forall i, \quad \sum_i x_i = 1 \quad \text{mit} \quad a_i \geq 0$$

Methode der Lagrange Koeffizienten:

$$F = \sum_i a_i \ln x_i + \lambda \left(\sum_i x_i - 1 \right) \to \min_\lambda \max_x$$

$$\frac{dF}{dx_i} = \frac{a_i}{x_i} + \lambda = 0$$

$$\frac{dF}{\lambda} = \sum_i x_i - 1 = 0$$

$$x_i = c \cdot a_i$$

$$\sum_i c \cdot a_i - 1 = 0$$

$$x_i = \frac{a_i}{\sum_{i'} a_{i'}}$$

Die optimale Wahrscheinlichkeitsverteilung für das Beispiel 1:
Zähle, wieviel mal jedes k in der Lernstichprobe vorhanden ist und normiere auf 1.

Wahrscheinlichketsdichten

Zum Beispiel Gaussian, d.h. eine parametrisierte Wahrscheinlichketsdichte

$$p(x;\mu,\sigma) = \frac{1}{(\sqrt{2\pi}\sigma)^n} \exp\left[-\frac{\|x-\mu\|^2}{2\sigma^2}\right],$$

d.h. $\Theta = (\mu, \sigma)$, mit $\mu \in \mathbb{R}^n$, $\sigma \in \mathbb{R}$.

Lernstichprobe $L = (x^1, x^2, \ldots, x^{|L|})$ – jeder Wert von x ist nur ein Mal (?) vorhanden.

ML:

$$\ln p(L;\mu,\sigma) = \sum_l \left[-n \ln \sigma - \frac{\|x^l-\mu\|^2}{2\sigma^2}\right] =$$

$$= -|L| \cdot n \cdot \ln\sigma - \frac{1}{2\sigma^2} \sum_l \|x^l-\mu\|^2 \to \max_{\mu,\sigma}$$

$$\frac{d \ln p(L;\mu,\sigma)}{d\mu} = 0 \quad \Rightarrow \quad \mu = \frac{1}{|L|} \sum_l x^l$$

$$\frac{d \ln p(L;\mu,\sigma)}{d\sigma} = 0 \quad \Rightarrow \quad \sigma = \frac{1}{n \cdot |L|} \sum_l \|x^l-\mu\|^2$$

ML für Markovsche Ketten

Das Modell:

$$p(x, y; \Theta) = p(y_1) \cdot \prod_{i=2}^{n} p(y_i|y_{i-1}) \cdot \prod_{i=1}^{n} p(x_i|y_i)$$

Unbekannte Parameter: $\Theta = (q_1, g_i, q_i)$ mit

- Startwahrscheinlichkeitsverteilung $q_1(k)$ (entspricht $p(y_1)$),
- Übergangswahrscheinlichkeitsverteilungen $g_i(k, k')$ (entsprechen $p(y_i|y_{i-1})$),
- Bedingte Wahrscheinlichkeitsverteilungen $q_i(c, k)$ (entsprechen $p(x_i|y_i)$).

Lernstichprobe: $L = \Big((x^1, y^1), (x^2, y^2) \dots (x^l, y^l)\Big)$

Likelihood:

$$\ln p(L, \Theta) = \sum_l \ln p(x^l, y^l; \Theta) =$$

$$\sum_l \left[\ln q_1(y_1^l) + \sum_{i=2}^{n} \ln g_i(y_i^l, y_{i-1}^l) + \sum_{i=1}^{n} \ln q_i(x_i^l, y_i^l) \right]$$

$$\sum_l \ln q_1(y_1^l) + \sum_{i=2}^{n} \sum_l \ln g_i(y_i^l, y_{i-1}^l) + \sum_{i=1}^{n} \sum_l \ln q_i(x_i^l, y_i^l) \to \max_\Theta$$

Jeder Summand kann unabhängig von anderen optimiert werden!!!

$$\sum_l \ln q_1(y_1^l) = \sum_k \sum_{l:y_1^l=k} \ln q_1(y_1^l) = \sum_k \sum_{l:y_1^l=k} \ln q_1(k) = \sum_k n_1(k) \ln q_1(k) \to \max_{q_1}$$

mit relativen Häufigkeiten $n_1(k)$ (in der Lernstichprobe) der Zustände im ersten Zeitpunkt. Nach dem Schannonschen Lemma (da $q_1(k) \geq 0$ und $\sum_k q_1(k) = 1$)

$$q_1(k) \sim n_1(k)$$

(Fast) Analog für alle Übergangsmatrizen $g_i(k, k')$:

$$\sum_l \ln g_i(y_i^l, y_{i-1}^l) = \sum_k \sum_{k'} \sum_{l:y_i^l=k, y_{i-1}=k'} \ln g_i(k, k') =$$

$$\sum_k \sum_{k'} n_i(k, k') \ln g_i(k, k') \to \max_{g_i}$$

$$\sum_k n_i(k, k') \ln g_i(k, k') \to \max_{g_i} \quad \forall k'$$

$$\Rightarrow g_i(k, k') \sim n_i(k, k')$$

Setze die Parameter des Modells auf die aus der Lernstichprobe entnommenen Statistiken.

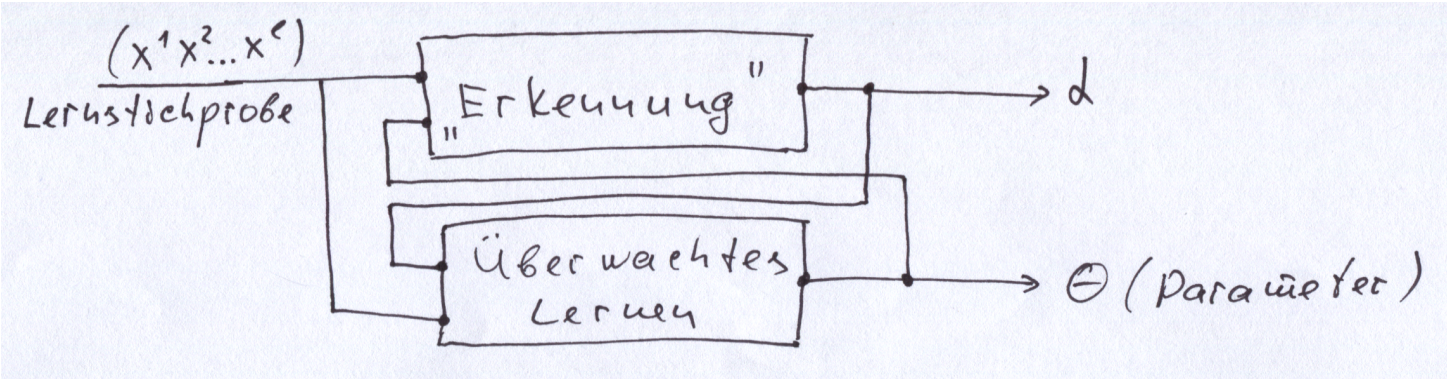
Unüberwachtes Lernen, EM (Idee)

(Allgemein): Das Modell ist eine Wahrscheinlichkeitsverteilung $p(x, k; \Theta)$
für Paare x (Beobachtung) und k (Klasse)

In der Lernstichprobe ist die Information unvollständig
– die Klasse wird nicht beobachtet, d.h. $L = (x^1, x^2 \dots x^l)$

Die Aufgabe nach dem Maximum Likelihood Prinzip: $\ln p(L; \Theta) \to \max_{\Theta}$

Die Idee – ein iteratives Verfahren:



1. „Erkennung“ (Vervollständigung der Daten): $(x^1, x^2 \dots x^l)$, $\Theta \Rightarrow$ „Klassen“
2. Überwachtes Lernen: „Klassen“, $(x^1, x^2 \dots x^l) \Rightarrow \Theta$

Achtung!!! Bayessche Erkennung ist nicht möglich, denn es gibt keine Kostenfunktion.

$$\ln p(L;\Theta) = \sum_l \ln p(x^l) = \sum_l \ln \sum_k p(x^l, k; \Theta) \to \max_\Theta$$

Expectation-Maximization Algorithmus:

Man führt eine „Nahrhafte Eins“ wie folgt ein:

$$\sum_l \left[\sum_k \alpha_l(k) \ln p(k, x^l; \Theta) - \sum_k \alpha_l(k) \ln \frac{p(k, x^l; \Theta)}{\sum_{k'} p(k', x^l; \Theta)} \right]$$

mit $\alpha_l(k) \geq 0$, $\sum_k \alpha_l(k) = 1$ für alle l.

Dann ist dieser Ausdruck dem oberen äquivalent (Beweis nur für einen Muster x^l):

$$\sum_k \alpha_l(k) \ln p(k, x^l; \Theta) - \sum_k \alpha_l(k) \ln \frac{p(k, x^l; \Theta)}{\sum_{k'} p(k', x^l; \Theta)} =$$

$$\sum_k \left[\alpha_l(k) \ln p(k, x^l; \Theta) - \left[\alpha_l(k) \ln p(k, x^l; \Theta) - \alpha_l(k) \ln \sum_{k'} p(k', x^l; \Theta) \right] \right] =$$

$$\sum_k \alpha_l(k) \ln \sum_{k'} p(k', x^l; \Theta) = \ln \sum_{k'} p(k', x^l; \Theta) \cdot \sum_k \alpha_l(k) = \ln \sum_{k'} p(k', x^l; \Theta)$$

$$\ln p(L;\Theta) = F(\Theta,\alpha) - G(\Theta,\alpha), \quad \text{mit}$$

$$F(\Theta,\alpha) = \sum_l \sum_k \alpha_l(k) \ln p(k, x^l;\Theta)$$

$$G(\Theta,\alpha) = \sum_l \sum_k \alpha_l(k) \ln \frac{p(k, x^l;\Theta)}{\sum_{k'} p(k', x^l;\Theta)} = \sum_l \sum_k \alpha_l(k) \ln p(k|x^l;\Theta)$$

Man starte mit einem beliebigen Parametersatz $\Theta^{(0)}$ und wiederhole:

Expectation Schritt – „die Fehlenden Daten vervollständigen“:
Man wähle $\alpha^{(t)}$ so, dass das Maximum von G bezüglich Θ genau an der Stelle $\Theta^{(t)}$ eintritt.
Laut Schannonsches Lemma:

$$\alpha_l^{(t)}(k) = p(k|x^l;\Theta^{(t)})$$

Achtung!!! Das ist keine Optimierung, das ist eine Abschätzung der oberen Schranke für G.

Maximization Schritt – „überwachtes Lernen“:
Man maximiere F bezüglich Θ:

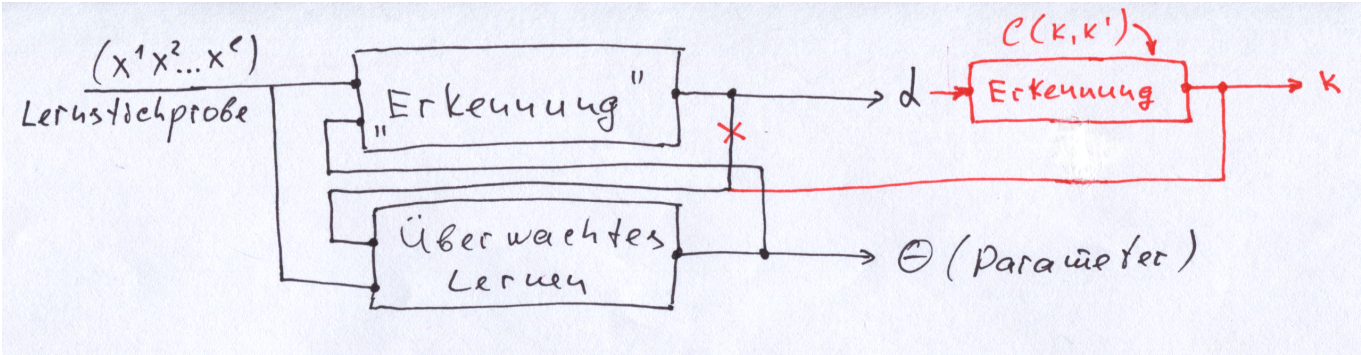
$$\Theta^{(t+1)} = \arg\max_{\Theta} F(\Theta, \alpha^{(t)})$$

Zusätzliche Bemerkungen

Der Maximum-Likelihood Schätzer ist **konsistent**,
d.h. er liefert die tatsächlichen Parameter bei unendlichen Lernstichproben.

Der Maximum-Likelihood Schätzer ist nicht immer **erwartungswerttreu**,
d.h. bei endlichen Lernstichproben stimmt der Mittelwert des geschätzten Parameters nicht unbedingt mit dem tatsächlichen überein.
Beispiele: ML für μ ist erwartungswerttreu, ML für σ – nicht.

Expectation-Maximization Algorithmus konvergiert immer,
aber nur zum lokalen Optimum (nicht global).



Ersetzt man im Expectation Schritt die Berechnung der a-posteriori Wahrscheinlichkeiten durch Erkennung, so erhält man etwas, was dem K-Means Algorithmus ähnlich ist. Oft nennt man das (fälschlicherweise) „EM-like Scheme". Das ist **kein** ML! Allerdings ist dies sehr populär, denn es ist unter Umständen viel einfacher anstatt der benötigten marginalen Verteilungen zum Beispiel MAP-Entscheidung zu berechnen.

EM für Markovsche Ketten

Das Modell:

$$p(x, y; \Theta) = p(y_1) \cdot \prod_{i=2}^{n} p(y_i|y_{i-1}) \cdot \prod_{i=1}^{n} p(x_i|y_i)$$

Unbekannte Parameter: $\Theta = (q_1, g_i, q_i)$ mit

- Startwahrscheinlichkeitsverteilung $q_1(k)$ (entspricht $p(y_1)$),
- Übergangswahrscheinlichkeitsverteilungen $g_i(k, k')$ (entsprechen $p(y_i|y_{i-1})$),
- Bedingte Wahrscheinlichkeitsverteilungen $q_i(c, k)$ (entsprechen $p(x_i|y_i)$).

Lernstichprobe: (jetzt anders!!!) $L = (x^1, x^2 \dots x^l)$

Die nicht beobachtbaren Folgen y entsprechen den „Klassen“ k im allgemeinen Fall – das, worüber summiert wird.

Likelihood:

$$\ln p(L, \Theta) = \sum_l \ln p(x^l; \Theta) = \sum_l \ln \sum_y p(x^l, y; \Theta) \to \max_\Theta$$

Man führe $\alpha_l(y) \geq 0$ mit $\sum_y \alpha_l(y) = 1$ für alle l ein.

In **Expectation** Schritt setzt man $\alpha_l(y) = p(y|x^l; \Theta)$.
Dies ist technisch nicht möglich – nur Hilfskonstrukt.

Maximization Schritt:

$$\sum_l \sum_y \alpha_l(y) \ln p(x^l, y; \Theta) =$$

$$\sum_l \sum_y \alpha_l(y) \cdot \left[\ln q_1(y_1) + \sum_{i=2}^{n} \ln g_i(y_i, y_{i-1}) + \sum_{i=1}^{n} \ln q_i(x_i^l, y_i) \right] =$$

$$\sum_k \alpha_1(k) \ln q_1(k) + \sum_{i=2}^{n} \sum_{kk'} \alpha_i(k, k') \ln g_i(k, k') + \sum_{i=1}^{n} \alpha_i(c, k) \ln q_i(c, k) \to \max_\Theta$$

mit

$$\alpha_1(k) = \sum_l \sum_{y: y_1 = k} \alpha_l(y) = \sum_l \sum_{y: y_1 = k} p(y|x^l; \Theta) = \sum_l p(y_1 = k|x^l; \Theta)$$

$$\alpha_i(k, k') = \sum_l \sum_{y: y_i = k, y_{i-1} = k'} \alpha_l(y) = \sum_l p(y_i = k, y_{i-1} = k'|x^l; \Theta)$$

$$\alpha_i(c, k) = \sum_l \sum_{y: y_i = k, x_i = c} \alpha_l(y) = \sum_l p(y_i = k, x_i = c|x^l; \Theta)$$

Die $\alpha_l(y)$ werden nicht explizit benötigt, sondern nur die „marginalen“ α-s.
Diese lassen sich effizient mit dem SumProd Algorithmus berechnen.

EM für Markovsche Ketten

Zusammenfassend:

Initialisiere die Parameter $q_1^{(0)}$, $g_i^{(0)}$ und $q_i^{(0)}$

Wiederhole

1. **Expectation**: berechne

$$\alpha_1(k) = \sum_l p(y_1 = k | x^l; q_1^{(t)}, g_i^{(t)}, q_i^{(t)})$$

$$\alpha_i(k, k') = \sum_l p(y_i = k, y_{i-1} = k' | x^l; q_1^{(t)}, g_i^{(t)}, q_i^{(t)})$$

$$\alpha_i(c, k) = \sum_l p(y_i = k, x_i = c | x^l; q_1^{(t)}, g_i^{(t)}, q_i^{(t)})$$

   mit dem SumProd Algorithmus.

2. **Maximization**: setze

$$q_1^{(t+1)}(k) \sim \alpha_1(k) \quad \text{normiert so, dass} \quad \sum_k q_1^{(t+1)}(k) = 1$$

$$g_i^{(t+1)}(k, k') \sim \alpha_i(k, k') \quad \text{normiert so, dass} \quad \sum_k g_i^{(t+1)}(k, k') = 1 \quad \forall k'$$

$$q_i^{(t+1)}(c, k) \sim \alpha_i(c, k) \quad \text{normiert so, dass} \quad \sum_c q_i^{(t+1)}(c, k) = 1 \quad \forall k$$