

§76: VERBORGENE MARKOWMODELLE

76.1 Motivation

Verborgene Markowmodelle (hidden Markov models, HMM) spielen eine große Rolle in der Bioinformatik (Genomanalyse), Spracherkennung und Mustererkennung. Mit Hilfe der aus dem Bereich der Markowketten bekannten Übergangsmatrizen (§75) sucht man nach der wahrscheinlichsten Zustandsfolge, die eine gegebene Beobachtung erzeugt haben könnte.

76.2 Beispiel: Spieler mit fairer und unfaierer Münze

Ein Spieler besitzt eine faire Münze, bei der Kopf und Zahl gleich wahrscheinlich sind ($p^+(0) = \frac{1}{2} = p^+(1)$) und eine gezinkte Münze, bei der Zahl wahrscheinlicher ist ($p^+(0) = \frac{1}{4}, p^+(1) = \frac{3}{4}$). Wir beobachten n Münzwürfe und wollen entscheiden, ob er die faire oder die unfaire Münze genommen hat.

Sei k die Zahl der Ergebnisse, bei denen Zahl beobachtet wurde. Die einzelnen Ergebnisse seien $x_1, \dots, x_n$.

Fall 1: Münze war fair.

Wahrscheinlichkeit, dass die Beobachtung $x = (x_1, \dots, x_n)$ eintritt, ist

$$P(x|\text{faire Münze}) = \prod_{i=1}^n p^+(x_i) = \frac{1}{2^n}$$

Fall 2: Münze war unfair.

$$P(x|\text{unfaire Münze}) = \left(\frac{1}{4}\right)^{n-k} \left(\frac{3}{4}\right)^k = \frac{3^k}{4^n}$$

Wir vermuten, dass die Münze fair war, falls

$$P(x|\text{faire Münze}) > P(x|\text{unfaire Münze})$$

$$\begin{aligned} \frac{1}{2^n} &> \frac{3^k}{4^n} \\ \Rightarrow 2^n &> 3^k \\ \Rightarrow n &> k \log_2 3 \\ \Rightarrow \frac{k}{n} &< \frac{1}{\log_2 3} \quad (*) \end{aligned}$$

76.3 Schwierigeres Beispiel

Wir nehmen an, dass der Spieler mit einer geringen Wahrscheinlichkeit von 0.1 die Münze zwischen zwei Würfeln austauscht und wollen wissen, wann er welche Münze genommen hat.

Naive Ansatz: Wir betrachten die Beobachtung innerhalb eines Fensters und überprüfen, ob (*) zutrifft oder nicht.

Problem: Lösung hängt von der Fenstergröße ab und wir wissen nicht, wann der Spieler die Münze wechselt. Verborgene Markowmodelle bieten einen alternativen Zugang.

76.4 Def.:

Ein verborgenes Markowmodell (hidden Markov model, HMM) ist ein Tupel $\mathcal{M} = (\Sigma, Q, M, E)$. Dabei bezeichnen:

- Σ : Alphabet von Symbolen (Ann.: $|\Sigma| = r$)
- Q : Menge von Zuständen mit Symbolen aus Σ (Ann.: $|Q| = k$)
- M : stochastische Matrix der Übergangswahrscheinlichkeiten

$$M = (p_{ij}) \in \mathbb{R}^{k \times k}$$

- E : Matrix der Emissionswahrscheinlichkeiten

$$E \in \mathbb{R}^{k \times r}, \quad E = (e_i(b))$$

76.5 Beispiel

In Beispiel 76.3 bezeichnen:

- $\Sigma = \{0, 1\}$ Kopf (0) oder Zahl (1)
- $Q = \{f, b\}$ faire (f) oder unfaire (b =biased) Münze
- $M = \begin{pmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{pmatrix}$ Übergangsmatrix für Münzwechsel
- E : Matrix mit Komponenten:
 $e_f(0) = \frac{1}{2}, e_f(1) = \frac{1}{2}$ Emissionswahrscheinlichkeiten für faire Münze
 $e_b(0) = \frac{1}{4}, e_b(1) = \frac{3}{4}$ Emissionswahrscheinlichkeiten für unfäre Münze

Die vom Spieler tatsächlich verwendete Münzfolge ist $\pi = (\pi_1, \dots, \pi_n)$ mit $\pi_i \in Q$.

Sie bleibt uns „verborgen“.

Wir beobachten nur den Ergebnisvektor $x = (x_1, \dots, x_n)$ mit $x_i \in \Sigma$. Die Wahrscheinlichkeit, dass eine beobachtete Sequenz x durch einen „Pfad“ π erzeugt wurde, ist

$$P(x|\pi) = \prod_{i=1}^n P(x_i|\pi_i) \cdot \underbrace{P(\pi_i|\pi_{n+1})}_{\text{Übergangswahrscheinlichkeit}} \quad \text{mit } P(\pi_n|\pi_{n+1}) := 1$$

76.6 Problemstellung

Gelöst werden soll das Dekodierungsproblem:

Finde zu einem gegebenen HMM $\mathcal{M} = (\Sigma, Q, M, E)$ und einer Beobachtung $x = (x_1, \dots, x_n)$ einen optimalen Pfad $\pi^* = (\pi_1^*, \dots, \pi_n^*)$, der $P(x|\pi)$ maximiert.

$$\pi^* = \operatorname{argmax}_{\pi} P(x|\pi)$$

76.7 Der Viterbi-Algorithmus

Grundidee: optimaler Pfad $(\pi_1^*, \dots, \pi_{i+1}^*)$ zur Beobachtung $(x_1, \dots, x_{i+1})$ ist optimal unter allen Pfaden, die in dem (unbekannten) Zustand $\pi_i^* = m \in Q$ enden.

$s_m(i)$: Wahrscheinlichkeit, dass optimale Pfad $(\pi_1^*, \dots, \pi_i^*)$ zur Beobachtung $(x_1, \dots, x_i)$ in m endet ($1 \leq i \leq n$).

$$\Rightarrow s_l(i+1) = \underbrace{e_l(x_{i+1})}_{\text{Emissionswahrscheinlichkeit für } l \text{ zur Zeit } i+1} \cdot \underbrace{\max_{m \in Q} \{s_m(i) \cdot p_{lm}\}}_{\text{Übergang } m \rightarrow l} \quad (**)$$

Initialisiere:

$$\begin{aligned} s_0(0) &= 1 \\ s_m(0) &= 0 \quad m \in \{1, \dots, n\} \end{aligned}$$

Für den gesuchten Wert π^* ist

$$P(x|\pi^*) = \max_{m \in Q} \{s_m(i) \cdot p_{n_m}\}$$

Die Laufzeit für den Viterbi-Algorithmus beträgt $\mathcal{O}(kn)$. Leider liefert der Viterbi-Algorithmus nur ein lokales Optimum, wird aber aus Effizienzgründen gerne verwendet.