



## Übungsblatt 7

Ausgabe: 23.06.2016; Abgabe: bis 07.07.2016, 23:59 Uhr.

### Aufgabe 1 (Hidden Markov Model)

(20 Punkte)

Gegeben sei das folgende Hidden Markov Model:

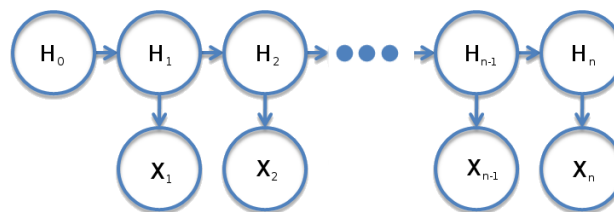


Abbildung 1: Das zu untersuchende Hidden Markov Model

Die Übergangswahrscheinlichkeiten der verborgenen Zustände seien gegeben durch

$$\begin{aligned}
 & \begin{matrix} H_1 = 1 & H_1 = 2 \\ H_0 = 1 & \begin{pmatrix} 0.4 & 0.6 \end{pmatrix}, \end{matrix} \quad \begin{matrix} H_2 = 1 & H_2 = 2 \\ H_1 = 1 & \begin{pmatrix} 0.3 & 0.7 \\ 0.7 & 0.3 \end{pmatrix}, \end{matrix} \quad \begin{matrix} H_3 = 1 & H_3 = 2 \\ H_2 = 1 & \begin{pmatrix} 0.8 & 0.2 \\ 0.2 & 0.8 \end{pmatrix}, \end{matrix} \\
 & \begin{matrix} H_4 = 1 & H_4 = 2 \\ H_3 = 1 & \begin{pmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{pmatrix}, \end{matrix} \quad \begin{matrix} H_5 = 1 & H_5 = 2 \\ H_4 = 1 & \begin{pmatrix} 0.6 & 0.4 \\ 0.4 & 0.6 \end{pmatrix} \end{matrix}
 \end{aligned}$$

und die der sichtbaren Zustände durch

$$\begin{aligned}
 & \begin{matrix} X_1 = A & X_1 = B \\ H_1 = 1 & \begin{pmatrix} 0.1 & 0.9 \\ 0.9 & 0.1 \end{pmatrix}, \end{matrix} \quad \begin{matrix} X_2 = C & X_2 = D \\ H_2 = 1 & \begin{pmatrix} 0.4 & 0.6 \\ 0.6 & 0.4 \end{pmatrix}, \end{matrix} \quad \begin{matrix} X_3 = E & X_3 = F \\ H_3 = 1 & \begin{pmatrix} 0.7 & 0.3 \\ 0.3 & 0.7 \end{pmatrix}, \end{matrix} \\
 & \begin{matrix} X_4 = G & X_4 = H \\ H_4 = 1 & \begin{pmatrix} 0.6 & 0.4 \\ 0.4 & 0.6 \end{pmatrix}, \end{matrix} \quad \begin{matrix} X_5 = I & X_5 = J \\ H_5 = 1 & \begin{pmatrix} 0.8 & 0.2 \\ 0.2 & 0.8 \end{pmatrix}. \end{matrix}
 \end{aligned}$$

- a) Man erstelle in Matlab eine geeignete Repräsentation des Hidden Markov Models. (Siehe dazu auch die unten stehenden Hinweise.)
- b) Man implementiere eine Funktion `forward(X, t, model)`, die zu einer gegebenen Sequenz  $X_1, \dots, X_n$  an der  $t$ -ten Stelle den *Forward-Algorithmus* auswertet. (Wenn das Modell aus mehreren Variablen besteht, verändert sich die Signatur der Funktion entsprechend.) (4 Punkte)
- c) Man implementiere eine Funktion `backward(X, t, model)`, die zu einer gegebenen Sequenz  $X_1, \dots, X_n$  an der  $t$ -ten Stelle den *Backward-Algorithmus* auswertet. (Für Modelle aus mehreren Variablen gilt derselbe Hinweis wie in der letzten Teilaufgabe.) (4 Punkte)
- d) Man implementiere eine Funktion `viterbi(X, model)`, die zu einer gegebenen Sequenz  $X_1, \dots, X_n$  den *Viterbi-Algorithmus* auswertet. (Die Ausgabe ist hier keine Zahl, sondern ein Pfad.) (8 Punkte)
- e) Man bestimme  $\mathbb{P}(H_5 = 1 | X = \text{ACFGI})$ . (1 Punkt)
- f) Man berechne  $\mathbb{P}(H_5 = 1 | X = \text{BCEGJ})$ . (1 Punkt)
- g) Man bestimme die wahrscheinlichste Sequenz  $H_1, \dots, H_n$ , die die Sequenz  $X = \text{ACFGI}$  hervorgerufen haben könnte. (2 Punkte)

#### Hinweise:

- Ist es für die Rekursion nötig, dann darf den Anfangs- und Endzuständen die Wahrscheinlichkeit 1 zuwiesen werden (wie in der Vorlesung erwähnt).
- Aufgabenteil d) ist nicht trivial, wenn Ihr nicht viel Erfahrung bei der Matlab-Programmierung habt. Überlegt Euch dazu, wie man sich während der Suche nach dem Maximum die Knoten merken kann.
- Um die Abtipparbeit der obigen Übergangswahrscheinlichkeiten zu reduzieren, wird das Matlab-Skript `createHmm.m` zur Verfügung gestellt.