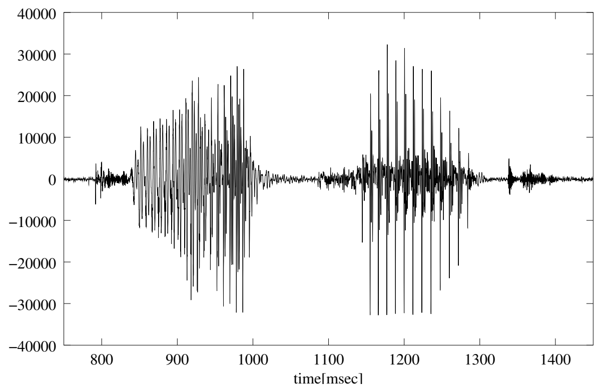
12 Hidden Markov Modelle

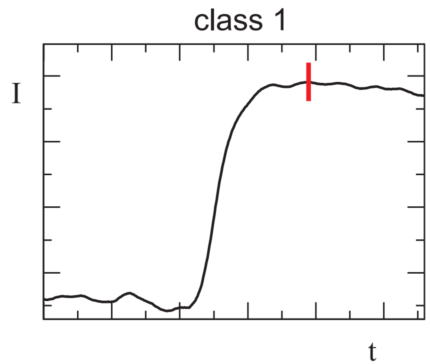
Muster sind oft als Signale über der Zeit gegeben, diskretisiert also als Folge

$$\{\vec{c}_t\}, \text{ mit } t = 1, \ldots, T$$

falls die Länge dieser Folge je Muster variiert, können wir die $\vec{c}_t$ auch nicht zu einem neuen Merkmalsvektor zusammenfassen $\vec{c}_1, \ldots, \vec{c}_T)^T$

Beispiele: gesprochene Wörter, EKG-Signale oder allgemein Messsignale

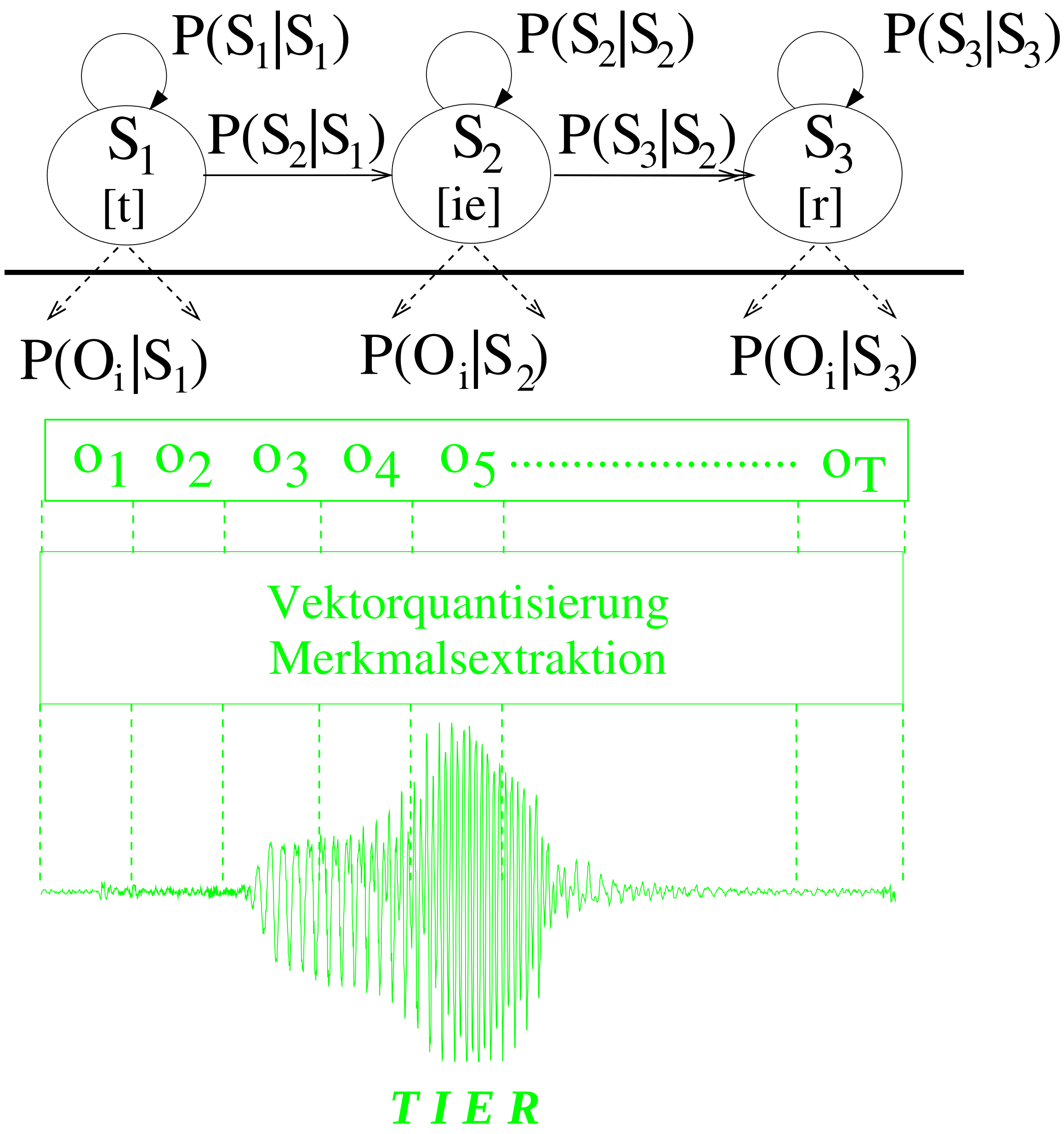




für die Klassifikation solcher Muster sind **Hidden Markov Modelle** sehr geeignet

Anwendungen in der Bioinformatik:

- Gen-Detektion
- multiples Alignment
- Modellierung/Detektion von Domänen, motifs (variabler Länge)



Modell

- interne Zustände mit Übergangswahrscheinlichkeiten $P(S_j \mid S_i)$
- stochastisch emittierte Symbole $P(O_k \mid S_j)$

(Sprach-)Erkennung

- je Musterklasse (z.B. Wort) ein HMM
- Lernen aus Beispielen
- entscheide für das Wort, dessen HMM am besten “passt”

Realer Prozess (hier: Sprache)

- Sprachsignal $\rightarrow$ Symbole

12.1 Das Modell

Markov Modelle beschreiben einen stochastischen Prozeß, der zeitdiskret Zustände annimmt und in jedem Zustand ein Symbol emittiert

Zustände

- endlich Menge von *Zuständen* $\mathbf{S} = \{S_1, S_2, \ldots, S_N\}$
- diskrete Folge $\vec{s} = (s_1, s_2, \ldots, s_t, \ldots, s_T)$ von T eingenommenen Zuständen s_t zum Zeitpunkt t, $s_t \in \mathbf{S}$
- Wahrscheinlichkeit $P(s_t = S_j)$ (zum Zeitpunkt t befinden wir uns im Zustand S_j) hängt **nur** von Zustand in $t-1$ ab (Markov Prozessen erster Ordnung):

$$P(s_t = S_j \mid s_{t-1}, \ldots, s_1) = P(s_t = S_j \mid s_{t-1})$$

- Übergangswahrscheinlichkeiten in $N \times N$-Matrix

$$\underline{A} = [a_{ij}] \quad \text{mit} \quad a_{ij} = P(s_t = S_j \mid s_{t-1} = S_i) \quad \text{für} \quad 1 \leq i, j \leq N$$

- Initialisierung eines Prozesses durch Anfangswahrscheinlichkeiten:

$$\underline{\pi} = [\pi_i] = [P(s_1 = S_i)] \text{ , } i = 1, \ldots, N$$

Ausgabe

- bei Einnahme eines Zustands wird jeweils eine Ausgabe erzeugt, die beobachtet werden kann (im Gegensatz zum Zustand selbst)
- Ausgabe ist
  - Symbol aus einem endlichen Symbolvorrat $\mathbf{O} = \{O_1, \ldots, O_q\}$ $\rightarrow$ diskretes HMM
  - Vektor $\vec{c} \in \mathbb{R}^q$ $\rightarrow$ kontinuierliches HMM
- auch die Ausgabe wird stochastisch erzeugt und hängt nur von eingenommenen Zustand ab:
  - diskretes HMM: Ausgabewahrscheinlichkeiten als $N \times q$-Matrix

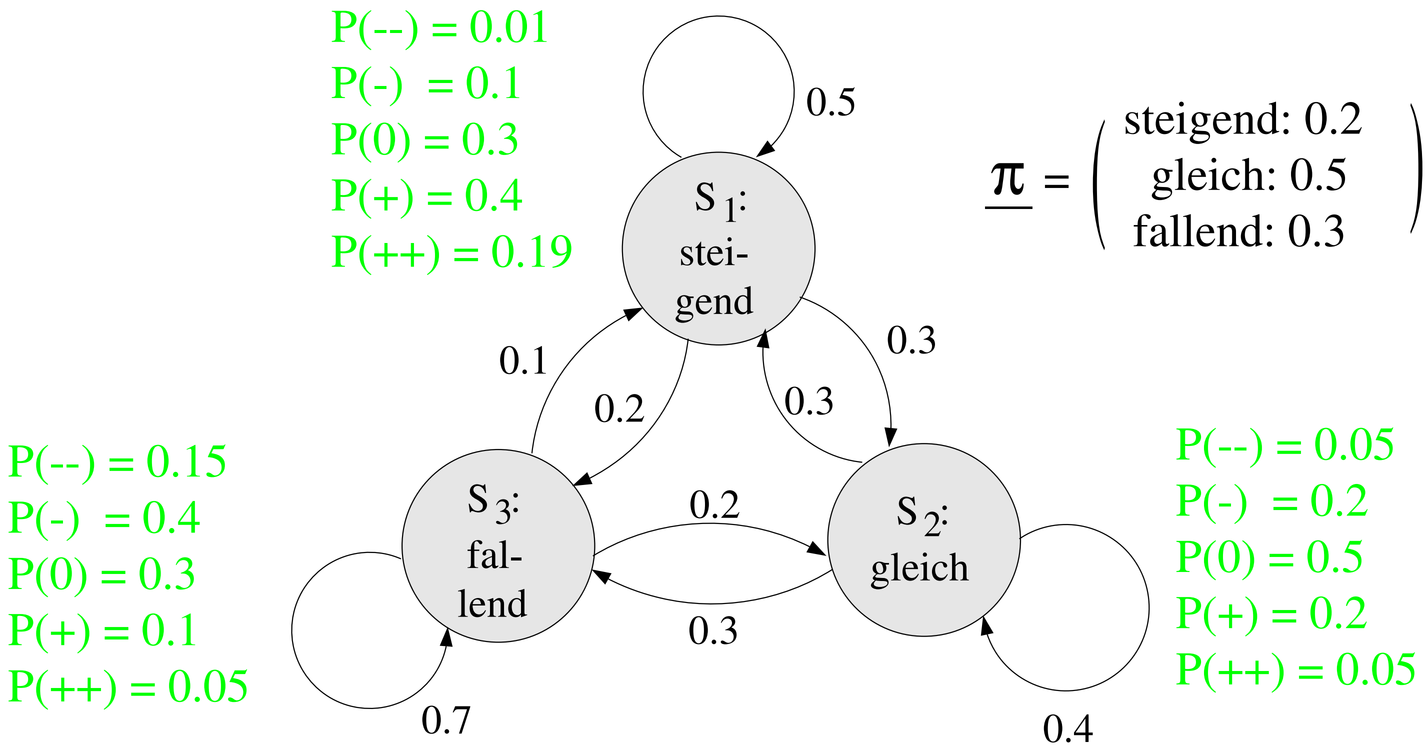
$$\underline{B} = [b_{jk}] \quad \text{mit} \quad b_{jk} = P(o_t = O_k \mid s_t = S_j) \quad \text{für} \quad 1 \leq j \leq N, 1 \leq k \leq q$$

  - kontinuierliches HMM: N-dimensionaler Vektor von Dichten:

$$\underline{B} = [b_j] \quad \text{mit} \quad b_j(\vec{c}) = p(o_t = \vec{c} \mid s_t = S_j) \quad \text{für} \quad 1 \leq j \leq N, \vec{c} \in \mathbb{R}^q$$

Ein **HMM** λ ist also durch das Tripel $\lambda = (\underline{\pi}, \underline{A}, \underline{B})$ vollständig bestimmt

Beispiel: Modellierung des Kursverhaltens an der Börse als HMM



Aktueller Zustand (leider) nicht beobachtbar,
sondern nur die Kursdifferenz zum Vortag: $\{--, -, 0, +, ++\}$

drei **zentrale Probleme** bei Verwendung von HMMs

1. berechne die Produktionswahrscheinlichkeit $P(\underline{o} \mid \lambda)$ für eine Beobachtungsfolge $\underline{o} = o_1 o_2 \ldots o_T$

   (Wie groß ist die WK, daß folgende Folge von Kursdifferenzen auftritt? $\{++, --, +, 0, ++\}$)

2. berechne die Zustandsfolge $\underline{s}^* = s_1 s_2 \ldots s_T$, die mit größter Wahrscheinlichkeit zur Ausgabe der Folge $\underline{o} = o_1 o_2 \ldots o_T$ geführt hat?
   d.h. $P(\underline{s}^*, \underline{o} \mid \lambda)$ ist maximal bzgl. aller möglichen Zustandsfolgen $\underline{s} \in \mathbf{S}$

   (Welche Zustandsfolge ist am wahrscheinlichsten, falls obige Folge von Kursdifferenzen auftritt?)

3. bestimme automatisch die Parameter des HMM für eine gegebene Beobachtung $\underline{o}$, z.B. ML-Schätzung

$$\lambda^* = \underset{\lambda=(\underline{\pi},\underline{A},\underline{B})}{\operatorname{argmax}} P(\underline{o} \mid \lambda) \qquad \text{bzw.} \qquad \lambda^* = \underset{\lambda=(\underline{\pi},\underline{A},\underline{B})}{\operatorname{argmax}} P(\underline{o}, \underline{s}^* \mid \lambda)$$

12.2 Berechnung der Produktionswahrscheinlichkeit

effiziente Lösung mit Hilfe der dynamischen Programmierung

(im diskreten Fall analog mit $b_j(o_t) \to b_{jk},$ für $o_t = O_k$)

- sei $\alpha_{tj} := P(o_1 \dots o_t, s_t = S_j \mid \lambda)$
  die Wahrscheinlichkeit, die ersten t Ausgaben von O zu beobachten und im Zeitpunkt t im Zustand S_j zu sein
  dann gilt

$$\alpha_{1j} = \pi_j b_j(o_1) \qquad \text{für} \quad j = 1, \dots, N \tag{12.1}$$

$$\alpha_{t+1,j} = \left(\sum_{i=1}^{N} \alpha_{ti} a_{ij} \right) b_j(o_{t+1}) \qquad \text{für} \quad t = 1, \dots, T-1, \quad j = 1, \dots, N \tag{12.2}$$

$$P(\underline{o} \mid \lambda) = \sum_{i=1}^{N} \alpha_{Ti} \tag{12.3}$$

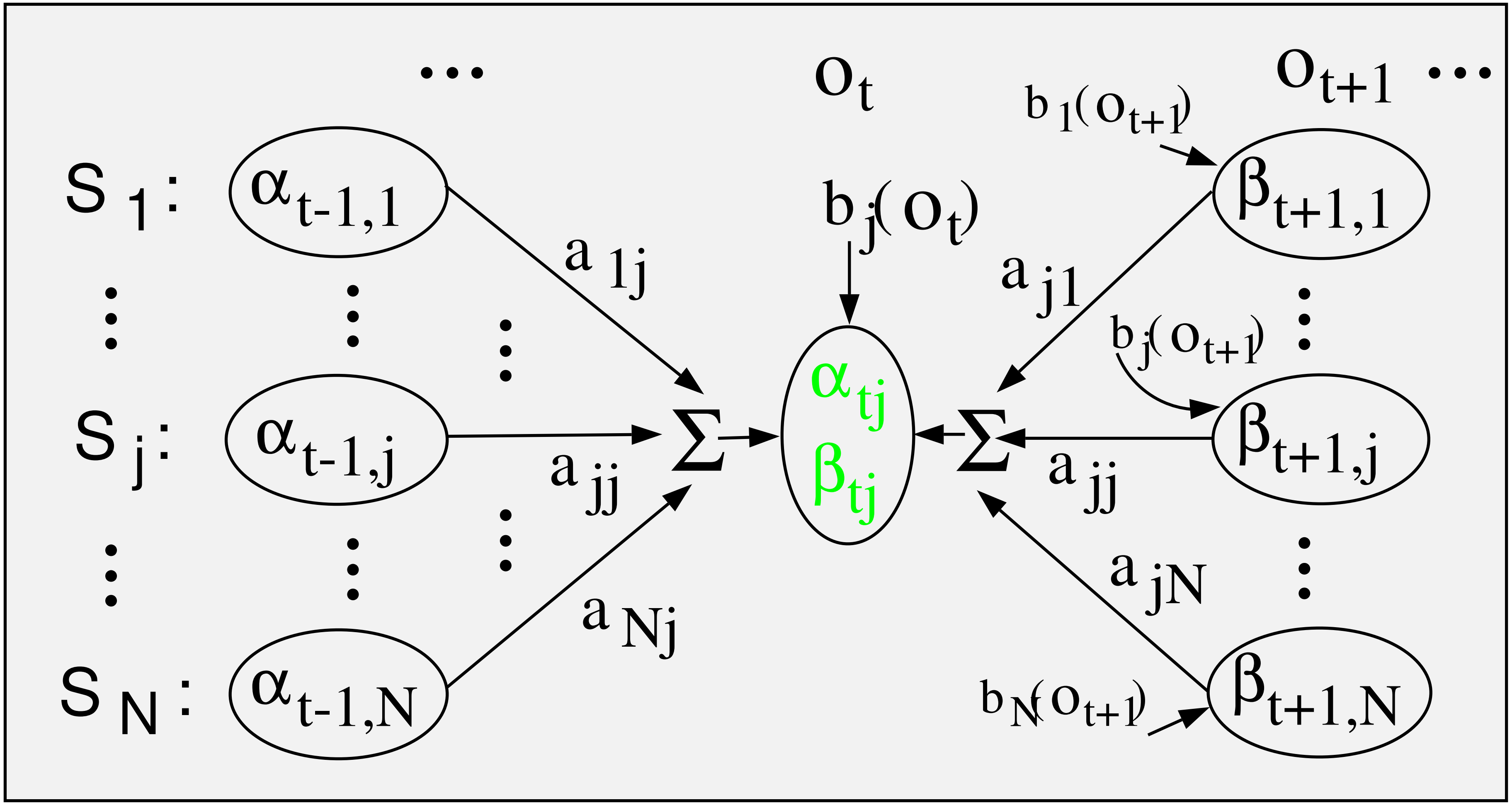
- der resultierende Algorithmus heißt **forward-Algorithmus**
  Komplexität $\Theta(N^2 \cdot T)$

- analog läßt sich eine Rückwärtsrekursion angeben
  sei $\beta_{tj} := P(o_{t+1} \ldots o_T, s_t = S_j \mid \lambda)$
  die Wahrscheinlichkeit, die Symbole ab dem Zeitpunkt $t+1$ zu beobachten, falls man zum Zeitpunkt t im Zustand S_j ist
  dann gilt:

$$
\begin{aligned}
\beta_{Tj} &= 1 \qquad \text{für} \quad j = 1, \ldots, N \\
\beta_{tj} &= \sum_{i=1}^{N} \beta_{t+1,i} a_{ji} b_i(o_{t+1}) \qquad \text{für} \quad t = T-1, \ldots, 1, \quad j = 1, \ldots, N \\
P(\underline{o} \mid \lambda) &= \sum_{i=1}^{N} \pi_i b_i(o_1) \beta_{1i}
\end{aligned}
$$

- der resultierende Algorithmus heißt **backward-Algorithmus**
- gemäß der Definition der Variablen α und β gilt ebenfalls:

$$
P(\underline{o} \mid \lambda) = \sum_{i=1}^{N} \alpha_{ti} \beta_{ti} \qquad \text{für beliebiege } t
$$



Rekursives Schema zur Berechnung der Produktionswahrscheinlichkeit

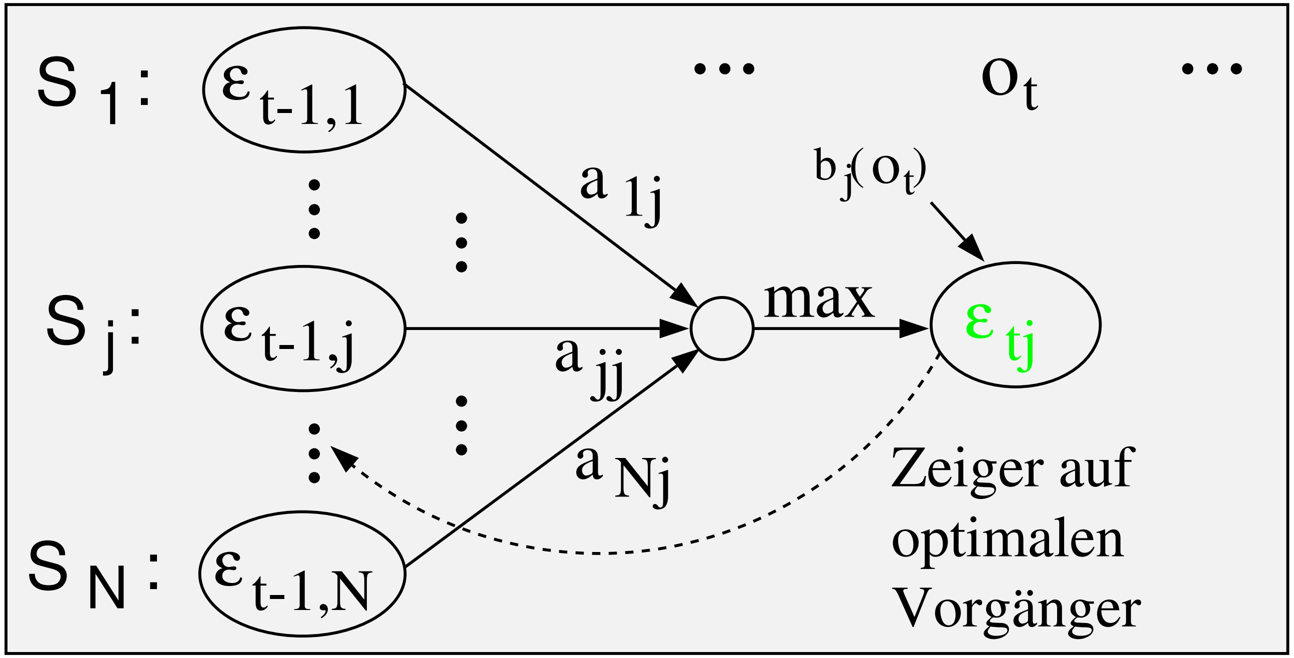
12.3 Berechnung der optimalen Zustandsfolge

- ebenfalls sehr effizient mit Hilfe der dynamischen Programmierung, wird als **Viterbi-Algorithmus** bezeichnet
- analog der Berechnung der α-Variablen in (12.1), es wird lediglich die Summe durch das Maximum ersetzt und der zugehörige Vorgängerknoten gespeichert wird
- sei ϵ_{tj} die Wahrscheinlichkeit der optimalen Zustandfolge, welche die ersten t Ausgaben beobachtet und im Zustand S_j endet, d.h. $s_t = S_j$

  dann gilt

$$\begin{aligned} \epsilon_{1j} &= \pi_j b_j(o_1) \qquad \text{für} \quad j = 1, \ldots, N \\ \epsilon_{t+1,j} &= \left(\max_{i=1,\ldots,N} \epsilon_{ti} a_{ij} \right) b_j(o_{t+1}) \qquad \text{für} \quad t = 1, \ldots, T-1, \quad j = 1, \ldots, N \end{aligned}$$

- Rückverfolgen ab maximalem ϵ_{Ti} liefert optimale Zustandsfolge



Rekursives Schema zur Berechnung der optimalen Zustandsfolge

12.4 Parameterschätzung

- das schwierigste Problem beim Einsatz von HMMs ist die Berechnung der optimalen Modellparameter $(\underline{\pi}, \underline{A}, \underline{B})$ anhand einer Beobachtung $\underline{o}$
- es gibt kein analytisches Lösungsverfahren gibt

  angewendet werden iterative Verfahren, die sogenannte EM-Algorithmen sind (expectation-maximization):

expectation	unter der Annahme, daß die Parameter bekannt sind, lassen sich mittels der Algorithmen zur Berechnung der Produktionswahrscheinlichkeit diese für eine gegebene Beobachtungsfolge berechnen
maximization	auf der Grundlage dieser Werte werden die Parameter dann neu geschätzt

  liefert i.A. ein lokales Minimum

Schätzung der HMM-Parameter mittels Baum-Welch-Algorithmus

iteratives Verfahren, so daß gilt: $P(\underline{o} \mid \hat{\lambda}) \geq P(\underline{o} \mid \lambda)$

1. initialisiere zufällig und/oder per Gleichverteilung
2. schätze die Produktionswahrscheinlichkeiten aus einer Beobachtungsfolge $\underline{o}$
   - berechne die Produktionswahrscheinlichkeit und die α- und β-Variablen für $\underline{o}$
   - Sei ξ_{tij} die Wahrscheinlichkeit, zum Zeitpunkt t im Zustand S_i und zum Zeitpunkt $t+1$ im Zustand S_j zu sein (und dabei $\underline{o}$ beobachtet zu haben), so gilt:

$$\xi_{tij} = P(s_t = S_i, s_{t+1} = S_j \mid \underline{o}, \lambda) = \frac{\alpha_{ti} a_{ij} b_j(o_{t+1}) \beta_{t+1,j}}{P(\underline{o} \mid \lambda)}$$

damit ergibt sich die Wahrscheinlichkeit, zum Zeitpunkt t in S_i zu sein:

$$\gamma_{ti} = P(s_t = S_i \mid \underline{o}, \lambda) = \sum_{j=1}^{N} \xi_{tij}$$

3. schätze daraus neue Parameter wie folgt:

$$\hat{\pi}_i = \text{Schätzwert zum Zeitpunkt } t = 1 \text{ im Zustand } S_i \text{ zu sein} = \gamma_{1i} \tag{12.4}$$

$$\hat{a}_{ij} = \frac{\text{Schätzung der Übergänge von } S_i \text{ zu } S_j}{\text{Schätzung in } S_i \text{ zu sein}} = \left(\sum_{t=1}^{T-1} \xi_{tij}\right) / \left(\sum_{t=1}^{T-1} \gamma_{ti}\right) \tag{12.5}$$

$$\hat{b}_{ik} = \frac{\text{Schätzung der Emissionen von } O_k \text{ in } S_i}{\text{Schätzung in } S_i \text{ zu sein}} = \left(\sum_{\forall t: o_t = O_k} \gamma_{ti}\right) / \left(\sum_{t=1}^{T} \gamma_{ti}\right) \tag{12.6}$$

$$\hat{b}_i(\vec{c}) = \mathcal{N}_{\vec{c}}(\underline{\hat{\mu}}_i, \underline{\hat{K}}_i) \text{ , mit} \tag{12.7}$$

$$\underline{\hat{\mu}}_i = \left(\sum_{t=1}^{T} \gamma_{ti} \vec{c}_t\right) / \left(\sum_{t=1}^{T} \gamma_{ti}\right)$$

$$\underline{\hat{K}}_i = \left(\sum_{t=1}^{T} \gamma_{ti} (\vec{c}_t - \underline{\hat{\mu}}_i)(\vec{c}_t - \underline{\hat{\mu}}_i)^T\right) / \left(\sum_{t=1}^{T} \gamma_{ti}\right)$$

iteriere die Schritte 2 (=Expectation) und 3 =Maximization)

Bemerkungen

1. in der Regel liegen viele Beobachtungefolgen $\underline{o}$ vor:
   - berechne die ξ_{tij} und γ_{ti} für die unterschiedlichen Beobachtungen
   - bilde die Mittelwerte über dies Werte
2. analoges Vorgehen mittels Viterbi für $\lambda^* = \operatorname*{argmax}_{\lambda=(\underline{\pi},\underline{A},\underline{B})} P(\underline{o}, \underline{s}^* \mid \lambda)$

   $\Rightarrow$ Viterbi-Training

12.5 HMM als Klassifikator

- Trainiere (Viterbi- oder Baum-Welch-Training) je Klasse ω_k ein eigenes HMM λ^k nur mit Beobachtungsfolgen von Merkmalsvektoren, die zur Klasse ω_k gehören $\rightarrow$ klassifizierte Stichprobe
- Definiere je nach Trainingsart die Unterscheidungsfunktion $\vec{d}(\vec{c})$ zu

$$\vec{d}(\vec{c}) = \begin{pmatrix} p(\underline{o}, \underline{s}^* \mid \lambda^1) \\ \vdots \\ p(\underline{o}, \underline{s}^* \mid \lambda^K) \end{pmatrix}^1 \qquad \text{oder zu} \qquad \vec{d}(\vec{c}) = \begin{pmatrix} p(\underline{o} \mid \lambda^1) \\ \vdots \\ p(\underline{o} \mid \lambda^K) \end{pmatrix}^1$$

- wende folgende Entscheidungsregel an:

$$g(\vec{c}) = \hat{\omega} = e(\vec{d}(\vec{c})) = \omega_l, \quad \text{falls } l \textit{ maximale} \text{ Komponente von } \vec{d}(\vec{c})$$

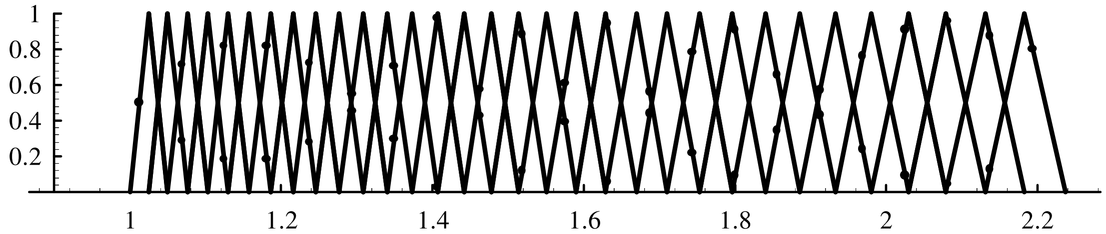
12.6 Beispielanwendung

12.6.1 Spracherkennung

aus dem Sprachsignal gesprochener Sprache

Spracherkennung	eine textuelle Darstellung bestimmen (i.d.R. Wörter erkennen)
Sprachverstehen	eine (interne) Repräsentation der Sprecherintension bestimmen

- **Merkmale**
  - Sprachsignal abtasten (meist: 16Khz)
  - konsekutive Abtastwerte werden als **frame** interpretiert (typisch: 10ms)
  - Fourieranalyse jedes frame
  - Mel-Cepstrum (je Frame)
    Faltung mit 12 Dreiecksfilter angepaßt an menschliches Gehör



    auf die entsehenden Koeffizienten der Cosinustransformation angewandt
    liefert mit zusätzlicher Energie 13 Merkmale
  - zeitliche Veränderung jedes Merkmals aus der Regressionsgeraden über 5 Fenster,
    analog zweite Ableitungen aus Regressionsgeraden der ersten Ableitung

- auf der Basis dieser Merkmale wird je Wort eine HMM trainiert
- diese bilden heute die Basis aller spracherkennenden System

12.6.2 Handgeschriebene Texte

on-line vs. off-line

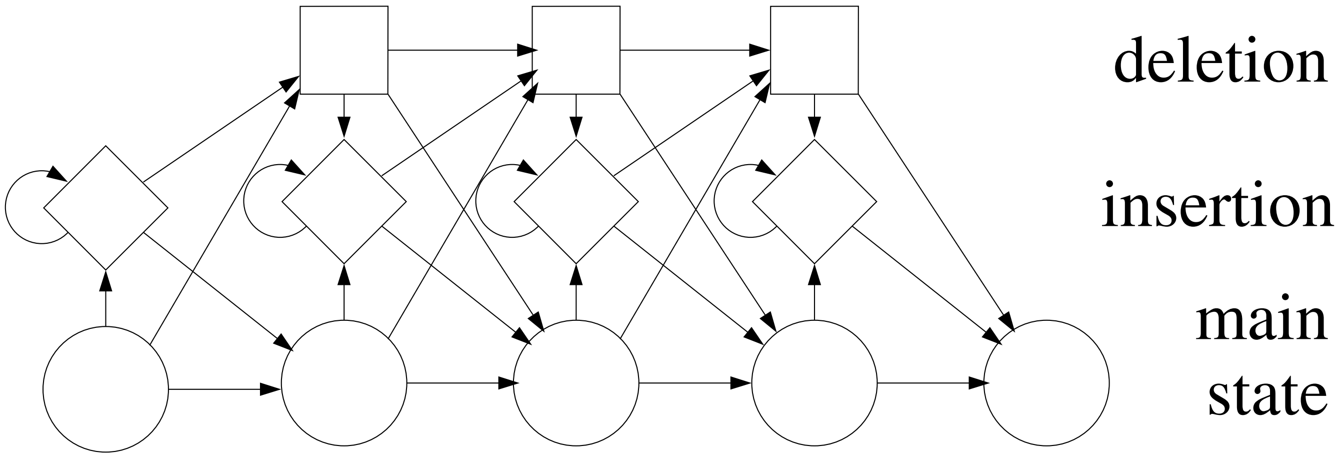
off-line:

- Vorverarbeitung
  - Zeilen und Basislinie detektieren
  - ev. Scherung normieren
- Binarisierung in Fenstern (fester Breite) und heuristische Merkmale detektieren
  - minimale, maximale x- und y-Koordinaten der Schrift
  - Schwerunkt, Masse

12.6.3 Modellierung von motifs

```
ACA---ATG
TCAACTATC
ACAC--AGC
AGA---ATC
ACCG--ATC
```

- multiples Alignment von DNA-Abschnitten, z.B. mit ähnlicher Funktion
- **Ziel:** Auffinden ähnlicher Gene in anderer Sequenz
- Modellierung als HMM:
  - konservierte Bereiche: main states
  - Einfügung, Löschung



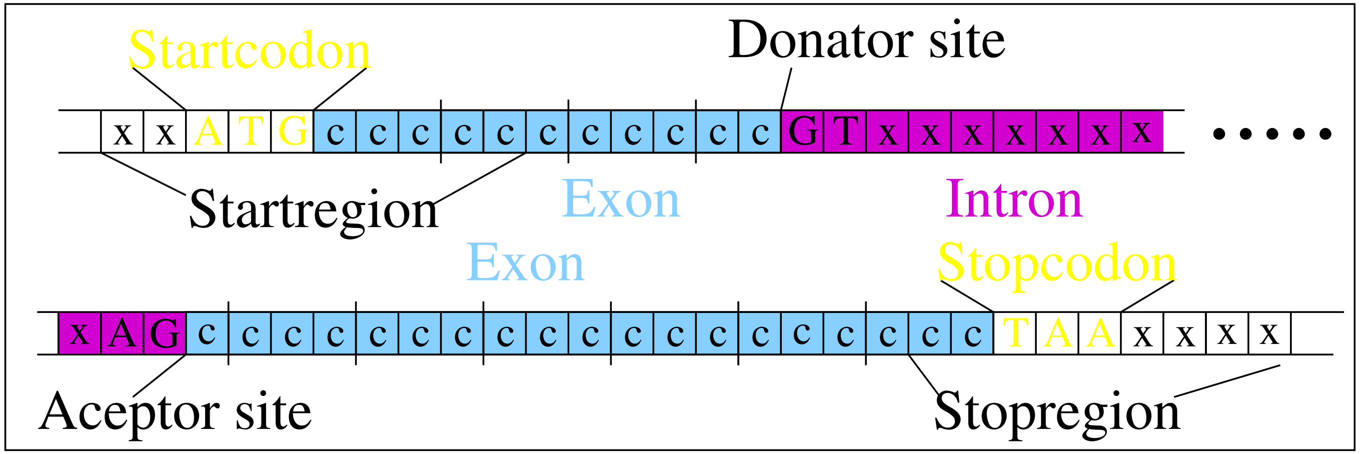
ACA---ATG	4.9
TCAACTATC	3.0
ACAC--AGC	5.3
AGA---ATC	4.9
ACCG--ATC	4.6
TGCT--AGG	–0.1
ACAC--ATC	6.7

- Parameter werden aus den Häufigkeiten der AS oder Base und der Übergänge geschätzt
- zusätzlich Pseudo-Counts, um nicht beobachtete Ereignisse (AS oder Basen, bzw Insertions/Deletions) mit geringer Wahrscheinlichkeit zuzulassen
- rechte Spalte: log odds:

  Wahrscheinlichkeit des besten Weges, normiert durch Null-Modell derselben Länge; das ganze logarithmiert

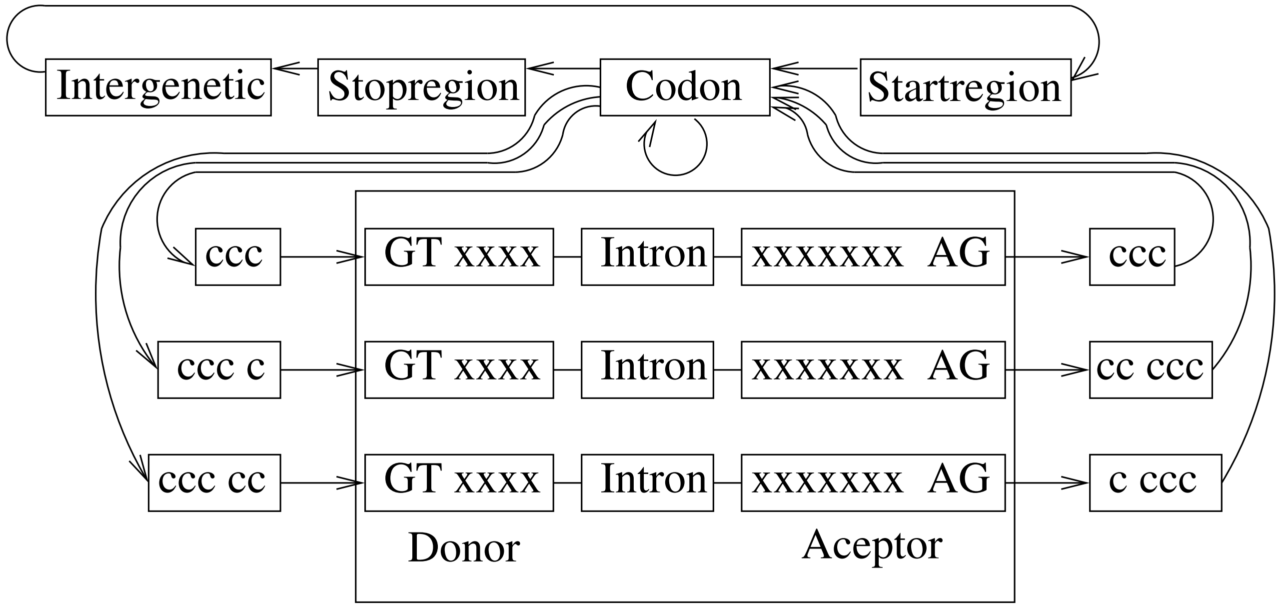
12.6.4 Auffinden von Genen

Modell



HMM

aus Teil-HMMs

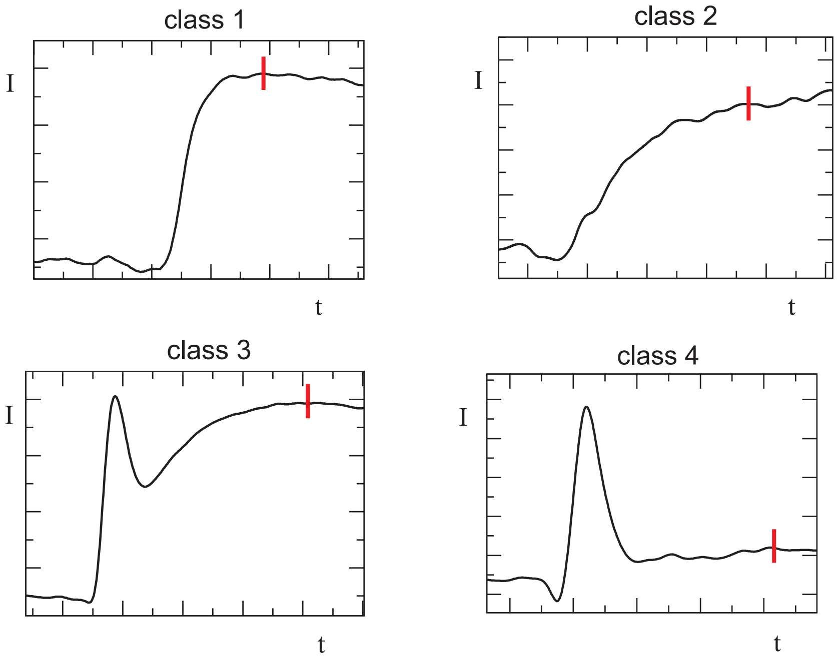


- Modellierung der Teilmodelle aus den Daten wie im Fall der motifs, ggfalls ohne deletion/insertion states
- Abstraktion,
  - **keine** Promotoren
  - keine überlappende Gene
  - **keine** nicht transkribierte 5’ und 3’ Regionen

12.6 Beispielanwendung

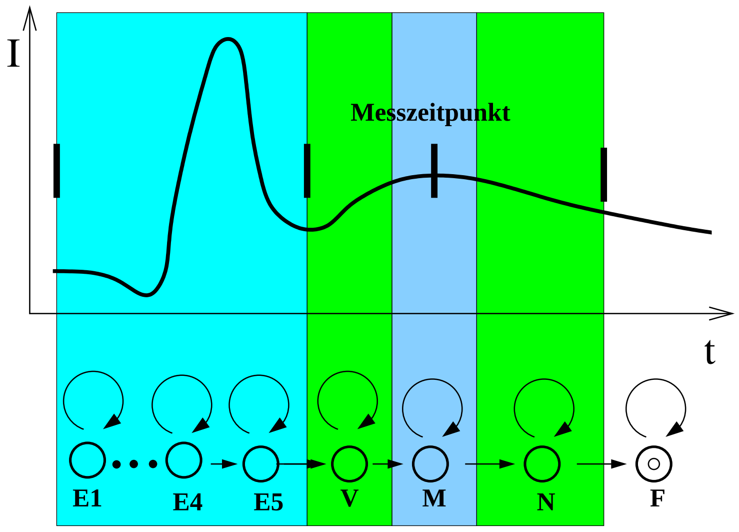
12.6.5 Analyse der Meßkurven amperometrischer Biosensoren

Messgerät

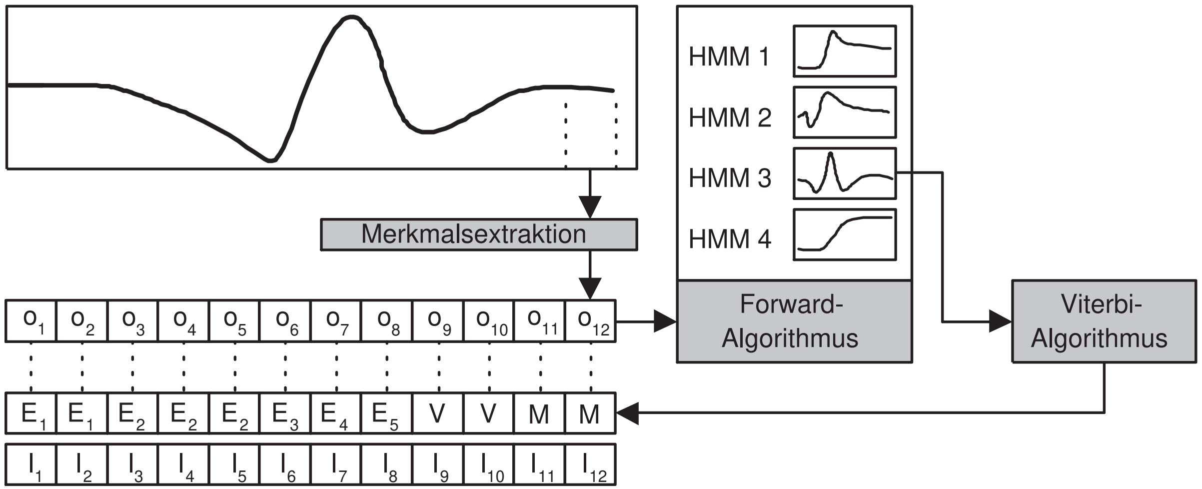


- biologisch sensitive Komponente und Transducer liefern Stromkurven
- gesucht ist die Analyt-Konzentration
- Bestimmung von
  - Messzeitpunkt → Signalstrom
  - Messende
- Signalstrom muss noch umgerechnet werden:

  Grundstrom und Skalierungsfaktor bestimmen
- Schwierigkeiten
  - variable Kurvenformen
  - unterschiedliche Zeitskalen
  - neue, unbekannte Kurvenformen (z.B. bei defekten Sensoren)



- Modellierung von Stromkurven
  - Phasen durch Zustände eines HMM
  - diskrete Symbole: Vektorquantisierung der Ableitungen (SOM)
  - je Kurvenklasse ein HMM
- Training
  - Kurventyp festlegen
  - Markierung der Phasen
  - Initialisierung der Emissionswahrscheinlichkeiten



- Analyse
  - Klassifikation des Kurventyps
  - Zustandsfolge (Viterbi)
  - Messende und -zeitpunkt