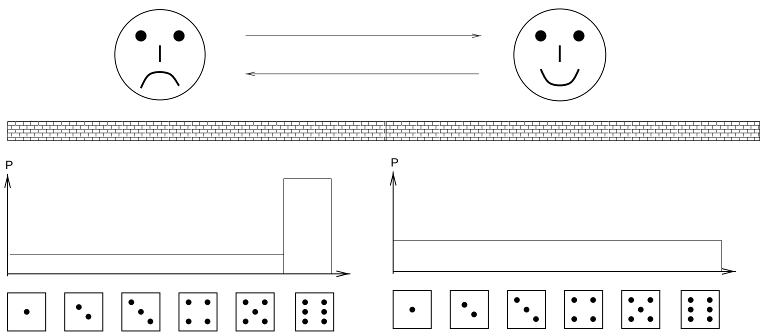
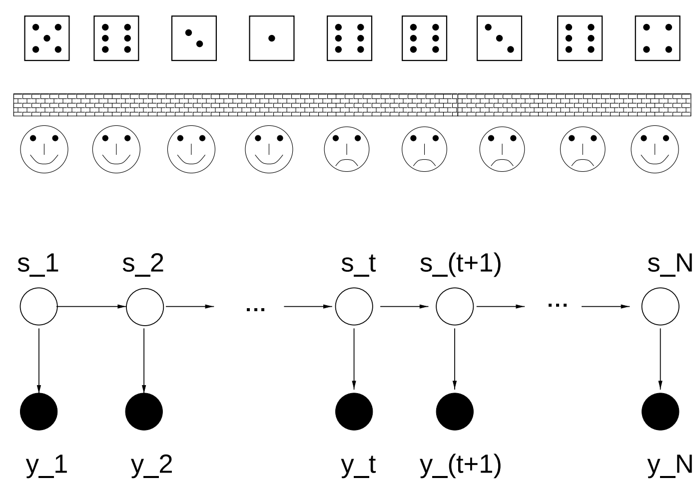
A Tutorial on Hidden Markov Models

Dirk Husmeier
Biomathematics and Statistics Scotland
at the Scottish Crop Research Institute
Invergowrie, Dundee DD2 5DA, UK
Email: dirk@bioss.ac.uk
http://www.bioss.ac.uk/∼dirk

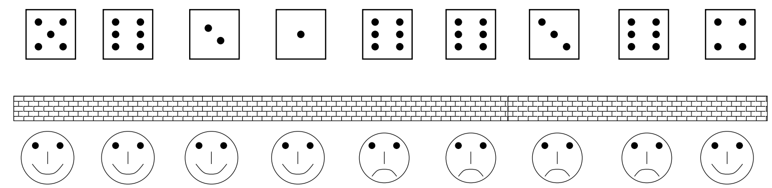
Example: The Occasionally Corrupt Casino



Example: HMM



The Most Likely State Sequence

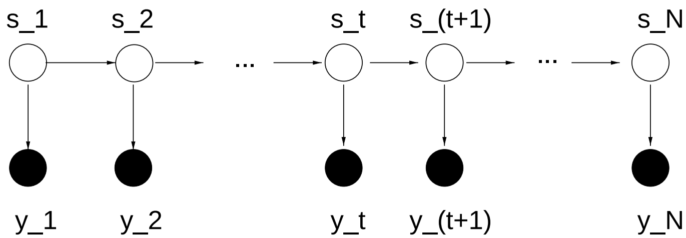


Find the mode of

$$P(S_1, \ldots, S_N | y_1, \ldots, y_N) = \frac{P(S_1, \ldots, S_N, y_1, \ldots, y_N)}{P(y_1, \ldots, y_N)} \propto P(S_1, \ldots, S_N, y_1, \ldots, y_N)$$

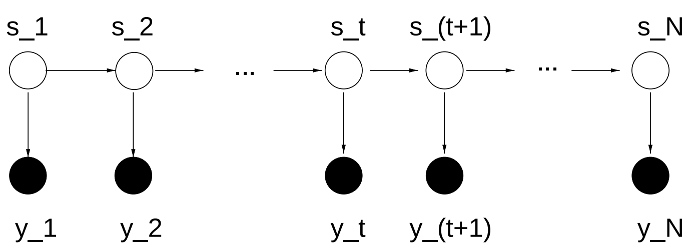
$S_n \in \mathcal{H} \quad \Longrightarrow \quad (S_1, \ldots, S_N) : |\mathcal{H}|^N$ terms.

Factorisation in HMMs



$$
\begin{aligned}
P(y_1,\ldots,y_N,S_1,\ldots,S_N) &= \prod_{t=1}^{N} P(y_t|S_t) \prod_{t=2}^{N} P(S_t|S_{t-1}) P(S_1) \\
&:= \prod_{t=1}^{N} P(y_t|S_t) \prod_{t=1}^{N} P(S_t|S_{t-1})
\end{aligned}
$$

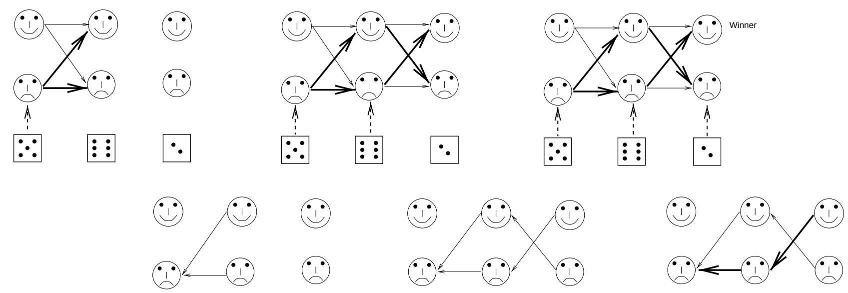
Conditional Independence in HMMs



$$
\begin{aligned}
P(y_t|y_1,\ldots,y_{t-1},y_{t+1},\ldots,y_N,S_1,\ldots,S_N) &= P(y_t|S_t) \\
P(S_{t+1}|S_1,\ldots,S_t,y_1,\ldots,y_t) &= P(S_{t+1}|S_t)
\end{aligned}
$$

Viterbi Algorithm

Initialisation	$\gamma_1(S_1) = P(y_1\|S_1)P(S_1)$ or, with $P(S_1) := P(S_1\|S_0)$: $\gamma_0(S_0) = 1$
Recursion ($n = 1,\ldots,N$)	$\gamma_n(S_n) = P(y_n\|S_n)\max_{S_{n-1}} P(S_n\|S_{n-1})\gamma_{n-1}(S_{n-1})$ $\text{Pointer}(S_{n-1}) = \text{argmax}_{S_{n-1}} P(S_n\|S_{n-1})\gamma_{n-1}(S_{n-1})$
Termination	$\hat{S}_N = \text{argmax}_{S_N} \gamma_N(S_N)$
Traceback ($n = N-1,\ldots,1$)	$\hat{S}_n = \text{Pointer}(\hat{S}_{n+1})$



The Most Likely State Sequence

$$
\begin{aligned}
\max_{S_1,\ldots,S_N} P(S_1,\ldots,S_N|y_1,\ldots,y_N) &= \max_{S_1,\ldots,S_N} P(S_1,\ldots,S_N,y_1,\ldots,y_N) \\
&= \max_{S_N} \gamma_N(S_N)
\end{aligned}
$$

$$
\begin{aligned}
\gamma_n(S_n) &= \max_{S_1,\ldots,S_{n-1}} P(y_1,\ldots,y_n,S_1,\ldots,S_n) \\
&= \max_{S_1,\ldots,S_{n-1}} \prod_{t=1}^{n} P(y_t|S_t)P(S_t|S_{t-1}) \\
&= \max_{S_1,\ldots,S_{n-1}} P(y_n|S_n)P(S_n|S_{n-1}) \prod_{t=1}^{n-1} P(y_t|S_t)P(S_t|S_{t-1}) \\
&= P(y_n|S_n) \max_{S_{n-1}} P(S_n|S_{n-1}) \max_{S_1,\ldots,S_{n-2}} \prod_{t=1}^{n-1} P(y_t|S_t)P(S_t|S_{t-1}) \\
&= P(y_n|S_n) \max_{S_{n-1}} P(S_n|S_{n-1})\gamma_{n-1}(S_{n-1})
\end{aligned}
$$

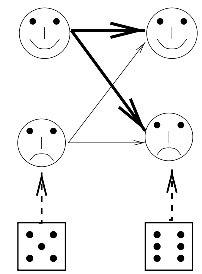
Example - Part 1

n y_n	$P(+\|+)\gamma_{n-1}(+)$ $P(y_n\|+)$	$P(+\|-)\gamma_{n-1}(-)$ $P(y_n\|-)$	$P(-\|+)\gamma_{n-1}(+)$ $\gamma_n(+)$	$P(-\|-)\gamma_{n-1}(-)$ $\gamma_n(-)$
1	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$

n y_n	$P(+\|+)\gamma_{n-1}(+)$ $P(y_n\|+)$	$P(+\|-)\gamma_{n-1}(-)$ $P(y_n\|-)$	$P(-\|+)\gamma_{n-1}(+)$ $\gamma_n(+)$	$P(-\|-)\gamma_{n-1}(-)$ $\gamma_n(-)$
1	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
5	$\frac{1}{6}$	$\frac{1}{10}$	$\frac{1}{2} \times \frac{1}{6} = \frac{1}{12}$	$\frac{1}{2} \times \frac{1}{10} = \frac{1}{20}$

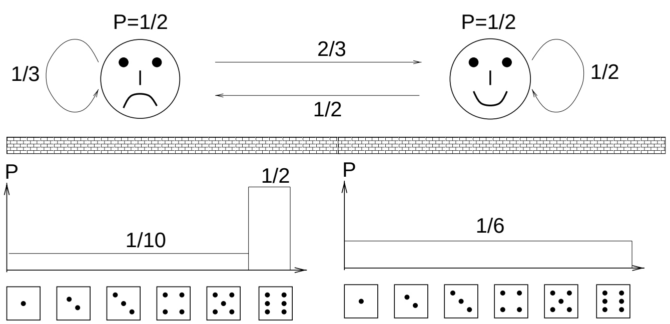
Example - Part 3

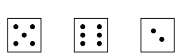
n y_n	$P(+\|+)\gamma_{n-1}(+)$ $P(y_n\|+)$	$P(+\|-)\gamma_{n-1}(-)$ $P(y_n\|-)$	$P(-\|+)\gamma_{n-1}(+)$ $\gamma_n(+)$	$P(-\|-)\gamma_{n-1}(-)$ $\gamma_n(-)$
1	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
5	$\frac{1}{6}$	$\frac{1}{10}$	$\frac{1}{2} \times \frac{1}{6} = \frac{1}{12}$	$\frac{1}{2} \times \frac{1}{10} = \frac{1}{20}$
2	$\mathbf{\frac{1}{2} \times \frac{1}{12} = \frac{1}{24}}$	$\frac{2}{3} \times \frac{1}{20} = \frac{1}{30}$	$\mathbf{\frac{1}{2} \times \frac{1}{12} = \frac{1}{24}}$	$\frac{1}{3} \times \frac{1}{20} = \frac{1}{60}$
6	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{1}{24} \times \frac{1}{6} = \frac{1}{144}$	$\frac{1}{24} \times \frac{1}{2} = \frac{3}{144}$



Computational Complexity and Example

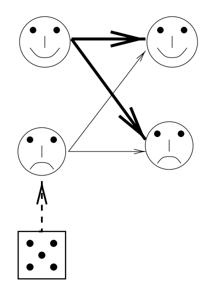
Computation complexity: $|\mathcal{H}|^N \longrightarrow N \times |\mathcal{H}|^2$



Observed sequence: 

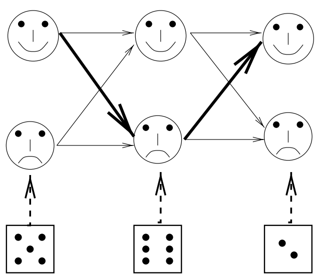
Example - Part 2

n y_n	$P(+\|+)\gamma_{n-1}(+)$ $P(y_n\|+)$	$P(+\|-)\gamma_{n-1}(-)$ $P(y_n\|-)$	$P(-\|+)\gamma_{n-1}(+)$ $\gamma_n(+)$	$P(-\|-)\gamma_{n-1}(-)$ $\gamma_n(-)$
1	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
5	$\frac{1}{6}$	$\frac{1}{10}$	$\frac{1}{2} \times \frac{1}{6} = \frac{1}{12}$	$\frac{1}{2} \times \frac{1}{10} = \frac{1}{20}$
2	$\mathbf{\frac{1}{2} \times \frac{1}{12} = \frac{1}{24}}$	$\frac{2}{3} \times \frac{1}{20} = \frac{1}{30}$	$\mathbf{\frac{1}{2} \times \frac{1}{12} = \frac{1}{24}}$	$\frac{1}{3} \times \frac{1}{20} = \frac{1}{60}$



Example - Part 5

n	$P(+\|+)\gamma_{n-1}(+)$	$P(+\|-)\gamma_{n-1}(-)$	$P(-\|+)\gamma_{n-1}(+)$	$P(-\|-)\gamma_{n-1}(-)$
y_n	$P(y_n\|+)$	$P(y_n\|-)$	$\gamma_n(+)$	$\gamma_n(-)$
1			$\frac{1}{2}$	$\frac{1}{2}$
5	$\frac{1}{6}$	$\frac{1}{10}$	$\frac{1}{2}\times\frac{1}{6}=\frac{1}{12}$	$\frac{1}{2}\times\frac{1}{10}=\frac{1}{20}$
2	$\frac{1}{2}\times\frac{1}{12}=\frac{1}{24}$	$\frac{2}{3}\times\frac{1}{20}=\frac{1}{30}$	$\frac{1}{2}\times\frac{1}{12}=\frac{1}{24}$	$\frac{1}{3}\times\frac{1}{20}=\frac{1}{60}$
6	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{1}{24}\times\frac{1}{6}=\frac{1}{144}$	$\frac{1}{24}\times\frac{1}{2}=\frac{3}{144}$
3	$\frac{1}{2}\times\frac{1}{144}=\frac{1}{288}$	$\mathbf{\frac{2}{3}}\times\mathbf{\frac{3}{144}}=\mathbf{\frac{2}{144}}$	$\frac{1}{2}\times\frac{1}{144}=\frac{1}{288}$	$\mathbf{\frac{1}{3}}\times\mathbf{\frac{3}{144}}=\mathbf{\frac{1}{144}}$
2	$\frac{1}{6}$	$\frac{1}{10}$	$\mathbf{\frac{2}{144}}\times\mathbf{\frac{1}{6}}$	$\frac{1}{144}\times\frac{1}{10}$



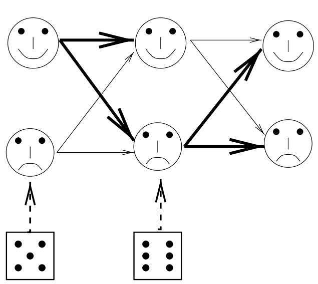
The Posterior Probability of the Viterbi Path

$$P(\hat{S}_1,\dots,\hat{S}_N|y_1,\dots,y_N) = \frac{P(\hat{S}_1,\dots,\hat{S}_N,y_1,\dots,y_N)}{P(y_1,\dots,y_N)}$$
$$P(y_1,\dots,y_N) = \sum_{S_1,\dots,S_N} P(S_1,\dots,S_N,y_1,\dots,y_N)$$

- Allows estimating how good the optimal state sequence is. $\longrightarrow$ Advantage over finite state automata.
- Computational complexity seems to be: $S_n \in \mathcal{H} \implies (S_1,\dots,S_N)$: $|\mathcal{H}|^N$ terms.
- Draw on the Markov property to reduce the computational complexity.

Example - Part 4

n	$P(+\|+)\gamma_{n-1}(+)$	$P(+\|-)\gamma_{n-1}(-)$	$P(-\|+)\gamma_{n-1}(+)$	$P(-\|-)\gamma_{n-1}(-)$
y_n	$P(y_n\|+)$	$P(y_n\|-)$	$\gamma_n(+)$	$\gamma_n(-)$
1			$\frac{1}{2}$	$\frac{1}{2}$
3	$\frac{1}{6}$	$\frac{1}{10}$	$\frac{1}{2}\times\frac{1}{6}=\frac{1}{12}$	$\frac{1}{2}\times\frac{1}{10}=\frac{1}{20}$
2	$\frac{1}{2}\times\frac{1}{12}=\frac{1}{24}$	$\frac{2}{3}\times\frac{1}{20}=\frac{1}{30}$	$\frac{1}{2}\times\frac{1}{12}=\frac{1}{24}$	$\frac{1}{3}\times\frac{1}{20}=\frac{1}{60}$
6	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{1}{24}\times\frac{1}{6}=\frac{1}{144}$	$\frac{1}{24}\times\frac{1}{2}=\frac{3}{144}$
3	$\frac{1}{2}\times\frac{1}{144}=\frac{1}{288}$	$\mathbf{\frac{2}{3}}\times\mathbf{\frac{3}{144}}=\mathbf{\frac{2}{144}}$	$\frac{1}{2}\times\frac{1}{144}=\frac{1}{288}$	$\mathbf{\frac{1}{3}}\times\mathbf{\frac{3}{144}}=\mathbf{\frac{1}{144}}$



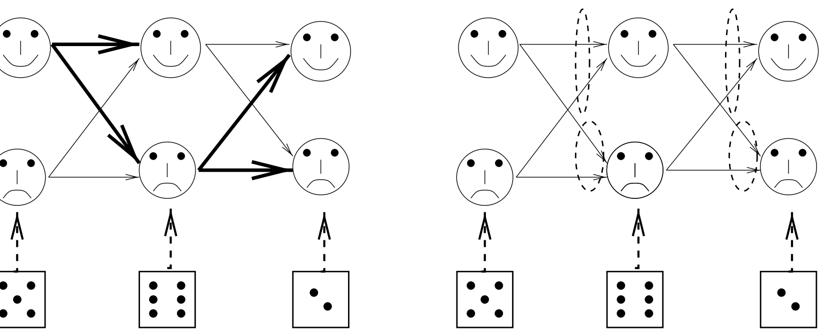
Logarithmic Version of the Viterbi Algorithm

Prevent numerical underflow

$$\max_{S_1,\dots,S_N} P(S_1,\dots,S_N|y_1,\dots,y_N) = \max_{S_1,\dots,S_N} \log P(S_1,\dots,S_N,y_1,\dots,y_N)$$
$$= \max_{S_N} \log \gamma_N(S_N)$$

$$\gamma_n(S_n) = \max_{S_1,\dots,S_{n-1}} P(y_1,\dots,y_n,S_1,\dots,S_n)$$
$$= \max_{S_1,\dots,S_{n-1}} \prod_{t=1}^{n} P(y_t|S_t)P(S_t|S_{t-1})$$
$$= P(y_n|S_n)\max_{S_{n-1}} P(S_n|S_{n-1})\gamma_{n-1}(S_{n-1})$$
$$\log\gamma_n(S_n) = \max_{S_1,\dots,S_{n-1}} \sum_{t=1}^{n}\Big[\log P(y_t|S_t)+\log P(S_t|S_{t-1})\Big]$$
$$= \log P(y_n|S_n)+\max_{S_{n-1}}[\log P(S_n|S_{n-1})+\log\gamma_{n-1}(S_{n-1})]$$

Comparison between the Viterbi and the Forward Algorithms



Viterbi Algorithm: Maximisation

Forward Algorithm: Summation

The Backward Algorithm

$$\begin{aligned}
\beta_n(S_n) &= P(y_{n+1},\ldots,y_N|S_n) \\
&= \sum_{S_{n+1}}\ldots\sum_{S_N} P(y_{n+1},\ldots,y_N,S_{n+1},\ldots,S_N|S_n) \\
&= \sum_{S_{n+1}}\ldots\sum_{S_N}\prod_{t=n+1}^{N} P(y_t|S_t)P(S_t|S_{t-1}) \\
&= \sum_{S_{n+1}}\ldots\sum_{S_N} P(y_{n+1}|S_{n+1})P(S_{n+1}|S_n)\prod_{t=n+2}^{N} P(y_t|S_t)P(S_t|S_{t-1}) \\
&= \sum_{S_{n+1}} P(y_{n+1}|S_{n+1})P(S_{n+1}|S_n)\sum_{S_{n+2}}\ldots\sum_{S_N}\prod_{t=n+2}^{N} P(y_t|S_t)P(S_t|S_{t-1}) \\
&= \sum_{S_{n+1}} P(y_{n+1}|S_{n+1})P(S_{n+1}|S_n)\beta_{n+1}(S_{n+1})
\end{aligned}$$

The Forward Algorithm

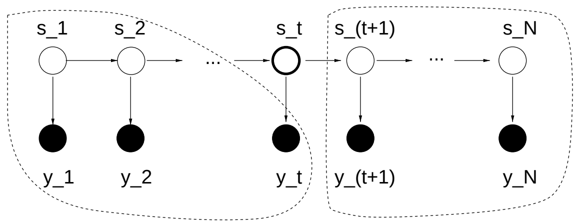
$$\begin{aligned}
P(y_1,\ldots,y_N) &= \sum_{S_N}\alpha_N(S_N) \\
\alpha_n(S_n) &= P(y_1,\ldots,y_n,S_n) \\
&= \sum_{S_1}\ldots\sum_{S_{n-1}} P(y_1,\ldots,y_n,S_1,\ldots,S_{n-1},S_n) \\
&= \sum_{S_1}\ldots\sum_{S_{n-1}}\prod_{t=1}^{n} P(y_t|S_t)P(S_t|S_{t-1}) \\
&= \sum_{S_1}\ldots\sum_{S_{n-1}} P(y_n|S_n)P(S_n|S_{n-1})\prod_{t=1}^{n-1} P(y_t|S_t)P(S_t|S_{t-1}) \\
&= P(y_n|S_n)\sum_{S_{n-1}} P(S_n|S_{n-1})\sum_{S_1}\ldots\sum_{S_{n-2}}\prod_{t=1}^{n-1} P(y_t|S_t)P(S_t|S_{t-1}) \\
&= P(y_n|S_n)\sum_{S_{n-1}} P(S_n|S_{n-1})\alpha_{n-1}(S_{n-1})
\end{aligned}$$

The Forward Algorithm

$$\begin{aligned}
P(y_1,\ldots,y_N) &= \sum_{S_N}\alpha_N(S_N) \\
\alpha_n(S_n) &= P(y_1,\ldots,y_n,S_n) \\
&= \sum_{S_1}\ldots\sum_{S_{n-1}} P(y_1,\ldots,y_n,S_1,\ldots,S_{n-1},S_n) \\
&= \sum_{S_1}\ldots\sum_{S_{n-1}}\prod_{t=1}^{n} P(y_t|S_t)P(S_t|S_{t-1}) \\
&= \sum_{S_1}\ldots\sum_{S_{n-1}} P(y_n|S_n)P(S_n|S_{n-1})\prod_{t=1}^{n-1} P(y_t|S_t)P(S_t|S_{t-1}) \\
&= P(y_n|S_n)\sum_{S_{n-1}} P(S_n|S_{n-1})\sum_{S_1}\ldots\sum_{S_{n-2}}\prod_{t=1}^{n-1} P(y_t|S_t)P(S_t|S_{t-1}) \\
&= P(y_n|S_n)\sum_{S_{n-1}} P(S_n|S_{n-1})\alpha_{n-1}(S_{n-1})
\end{aligned}$$

Posterior Probability of a Hidden State

$$\begin{aligned}
P(S_t|y_1,\ldots,y_N) &\propto P(y_N,\ldots,y_1,S_t)\\
P(y_N,\ldots,y_1,S_t) &= P(y_N,\ldots,y_{t+1}|y_t,\ldots,y_1,S_t)P(y_t,\ldots,y_1,S_t)\\
&= P(y_N,\ldots,y_{t+1}|S_t)P(y_t,\ldots,y_1,S_t)\\
&= \beta_t(S_t)\alpha_t(S_t)\\
P(S_t|y_1,\ldots,y_N) &= \frac{\beta_t(S_t)\alpha_t(S_t)}{\sum_{S'_t}\beta_t(S'_t)\alpha_t(S'_t)}
\end{aligned}$$

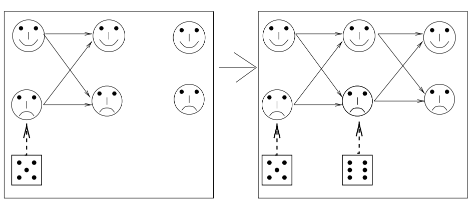


Parameter Optimisation: Baum-Welch Algorithm

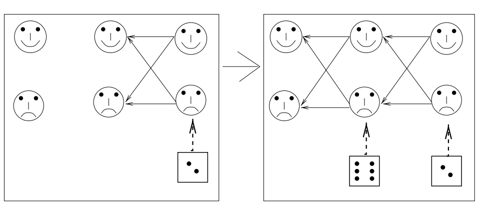
- Hidden states: $S_t \in \{\ldots,h,h',\ldots\}$
- Observations: $y_t \in \{\ldots,o,\ldots\}$
- Forward algorithm $\longrightarrow \alpha_t(S_t),\ P(\mathcal{D})$
- Backward algorithm $\longrightarrow \beta_t(S_t)$
- Update the parameters by computing the expected number of times each transition or emission is used, given the training sequence $\mathcal{D} = \{y_1,\ldots,y_N\}$.

$$\begin{aligned}
P(S_t|\mathcal{D}) &= \frac{\beta_t(S_t)\alpha_t(S_t)}{P(\mathcal{D})}\\
P(S_t,S_{t+1}|\mathcal{D}) &= \frac{\beta_{t+1}(S_{t+1})P(y_{t+1}|S_{t+1})P(S_{t+1}|S_t)\alpha_t(S_t)}{P(\mathcal{D})}\\
P(S_{t+1}=h'|S_t=h) &= \frac{\sum_t P(S_{t+1}=h',S_t=h|\mathcal{D})}{\sum_t P(S_t=h|\mathcal{D})}\\
P(y_t=o|S_t=h) &= \frac{\sum_t \delta(y_t,o)P(S_t=h|\mathcal{D})}{\sum_t P(S_t=h|\mathcal{D})}
\end{aligned}$$

Comparison between the Forward and the Backward Algorithms



Forward Algorithm



Backward Algorithm

Posterior Probability of a Pair of Hidden States

$$\begin{aligned}
P(S_t,S_{t+1}|y_1,\ldots,y_N) &\propto P(y_N,\ldots,y_1,S_{t+1},S_t)\\
P(y_N,\ldots,y_1,S_{t+1},S_t) &= P(y_N,\ldots,y_{t+2}|y_{t+1},\ldots,y_1,S_{t+1},S_t)P(y_{t+1},\ldots,y_1,S_{t+1},S_t)\\
&= P(y_N,\ldots,y_{t+2}|S_{t+1})P(y_{t+1}|y_t,\ldots,y_1,S_{t+1},S_t)\\
&\quad P(S_{t+1}|y_t,\ldots,y_1,S_t)P(y_t,\ldots,y_1,S_t)\\
&= \beta_{t+1}(S_{t+1})P(y_{t+1}|S_{t+1})P(S_{t+1}|S_t)\alpha_t(S_t)
\end{aligned}$$