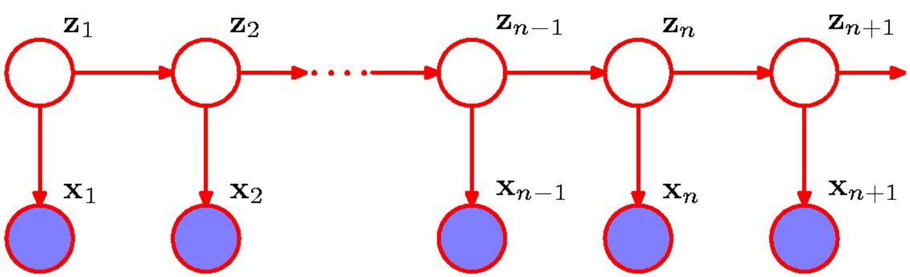
Hidden Markov Models

Algorithms for decoding



Christian Nørgaard Storm Pedersen

cstorm@birc.au.dk

HMM joint probability distribution

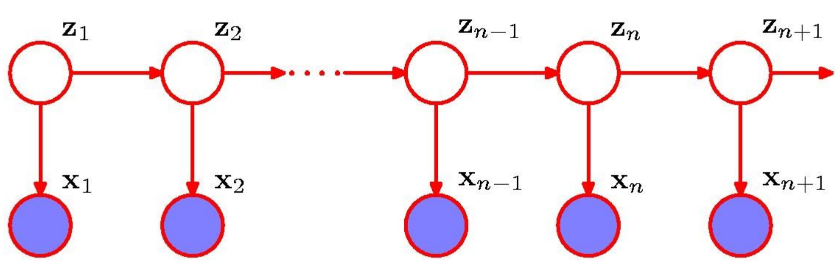
$$p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\Theta}) = p(\mathbf{z}_1|\pi) \left[\prod_{n=2}^{N} p(\mathbf{z}_n|\mathbf{z}_{n-1}, \mathbf{A}) \right] \prod_{n=1}^{N} p(\mathbf{x}_n|\mathbf{z}_n, \phi)$$

Sequence of *N* observables from a set of *D* symbols:

Sequence of *N* hidden states from a set of *K* states:

Model parameters:

$$\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \qquad \mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_N\} \qquad \boldsymbol{\Theta} = \{\pi, \mathbf{A}, \phi\}$$



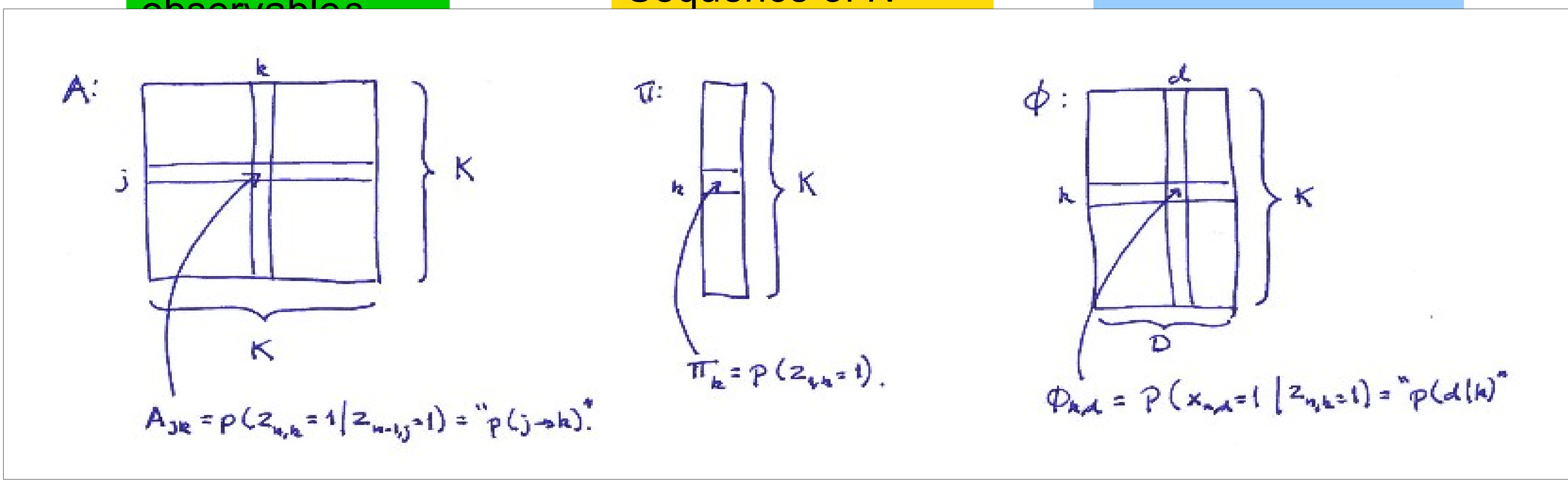
If A and $\boldsymbol{\Phi}$ are the same for all *n* then the HMM is *homogeneous*

HMM joint probability distribution

$$p(\mathbf{X},\mathbf{Z}|\mathbf{\Theta}) = p(\mathbf{z}_1|\pi)\left[\prod_{n=2}^{N} p(\mathbf{z}_n|\mathbf{z}_{n-1},\mathbf{A})\right]\prod_{n=1}^{N} p(\mathbf{x}_n|\mathbf{z}_n,\phi)$$

Sequence of *N*

Sequence of *N*



If A and ***Φ*** are the same for all *n* then the HMM is *homogeneous*

Decoding using HMMs

Given a HMM **Θ** and a sequence of observations $\mathbf{X} = \mathbf{x}_1,\ldots,\mathbf{x}_N$, find a plausible explanation, i.e. a sequence $\mathbf{Z}^* = \mathbf{z}^*_1,\ldots,\mathbf{z}^*_N$ of values of the hidden variable.

Viterbi decoding

$\mathbf{Z}^*$ is the overall most likely explanation of $\mathbf{X}$:

$$\mathbf{Z}^* = \arg\max_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z} | \mathbf{\Theta})$$

Posterior decoding

$\mathbf{z}^*_n$ is the most likely state to be in the *n*'th step:

$$\mathbf{z}^*_n = \arg\max_{\mathbf{z}_n} p(\mathbf{z}_n | \mathbf{x}_1, \ldots, \mathbf{x}_N)$$

Viterbi decoding

Given **X**, find **Z*** such that: $\mathbf{Z}^* = \arg\max_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\mathbf{\Theta})$

$$
\begin{aligned}
p(\mathbf{X}, \mathbf{Z}^*) &= \max_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}) = \max_{\mathbf{z}_1,\ldots,\mathbf{z}_N} p(\mathbf{x}_1, \ldots, \mathbf{x}_N, \mathbf{z}_1, \ldots, \mathbf{z}_N) \\
&= \max_{\mathbf{z}_N} \max_{\mathbf{z}_1,\ldots,\mathbf{z}_{N-1}} p(\mathbf{x}_1, \ldots, \mathbf{x}_N, \mathbf{z}_1, \ldots, \mathbf{z}_N) \\
&= \max_{\mathbf{z}_N} \omega(\mathbf{z}_N)
\end{aligned}
$$

$$
\mathbf{z}_N^* = \arg\max_{\mathbf{z}_N} \omega(\mathbf{z}_N)
$$

Where $\omega(\mathbf{z}_n) \equiv \max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{z}_1, \ldots, \mathbf{z}_n)$ is the probability of the most likely sequence of states $\mathbf{z}_1,\ldots,\mathbf{z}_n$ ending in $\mathbf{z}_n$ generating the observations $\mathbf{x}_1,\ldots,\mathbf{x}_n$

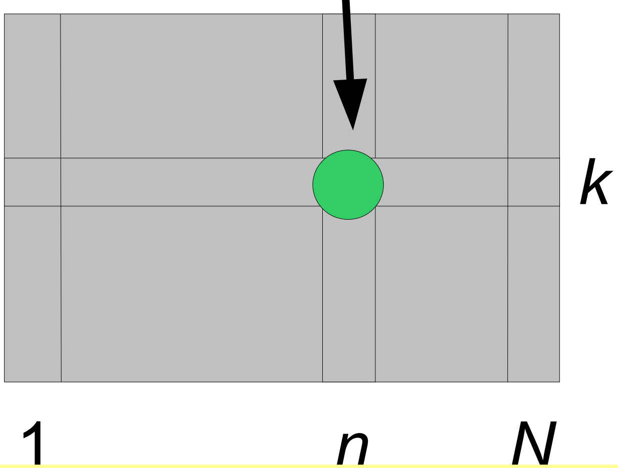
Viterbi decoding

Given **X**, find **Z*** such that: $\mathbf{Z}^* = \arg\max_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\mathbf{\Theta})$

$\omega[k][n] = \omega(\mathbf{z}_n)$ if $\mathbf{z}_n$ is state k

$$
\begin{aligned}
p(\mathbf{X}, \mathbf{Z}^*) &= \max_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}) = \max_{\mathbf{z}_1, \ldots, \mathbf{z}_N} \\
&= \max_{\mathbf{z}_N} \max_{\mathbf{z}_1, \ldots, \mathbf{z}_{N-1}} p(\mathbf{x}_1, \ldots, \mathbf{x}_N \\
&= \max_{\mathbf{z}_N} \omega(\mathbf{z}_N)
\end{aligned}
$$

$$
\mathbf{z}_N^* = \arg\max_{\mathbf{z}_N} \omega(\mathbf{z}_N)
$$



Where $\omega(\mathbf{z}_n) \equiv \max_{\mathbf{z}_1, \ldots, \mathbf{z}_{n-1}} p(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{z}_1, \ldots, \mathbf{z}_n)$ is the probability of the most likely sequence of states $\mathbf{z}_1, \ldots, \mathbf{z}_n$ ending in $\mathbf{z}_n$ generating the observations $\mathbf{x}_1, \ldots, \mathbf{x}_n$

The ω-recursion

$$
\begin{aligned}
\omega(\mathbf{z}_n) &= \max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{x}_1,\ldots,\mathbf{x}_n,\mathbf{z}_1,\ldots,\mathbf{z}_n) \\
&= \max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{z}_1)\prod_{i=2}^{n} p(\mathbf{z}_i|\mathbf{z}_{i-1})\prod_{i=1}^{n} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= p(\mathbf{x}_n|\mathbf{z}_n)\max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{z}_1)\prod_{i=2}^{n} p(\mathbf{z}_i|\mathbf{z}_{i-1})\prod_{i=1}^{n-1} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= p(\mathbf{x}_n|\mathbf{z}_n)\max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{z}_1)p(\mathbf{z}_n|\mathbf{z}_{n-1})\prod_{i=2}^{n-1} p(\mathbf{z}_i|\mathbf{z}_{i-1})\prod_{i=1}^{n-1} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= p(\mathbf{x}_n|\mathbf{z}_n)\max_{\mathbf{z}_{n-1}}\max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-2}} p(\mathbf{z}_1)p(\mathbf{z}_n|\mathbf{z}_{n-1})\prod_{i=2}^{n-1} p(\mathbf{z}_i|\mathbf{z}_{i-1})\prod_{i=1}^{n-1} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= p(\mathbf{x}_n|\mathbf{z}_n)\max_{\mathbf{z}_{n-1}} p(\mathbf{z}_n|\mathbf{z}_{n-1})\max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-2}} p(\mathbf{z}_1)\prod_{i=2}^{n-1} p(\mathbf{z}_i|\mathbf{z}_{i-1})\prod_{i=1}^{n-1} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= p(\mathbf{x}_n|\mathbf{z}_n)\max_{\mathbf{z}_{n-1}} p(\mathbf{z}_n|\mathbf{z}_{n-1})\omega(\mathbf{z}_{n-1})
\end{aligned}
$$

The ω-recursion

$$
\begin{aligned}
\omega(\mathbf{z}_n) &= \max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{x}_1,\ldots,\mathbf{x}_n,\mathbf{z}_1,\ldots,\mathbf{z}_n) \\
&= \max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{z}_1)\prod_{i=2}^{n} p(\mathbf{z}_i|\mathbf{z}_{i-1})\prod_{i=1}^{n} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= p(\mathbf{x}_n|\mathbf{z}_n)\max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{z}_1)\prod_{i=2}^{n} p(\mathbf{z}_i|\mathbf{z}_{i-1})\prod_{i=1}^{n-1} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= p(\mathbf{x}_n|\mathbf{z}_n)\max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{z}_1)p(\mathbf{z}_n|\mathbf{z}_{n-1})\prod_{i=2}^{n-1} p(\mathbf{z}_i|\mathbf{z}_{i-1})\prod_{i=1}^{n-1} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= p(\mathbf{x}_n|\mathbf{z}_n)\max_{\mathbf{z}_{n-1}}\max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-2}} p(\mathbf{z}_1)p(\mathbf{z}_n|\mathbf{z}_{n-1})\prod_{i=2}^{n-1} p(\mathbf{z}_i|\mathbf{z}_{i-1})\prod_{i=1}^{n-1} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= p(\mathbf{x}_n|\mathbf{z}_n)\max_{\mathbf{z}_{n-1}} p(\mathbf{z}_n|\mathbf{z}_{n-1})\max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-2}} p(\mathbf{z}_1)\prod_{i=2}^{n-1} p(\mathbf{z}_i|\mathbf{z}_{i-1})\prod_{i=1}^{n-1} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= p(\mathbf{x}_n|\mathbf{z}_n)\max_{\mathbf{z}_{n-1}} p(\mathbf{z}_n|\mathbf{z}_{n-1})\omega(\mathbf{z}_{n-1})
\end{aligned}
$$

The *ω*-recursion

$\omega(\mathbf{z}_n)$ is the probability of the most likely sequence of states $\mathbf{z}_1,\ldots,\mathbf{z}_n$ ending in $\mathbf{z}_n$ generating the observations $\mathbf{x}_1,\ldots,\mathbf{x}_n$

$$\omega(\mathbf{z}_n) \equiv \max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{x}_1,\ldots,\mathbf{x}_n,\mathbf{z}_1,\ldots,\mathbf{z}_n)$$

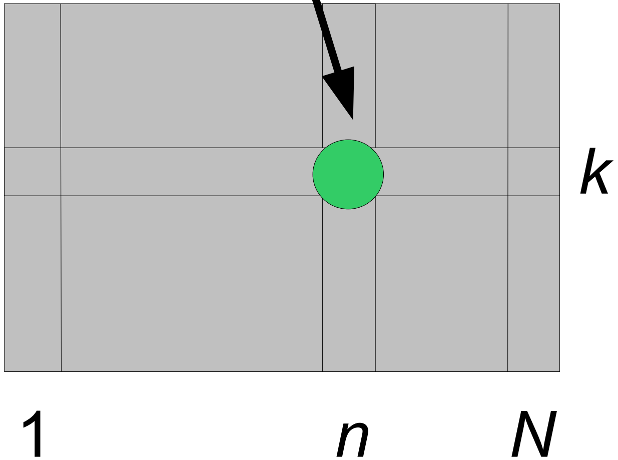
Recursion:

$\omega[k][n] = \omega(\mathbf{z}_n)$ if $\mathbf{z}_n$ is state k

$$\omega(\mathbf{z}_n) = p(\mathbf{x}_n|\mathbf{z}_n) \max_{\mathbf{z}_{n-1}} \omega(\mathbf{z}_{n-1}) p(\mathbf{z}_n|\mathbf{z}_{n-1})$$

Basis:

$$\omega(\mathbf{z}_1) = p(\mathbf{x}_1,\mathbf{z}_1) = p(\mathbf{z}_1)p(\mathbf{x}_1|\mathbf{z}_1)$$



The ω-recursion

```
// Pseudo code for computing ω[k][n] for some n>1
ω[k][n] = 0
for j = 1 to K:
    ω[k][n] = max( ω[k][n], p(x[n] | k) * ω[ j ][n-1] * p( k | j) )
```

of states $\mathbf{z}_1,\ldots,\mathbf{z}_n$

$$\omega(\mathbf{z}_n) \equiv \max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{x}_1,\ldots,\mathbf{x}_n,\mathbf{z}_1,\ldots,\mathbf{z}_n)$$

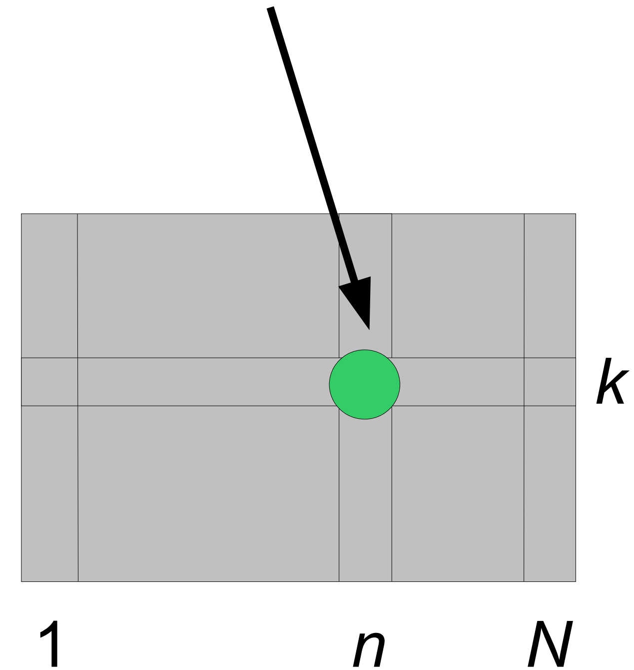
Recursion:

$\omega[k][n] = \omega(\mathbf{z}_n)$ if $\mathbf{z}_n$ is state k

$$\omega(\mathbf{z}_n) = p(\mathbf{x}_n|\mathbf{z}_n)\max_{\mathbf{z}_{n-1}} \omega(\mathbf{z}_{n-1})p(\mathbf{z}_n|\mathbf{z}_{n-1})$$

Basis:

$$\omega(\mathbf{z}_1) = p(\mathbf{x}_1,\mathbf{z}_1) = p(\mathbf{z}_1)p(\mathbf{x}_1|\mathbf{z}_1)$$



The *ω*-recursion

```
// Pseudo code for computing ω[k][n] for some n>1
ω[k][n] = 0
for j = 1 to K:
    ω[k][n] = max( ω[k][n], p(x[n] | k) * ω[ j ][n-1] * p( k | j) )
```

of states $\mathbf{z}_1,\ldots,\mathbf{z}_n$

$$\omega(\mathbf{z}_n) \equiv \max_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{x}_1,\ldots,\mathbf{x}_n,\mathbf{z}_1,\ldots,\mathbf{z}_n)$$

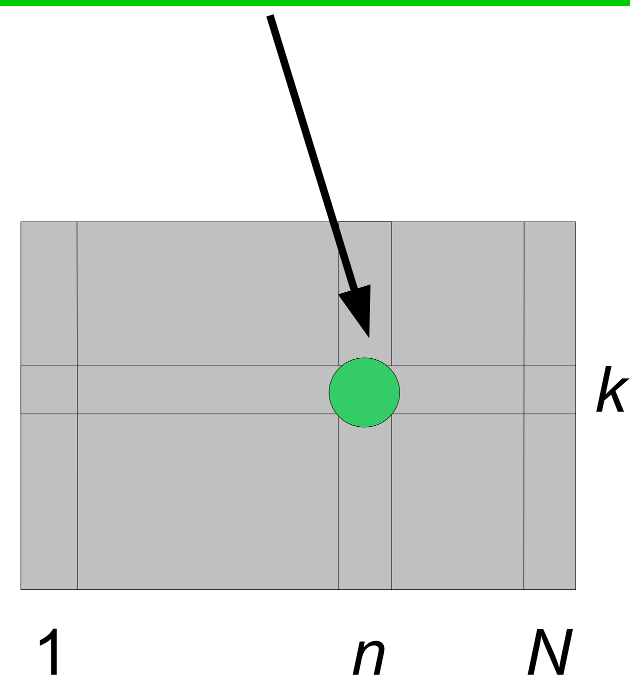
Recursion:

$\omega[k][n] = \omega(\mathbf{z}_n)$ if $\mathbf{z}_n$ is state k

$$\omega(\mathbf{z}_n) = p(\mathbf{x}_n|\mathbf{z}_n) \max_{\mathbf{z}_{n-1}} \omega(\mathbf{z}_{n-1}) p(\mathbf{z}_n|\mathbf{z}_{n-1})$$

Basis:

$$\omega(\mathbf{z}_1) = p(\mathbf{x}_1,\mathbf{z}_1) = p(\mathbf{z}_1)p(\mathbf{x}_1|\mathbf{z}_1)$$



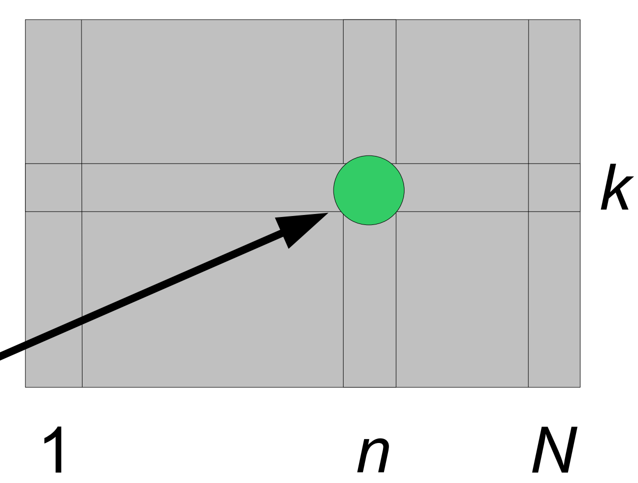
Computing *ω* takes **time O(*K*²*N*)** and **space O(*KN*)** using memorization

Viterbi decoding – Retrieving Z*

$\omega(\mathbf{z}_n)$ is the probability of the most likely sequence of states $\mathbf{z}_1,\ldots,\mathbf{z}_n$ ending in $\mathbf{z}_n$ generating the observations $\mathbf{x}_1,\ldots,\mathbf{x}_n$. We find **Z*** by backtracking:

$$
\begin{aligned}
\mathbf{z}_N^* &= \arg\max_{\mathbf{z}_N} \omega(\mathbf{z}_N) = \arg\max_{\mathbf{z}_N} \max_{\mathbf{z}_{N-1}} \left(p(\mathbf{x}_N|\mathbf{z}_N)\omega(\mathbf{z}_{n-1})p(\mathbf{z}_N|\mathbf{z}_{N-1}) \right) \\
\mathbf{z}_{N-1}^* &= \arg\max_{\mathbf{z}_{N-1}} \left(p(\mathbf{x}_N|\mathbf{z}_N^*)\omega(\mathbf{z}_{N-1})p(\mathbf{z}_N^*|\mathbf{z}_{N-1}) \right) \\
&\vdots
\end{aligned}
$$

$\omega[k][n] = \omega(\mathbf{z}_n)$ if $\mathbf{z}_n$ is state k



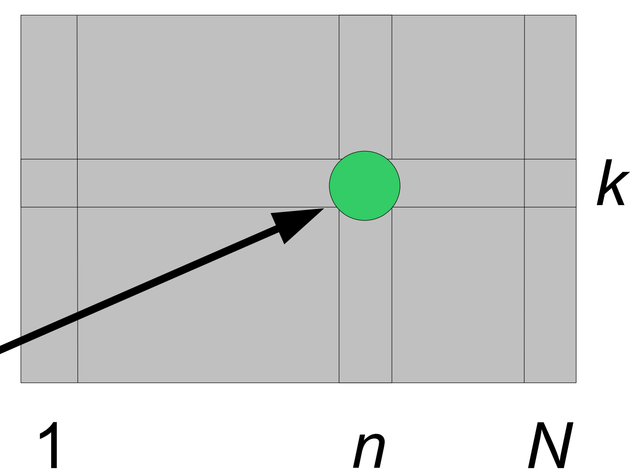
```
// Pseudocode for backtracking
z[1..N] = undef
z[N] = arg max_k ω[k][N]
for n = N-1 to 1:
       z[n] = arg max_k ( p(x[n+1] | z[n+1]) * ω[k][n] * p(z[n+1] | k ) )
print z[1..N]
```

eving Z*

of states $\mathbf{z}_1, \ldots, \mathbf{z}_n$
We find **Z*** by

$$\begin{aligned}
\mathbf{z}_N^* &= \arg\max_{\mathbf{z}_N} \omega(\mathbf{z}_N) = \arg\max_{\mathbf{z}_N} \max_{\mathbf{z}_{N-1}} \left(p(\mathbf{x}_N | \mathbf{z}_N) \omega(\mathbf{z}_{n-1}) p(\mathbf{z}_N | \mathbf{z}_{N-1}) \right) \\
\mathbf{z}_{N-1}^* &= \arg\max_{\mathbf{z}_{N-1}} \left(p(\mathbf{x}_N | \mathbf{z}_N^*) \omega(\mathbf{z}_{N-1}) p(\mathbf{z}_N^* | \mathbf{z}_{N-1}) \right) \\
&\vdots
\end{aligned}$$

$\omega[k][n] = \omega(\mathbf{z}_n)$ if $\mathbf{z}_n$ is state k



eving Z*

of states $\mathbf{z}_1,\ldots,\mathbf{z}_n$

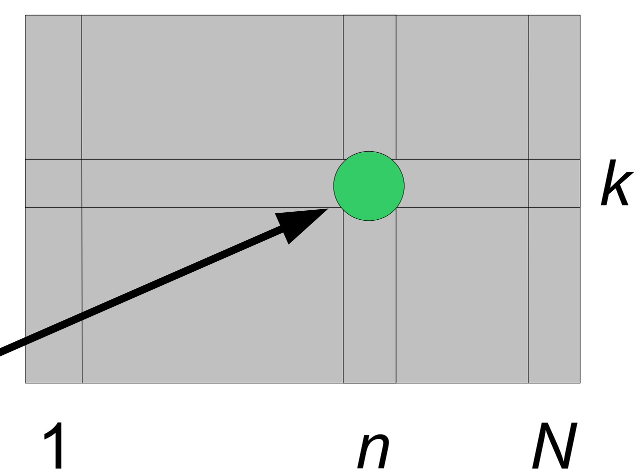
We find **Z*** by

```
// Pseudocode for backtracking
z[1..N] = undef
z[N] = arg max_k ω[k][N]
for n = N-1 to 1:
    z[n] = arg max_k ( p(x[n+1] | z[n+1]) * ω[k][n] * p(z[n+1] | k ) )
print z[1..N]
```

$$
\begin{aligned}
\mathbf{z}_N^* &= \arg\max_{\mathbf{z}_N} \omega(\mathbf{z}_N) = \arg\max_{\mathbf{z}_N} \max_{\mathbf{z}_{N-1}} \big(p(\mathbf{x}_N|\mathbf{z}_N)\omega(\mathbf{z}_{n-1})p(\mathbf{z}_N|\mathbf{z}_{N-1})\big) \\
\mathbf{z}_{N-1}^* &= \arg\max_{\mathbf{z}_{N-1}} \big(p(\mathbf{x}_N|\mathbf{z}_N^*)\omega(\mathbf{z}_{N-1})p(\mathbf{z}_N^*|\mathbf{z}_{N-1})\big) \\
&\vdots
\end{aligned}
$$

Backtracking takes **time O(*KN*)** and **space O(*KN*)** using ω

$\omega[k][n] = \omega(\mathbf{z}_n)$ if $\mathbf{z}_n$ is state k



Decoding using HMMs

Given a HMM **Θ** and a sequence of observations **X** = $\mathbf{x}_1,\ldots,\mathbf{x}_N$, find a plausible explanation, i.e. a sequence **Z*** = $\mathbf{z}^*_1,\ldots,\mathbf{z}^*_N$ of values of the hidden variable.

Viterbi decoding

Z* is the overall most likely explanation of **X**:

$$\mathbf{Z}^* = \arg\max_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z} | \mathbf{\Theta})$$

Posterior decoding

$\mathbf{z}^*_n$ is the most likely state to be in the *n*'th step:

$$\mathbf{z}^*_n = \arg\max_{\mathbf{z}_n} p(\mathbf{z}_n | \mathbf{x}_1, \ldots, \mathbf{x}_N)$$

Posterior decoding

Given **X**, find **Z***, where $\mathbf{z}_n^* = \arg\max_{\mathbf{z}_n} p(\mathbf{z}_n | \mathbf{x}_1, \ldots, \mathbf{x}_N)$

is the most likely state to be in the *n*'th step.

$$
\begin{aligned}
p(\mathbf{z}_n | \mathbf{x}_1, \ldots, \mathbf{x}_N) &= \frac{p(\mathbf{z}_n, \mathbf{x}_1, \ldots, \mathbf{x}_N)}{p(\mathbf{x}_1, \ldots, \mathbf{x}_N)} \\
&= \frac{p(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{z}_n) p(\mathbf{x}_{n+1}, \ldots, \mathbf{x}_N | \mathbf{z}_n, \mathbf{x}_1, \ldots, \mathbf{x}_n)}{p(\mathbf{x}_1, \ldots, \mathbf{x}_N)} \\
&= \frac{p(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{z}_n) p(\mathbf{x}_{n+1}, \ldots, \mathbf{x}_N | \mathbf{z}_n)}{p(\mathbf{x}_1, \ldots, \mathbf{x}_N)} \\
&= \frac{\alpha(\mathbf{z}_n) \beta(\mathbf{z}_n)}{p(\mathbf{X})}
\end{aligned}
$$

$$
\mathbf{z}_n^* = \arg\max_{\mathbf{z}_n} p(\mathbf{z}_n | \mathbf{x}_1, \ldots, \mathbf{x}_N) = \arg\max_{\mathbf{z}_n} \alpha(\mathbf{z}_n) \beta(\mathbf{z}_n) / p(\mathbf{X})
$$

Posterior decoding

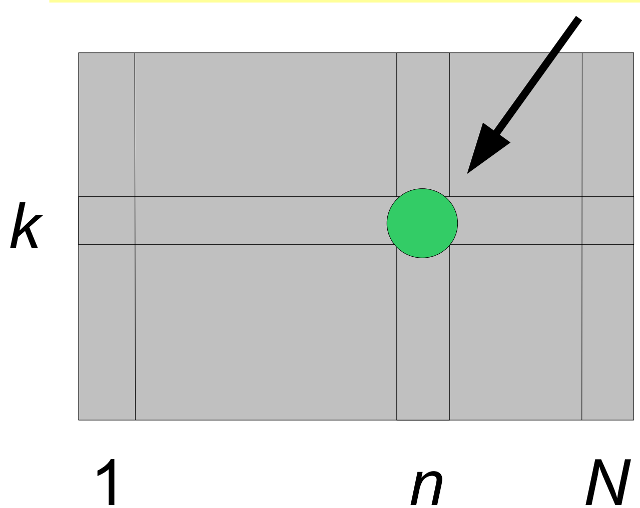
$\alpha(\mathbf{z}_n)$ is the joint probability of observing $\mathbf{x}_1,\ldots,\mathbf{x}_n$ and being in state $\mathbf{z}_n$

$$\alpha(\mathbf{z}_n) \equiv p(\mathbf{x}_1,\ldots,\mathbf{x}_n,\mathbf{z}_n)$$

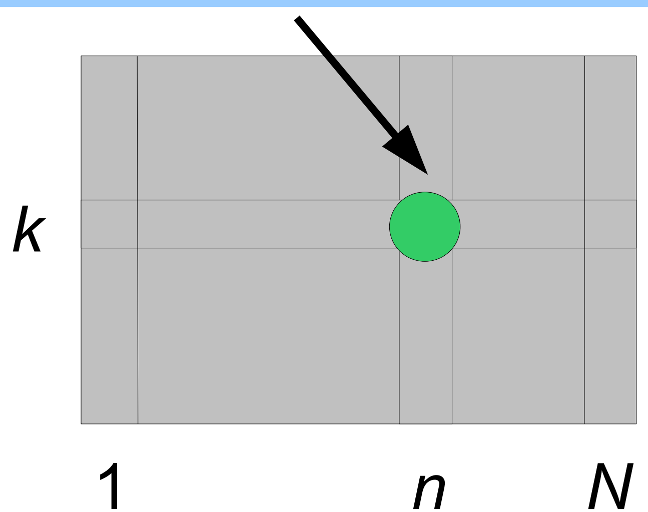
$\beta(\mathbf{z}_n)$ is the conditional probability of future observation $\mathbf{x}_{n+1},\ldots,\mathbf{x}_N$ assuming being in state $\mathbf{z}_n$

$$\beta(\mathbf{z}_n) \equiv p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{z}_n)$$

$\alpha[k][n] = \alpha(\mathbf{z}_n)$ if $\mathbf{z}_n$ is state k



$\beta[k][n] = \beta(\mathbf{z}_n)$ if $\mathbf{z}_n$ is state k



Posterior decoding

$\alpha(\mathbf{z}_n)$ is the joint probability of observing $\mathbf{x}_1,\ldots,\mathbf{x}_n$ and being in state $\mathbf{z}_n$

$$\alpha(\mathbf{z}_n) \equiv p(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{z}_n)$$

$\beta(\mathbf{z}_n)$ is the conditional probability of future observation $\mathbf{x}_{n+1},\ldots,\mathbf{x}_N$ assuming being in state $\mathbf{z}_n$

$$\beta(\mathbf{z}_n) \equiv p(\mathbf{x}_{n+1}, \ldots, \mathbf{x}_N | \mathbf{z}_n)$$

Using $\alpha(\mathbf{z}_n)$ and $\beta(\mathbf{z}_n)$ we get the likelihood of the observations as:

$$p(\mathbf{X}) = \sum_{\mathbf{z}_n} \alpha(\mathbf{z}_n)\beta(\mathbf{z}_n) \qquad p(\mathbf{X}) = \sum_{\mathbf{z}_N} \alpha(\mathbf{z}_N)$$

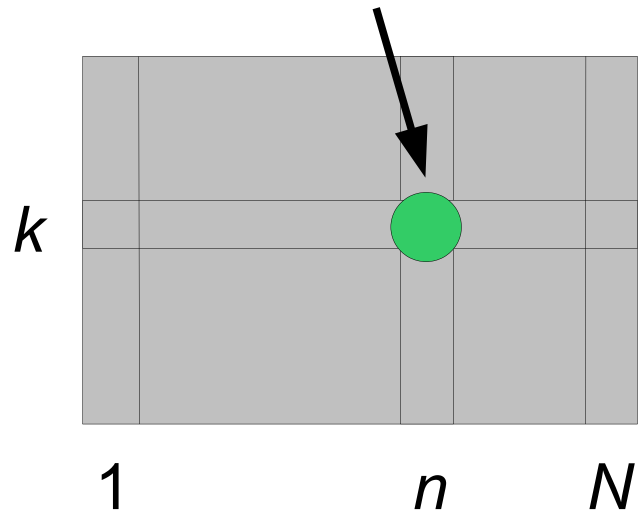
$$\mathbf{z}_n^* = \arg\max_{\mathbf{z}_n} p(\mathbf{z}_n | \mathbf{x}_1, \ldots, \mathbf{x}_N) = \arg\max_{\mathbf{z}_n} \alpha(\mathbf{z}_n)\beta(\mathbf{z}_n)/p(\mathbf{X})$$

The forward algorithm

$\alpha(\mathbf{z}_n)$ is the joint probability of observing $\mathbf{x}_1,\ldots,\mathbf{x}_n$ and being in state $\mathbf{z}_n$

$$\alpha(\mathbf{z}_n) \equiv p(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{z}_n)$$

$\alpha[k][n] = \alpha(\boldsymbol{z}_n)$ if $\boldsymbol{z}_n$ is state k



The α-recursion

$$
\begin{aligned}
\alpha(\mathbf{z}_n) &= p(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{z}_n) \\
&= \sum_{\mathbf{z}_1, \ldots, \mathbf{z}_{n-1}} p(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{z}_1, \ldots, \mathbf{z}_n) \\
&= \sum_{\mathbf{z}_1, \ldots, \mathbf{z}_{n-1}} p(\mathbf{z}_1) \prod_{i=2}^{n} p(\mathbf{z}_i | \mathbf{z}_{i-1}) \prod_{i=1}^{n} p(\mathbf{x}_i | \mathbf{z}_i) \\
&= p(\mathbf{x}_n | \mathbf{z}_n) \sum_{\mathbf{z}_1, \ldots, \mathbf{z}_{n-1}} p(\mathbf{z}_1) \prod_{i=2}^{n} p(\mathbf{z}_i | \mathbf{z}_{i-1}) \prod_{i=1}^{n-1} p(\mathbf{x}_i | \mathbf{z}_i) \\
&= p(\mathbf{x}_n | \mathbf{z}_n) \sum_{\mathbf{z}_1, \ldots, \mathbf{z}_{n-1}} p(\mathbf{z}_1) p(\mathbf{z}_n | \mathbf{z}_{n-1}) \prod_{i=2}^{n-1} p(\mathbf{z}_i | \mathbf{z}_{i-1}) \prod_{i=1}^{n-1} p(\mathbf{x}_i | \mathbf{z}_i) \\
&= p(\mathbf{x}_n | \mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} \sum_{\mathbf{z}_1, \ldots, \mathbf{z}_{n-2}} p(\mathbf{z}_1) p(\mathbf{z}_n | \mathbf{z}_{n-1}) \prod_{i=2}^{n-1} p(\mathbf{z}_i | \mathbf{z}_{i-1}) \prod_{i=1}^{n-1} p(\mathbf{x}_i | \mathbf{z}_i) \\
&= p(\mathbf{x}_n | \mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} p(\mathbf{z}_n | \mathbf{z}_{n-1}) \sum_{\mathbf{z}_1, \ldots, \mathbf{z}_{n-2}} p(\mathbf{z}_1) \prod_{i=2}^{n-1} p(\mathbf{z}_i | \mathbf{z}_{i-1}) \prod_{i=1}^{n-1} p(\mathbf{x}_i | \mathbf{z}_i) \\
&= p(\mathbf{x}_n | \mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} p(\mathbf{z}_n | \mathbf{z}_{n-1}) \alpha(\mathbf{z}_{n-1})
\end{aligned}
$$

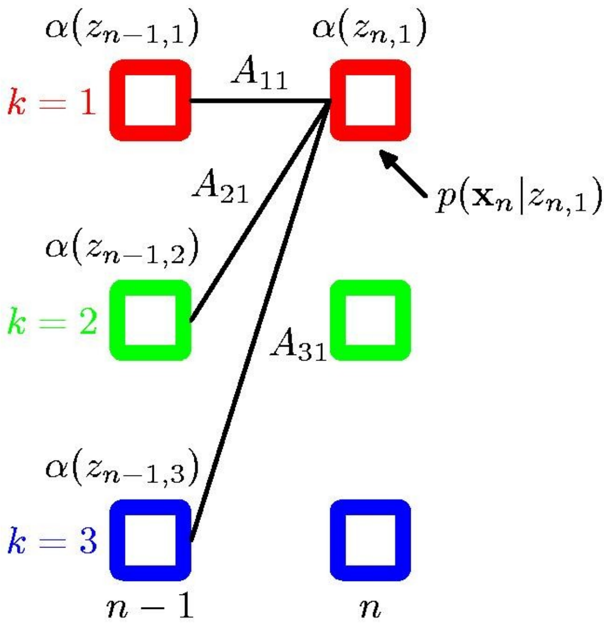
The α-recursion

$$
\begin{aligned}
\alpha(\mathbf{z}_n) &= p(\mathbf{x}_1,\ldots,\mathbf{x}_n,\mathbf{z}_n) \\
&= \sum_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{x}_1,\ldots,\mathbf{x}_n,\mathbf{z}_1,\ldots,\mathbf{z}_n) \\
&= \sum_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{z}_1)\prod_{i=2}^{n} p(\mathbf{z}_i|\mathbf{z}_{i-1})\prod_{i=1}^{n} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= p(\mathbf{x}_n|\mathbf{z}_n)\sum_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{z}_1)\prod_{i=2}^{n} p(\mathbf{z}_i|\mathbf{z}_{i-1})\prod_{i=1}^{n-1} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= p(\mathbf{x}_n|\mathbf{z}_n)\sum_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-1}} p(\mathbf{z}_1)p(\mathbf{z}_n|\mathbf{z}_{n-1})\prod_{i=2}^{n-1} p(\mathbf{z}_i|\mathbf{z}_{i-1})\prod_{i=1}^{n-1} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= p(\mathbf{x}_n|\mathbf{z}_n)\sum_{\mathbf{z}_{n-1}}\sum_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-2}} p(\mathbf{z}_1)p(\mathbf{z}_n|\mathbf{z}_{n-1})\prod_{i=2}^{n-1} p(\mathbf{z}_i|\mathbf{z}_{i-1})\prod_{i=1}^{n-1} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= p(\mathbf{x}_n|\mathbf{z}_n)\sum_{\mathbf{z}_{n-1}} p(\mathbf{z}_n|\mathbf{z}_{n-1})\sum_{\mathbf{z}_1,\ldots,\mathbf{z}_{n-2}} p(\mathbf{z}_1)\prod_{i=2}^{n-1} p(\mathbf{z}_i|\mathbf{z}_{i-1})\prod_{i=1}^{n-1} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= p(\mathbf{x}_n|\mathbf{z}_n)\sum_{\mathbf{z}_{n-1}} p(\mathbf{z}_n|\mathbf{z}_{n-1})\alpha(\mathbf{z}_{n-1})
\end{aligned}
$$

The forward algorithm

$\alpha(\mathbf{z}_n)$ is the joint probability of observing $\mathbf{x}_1,\ldots,\mathbf{x}_n$ and being in state $\mathbf{z}_n$

$$\alpha(\mathbf{z}_n) \equiv p(\mathbf{x}_1,\ldots,\mathbf{x}_n,\mathbf{z}_n)$$

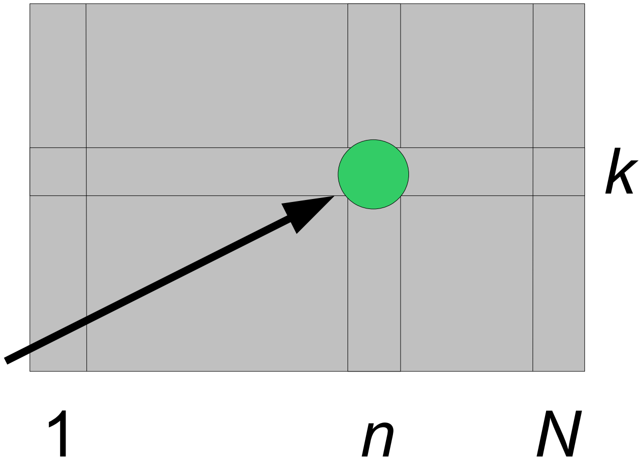


Recursion:

$$\alpha(\mathbf{z}_n) = p(\mathbf{x}_n|\mathbf{z}_n)\sum_{\mathbf{z}_{n-1}}\alpha(\mathbf{z}_{n-1})p(\mathbf{z}_n|\mathbf{z}_{n-1})$$

Basis:

$$\alpha(\mathbf{z}_1) = p(\mathbf{x}_1,\mathbf{z}_1) = p(\mathbf{z}_1)p(\mathbf{x}_1|\mathbf{z}_1)$$



$\alpha[k][n] = \alpha(\mathbf{z}_n)$ if $\mathbf{z}_n$ is state k

The forward algorithm

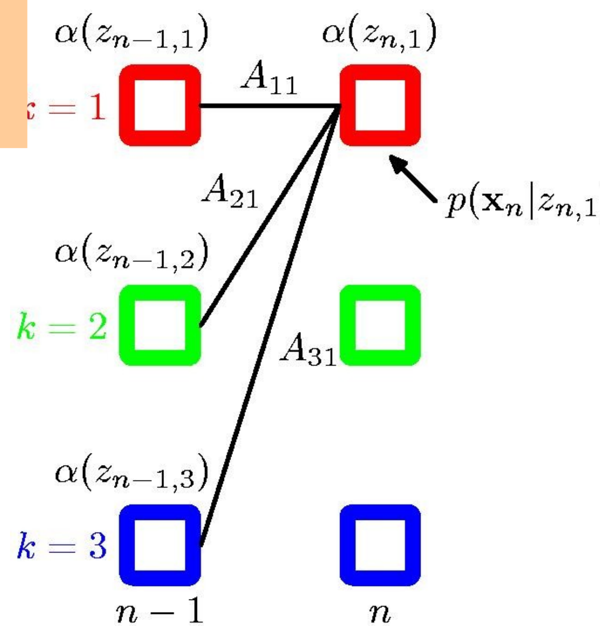
// Pseudo code for computing *α*[*k*][*n*] for some *n*>1

α[*k*][*n*] = 0

for *j* = 1 to *K*:

    α[*k*][*n*] = *α*[*k*][*n*] + *p*(x[*n*] | *k*) * *α*[*j*][*n*-1] * *p*(*k* | *j*)

being in state $\mathbf{z}_n$



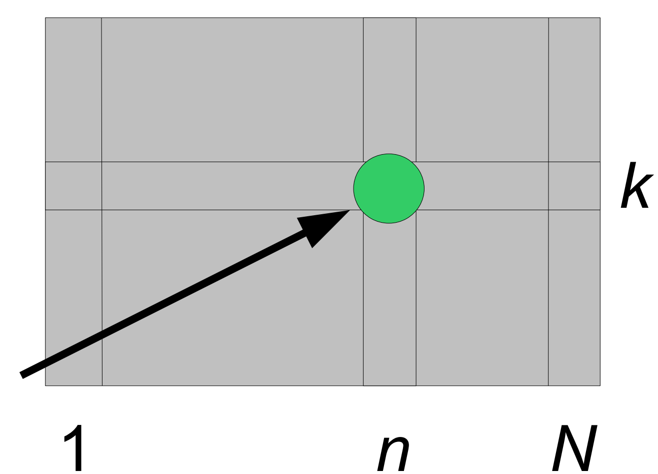
Recursion:

$$\alpha(\mathbf{z}_n) = p(\mathbf{x}_n|\mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} \alpha(\mathbf{z}_{n-1}) p(\mathbf{z}_n|\mathbf{z}_{n-1})$$

Basis:

$$\alpha(\mathbf{z}_1) = p(\mathbf{x}_1, \mathbf{z}_1) = p(\mathbf{z}_1) p(\mathbf{x}_1|\mathbf{z}_1)$$

α[*k*][*n*] = *α*($\mathbf{z}_n$) if $\mathbf{z}_n$ is state *k*



The forward algorithm

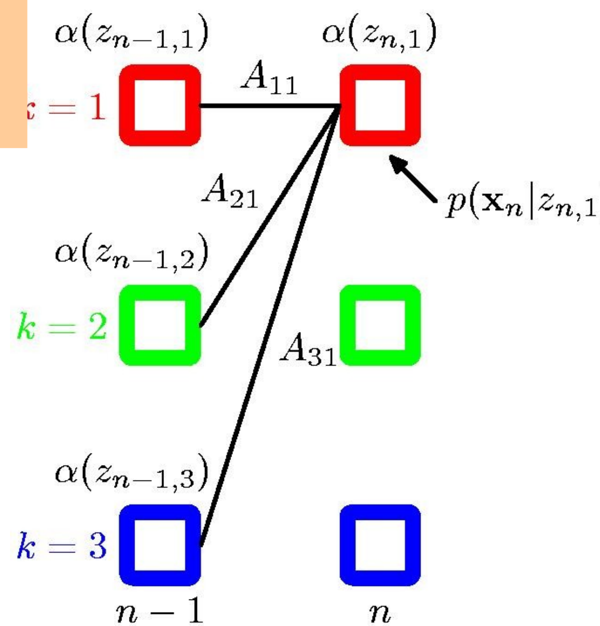
// Pseudo code for computing *α*[*k*][*n*] for some *n*>1

α[*k*][*n*] = 0

for *j* = 1 to *K*:

    α[*k*][*n*] = *α*[*k*][*n*] + *p*(x[*n*] | *k*) * *α*[*j*][*n*-1] * *p*(*k* | *j*)

being in state $\mathbf{z}_n$



Recursion:

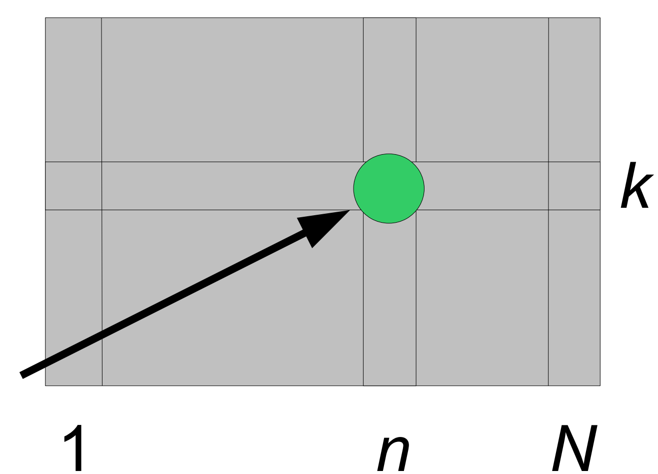
$$\alpha(\mathbf{z}_n) = p(\mathbf{x}_n|\mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} \alpha(\mathbf{z}_{n-1}) p(\mathbf{z}_n|\mathbf{z}_{n-1})$$

Basis:

$$\alpha(\mathbf{z}_1) = p(\mathbf{x}_1, \mathbf{z}_1) = p(\mathbf{z}_1) p(\mathbf{x}_1|\mathbf{z}_1)$$

Computing *α* takes **time O(*K*²*N*)** and **space O(*KN*)** using memorization

tate *k*

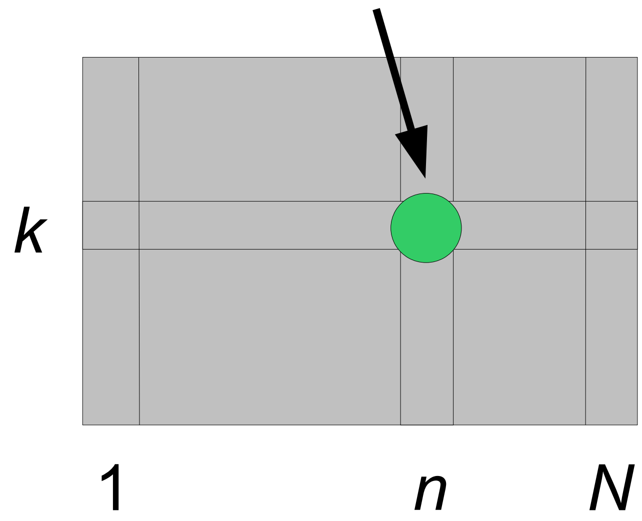


The backward algorithm

$\beta(\mathbf{z}_n)$ is the conditional probability of future observation $\mathbf{x}_{n+1},\ldots,\mathbf{x}_N$ assuming being in state $\mathbf{z}_n$

$$\beta(\mathbf{z}_n) \equiv p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{z}_n)$$

$\beta[k][n] = \beta(\mathbf{z}_n)$ if $\mathbf{z}_n$ is state k



The β-recursion

$$
\begin{aligned}
\beta(\mathbf{z}_n) &= p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{z}_n) \\
&= \sum_{\mathbf{z}_{n+1},\ldots,\mathbf{z}_N} p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N,\mathbf{z}_{n+1},\ldots,\mathbf{z}_N|\mathbf{z}_n) \\
&= \sum_{\mathbf{z}_{n+1},\ldots,\mathbf{z}_N} p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N,\mathbf{z}_n,\mathbf{z}_{n+1},\ldots,\mathbf{z}_N)/p(\mathbf{z}_n) \\
&= \sum_{\mathbf{z}_{n+1},\ldots,\mathbf{z}_N} p(\mathbf{z}_n) \prod_{i=n+1}^{N} p(\mathbf{z}_i|\mathbf{z}_{i-1}) \prod_{i=n+1}^{N} p(\mathbf{x}_i|\mathbf{z}_i)/p(\mathbf{z}_n) \\
&= \sum_{\mathbf{z}_{n+1},\ldots,\mathbf{z}_N} \prod_{i=n+1}^{N} p(\mathbf{z}_i|\mathbf{z}_{i-1}) \prod_{i=n+1}^{N} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= \sum_{\mathbf{z}_{n+1}} \sum_{\mathbf{z}_{n+2},\ldots,\mathbf{z}_N} p(\mathbf{z}_{n+1}|\mathbf{z}_n)p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1}) \prod_{i=n+2}^{N} p(\mathbf{z}_i|\mathbf{z}_{i-1}) \prod_{i=n+2}^{N} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= \sum_{\mathbf{z}_{n+1}} p(\mathbf{z}_{n+1}|\mathbf{z}_n)p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1}) \sum_{\mathbf{z}_{n+2},\ldots,\mathbf{z}_N} \prod_{i=n+2}^{N} p(\mathbf{z}_i|\mathbf{z}_{i-1}) \prod_{i=n+2}^{N} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= \sum_{\mathbf{z}_{n+1}} p(\mathbf{z}_{n+1}|\mathbf{z}_n)p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1})p(\mathbf{x}_{n+2},\ldots,\mathbf{x}_N|\mathbf{z}_{n+1}) \\
&= \sum_{\mathbf{z}_{n+1}} p(\mathbf{z}_{n+1}|\mathbf{z}_n)p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1})\beta(\mathbf{z}_{n+1})
\end{aligned}
$$

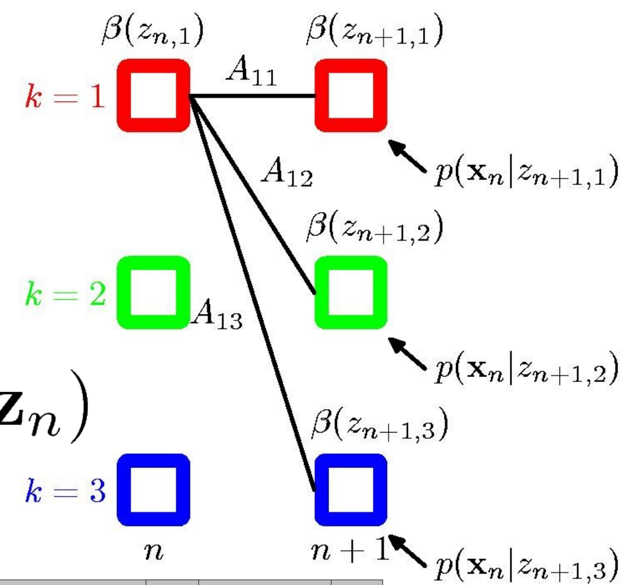
The β-recursion

$$
\begin{aligned}
\beta(\mathbf{z}_n) &= p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{z}_n) \\
&= \sum_{\mathbf{z}_{n+1},\ldots,\mathbf{z}_N} p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N,\mathbf{z}_{n+1},\ldots,\mathbf{z}_N|\mathbf{z}_n) \\
&= \sum_{\mathbf{z}_{n+1},\ldots,\mathbf{z}_N} p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N,\mathbf{z}_n,\mathbf{z}_{n+1},\ldots,\mathbf{z}_N)/p(\mathbf{z}_n) \\
&= \sum_{\mathbf{z}_{n+1},\ldots,\mathbf{z}_N} p(\mathbf{z}_n) \prod_{i=n+1}^{N} p(\mathbf{z}_i|\mathbf{z}_{i-1}) \prod_{i=n+1}^{N} p(\mathbf{x}_i|\mathbf{z}_i)/p(\mathbf{z}_n) \\
&= \sum_{\mathbf{z}_{n+1},\ldots,\mathbf{z}_N} \prod_{i=n+1}^{N} p(\mathbf{z}_i|\mathbf{z}_{i-1}) \prod_{i=n+1}^{N} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= \sum_{\mathbf{z}_{n+1}} \sum_{\mathbf{z}_{n+2},\ldots,\mathbf{z}_N} p(\mathbf{z}_{n+1}|\mathbf{z}_n)p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1}) \prod_{i=n+2}^{N} p(\mathbf{z}_i|\mathbf{z}_{i-1}) \prod_{i=n+2}^{N} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= \sum_{\mathbf{z}_{n+1}} p(\mathbf{z}_{n+1}|\mathbf{z}_n)p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1}) \sum_{\mathbf{z}_{n+2},\ldots,\mathbf{z}_N} \prod_{i=n+2}^{N} p(\mathbf{z}_i|\mathbf{z}_{i-1}) \prod_{i=n+2}^{N} p(\mathbf{x}_i|\mathbf{z}_i) \\
&= \sum_{\mathbf{z}_{n+1}} p(\mathbf{z}_{n+1}|\mathbf{z}_n)p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1})p(\mathbf{x}_{n+2},\ldots,\mathbf{x}_N|\mathbf{z}_{n+1}) \\
&= \sum_{\mathbf{z}_{n+1}} p(\mathbf{z}_{n+1}|\mathbf{z}_n)p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1})\beta(\mathbf{z}_{n+1})
\end{aligned}
$$

The backward algorithm

$\beta(\mathbf{z}_n)$ is the conditional probability of future observation $\mathbf{x}_{n+1},\ldots,\mathbf{x}_N$ assuming being in state $\mathbf{z}_n$

$$\beta(\mathbf{z}_n) \equiv p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{z}_n)$$



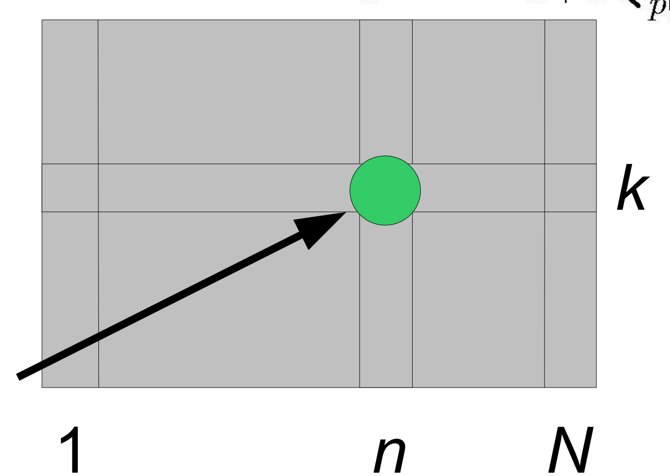
Recursion:

$$\beta(\mathbf{z}_n) = \sum_{\mathbf{z}_{n+1}} \beta(\mathbf{z}_{n+1}) p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1}) p(\mathbf{z}_{n+1}|\mathbf{z}_n)$$

Basis:

$$\beta(\mathbf{z}_N) = 1$$

$\beta[k][n] = \beta(\mathbf{z}_n)$ if $\mathbf{z}_n$ is state k



The backward algorithm

// Pseudo code for computing β[*k*][*n*] for some *n*<*N*

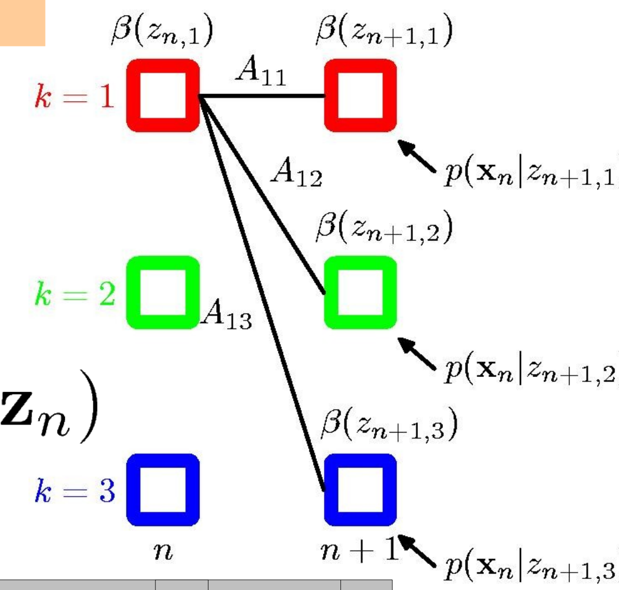
β[*k*][*n*] = 0

for *j* = 1 to *K*:

    β[*k*][*n*] = β[*k*][*n*] + *p*(*j* | *k*) * *p*(x[*n*+1] | *j*) * β[*j*][*n*+1]

ion $\mathbf{x}_{n+1},\ldots,\mathbf{x}_N$

$$\beta(\mathbf{z}_n) \equiv p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{z}_n)$$



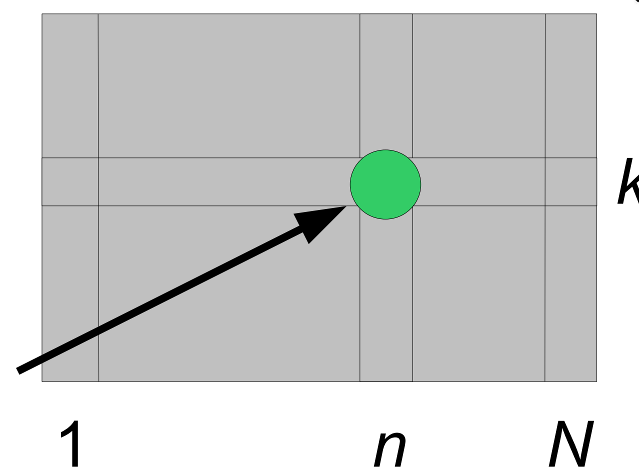
Recursion:

$$\beta(\mathbf{z}_n) = \sum_{\mathbf{z}_{n+1}} \beta(\mathbf{z}_{n+1}) p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1}) p(\mathbf{z}_{n+1}|\mathbf{z}_n)$$

Basis:

$$\beta(\mathbf{z}_N) = 1$$

$\beta[k][n] = \beta(\mathbf{z}_n)$ if $\mathbf{z}_n$ is state k



The backward algorithm

// Pseudo code for computing $\beta[k][n]$ for some $n<N$

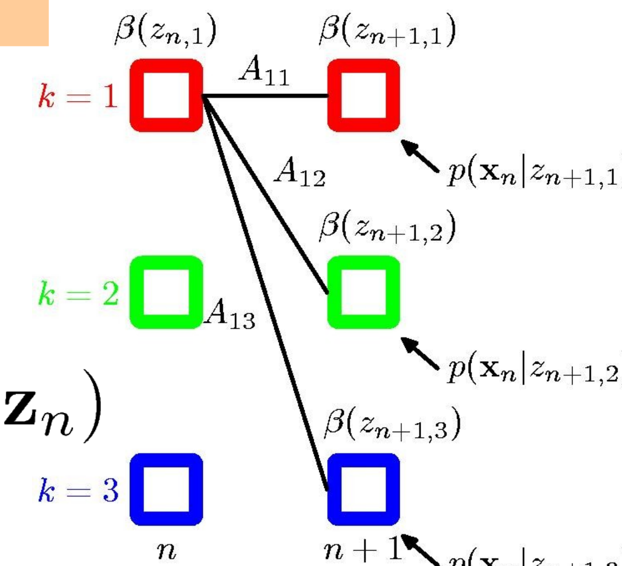
$\beta[k][n] = 0$

for j = 1 to K:

    $\beta[k][n] = \beta[k][n] + p(j \mid k) * p(x[n+1] \mid j) * \beta[j][n+1]$

…tion $\mathbf{x}_{n+1},\ldots,\mathbf{x}_N$

$$\beta(\mathbf{z}_n) \equiv p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{z}_n)$$



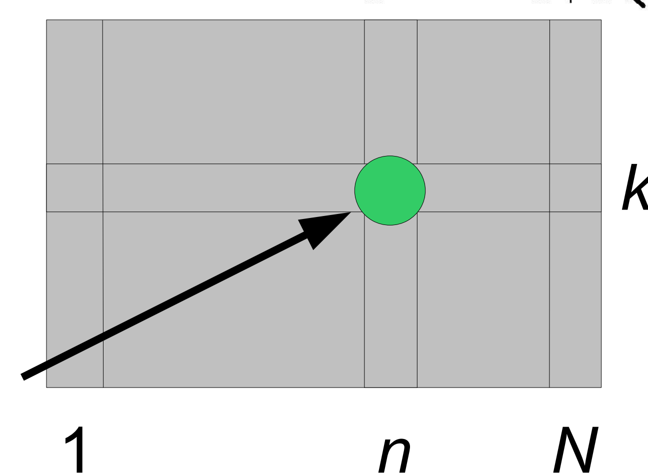
Recursion:

$$\beta(\mathbf{z}_n) = \sum_{\mathbf{z}_{n+1}} \beta(\mathbf{z}_{n+1}) p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1}) p(\mathbf{z}_{n+1}|\mathbf{z}_n)$$

Basis:

$$\beta(\mathbf{z}_N) = 1$$

Computing β takes **time O(K^2N)** and **space O(KN)** using memorization



Posterior decoding

$\alpha(\mathbf{z}_n)$ is the joint probability of observing $\mathbf{x}_1,\ldots,\mathbf{x}_n$ and being in state $\mathbf{z}_n$

$$\alpha(\mathbf{z}_n) \equiv p(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{z}_n)$$

$\beta(\mathbf{z}_n)$ is the conditional probability of future observation $\mathbf{x}_{n+1},\ldots,\mathbf{x}_N$ assuming being in state $\mathbf{z}_n$

$$\beta(\mathbf{z}_n) \equiv p(\mathbf{x}_{n+1}, \ldots, \mathbf{x}_N | \mathbf{z}_n)$$

Using $\alpha(\mathbf{z}_n)$ and $\beta(\mathbf{z}_n)$ we get the likelihood of the observations as:

$$p(\mathbf{X}) = \sum_{\mathbf{z}_n} \alpha(\mathbf{z}_n)\beta(\mathbf{z}_n) \qquad p(\mathbf{X}) = \sum_{\mathbf{z}_N} \alpha(\mathbf{z}_N)$$

$$\mathbf{z}_n^* = \arg\max_{\mathbf{z}_n} p(\mathbf{z}_n | \mathbf{x}_1, \ldots, \mathbf{x}_N) = \arg\max_{\mathbf{z}_n} \alpha(\mathbf{z}_n)\beta(\mathbf{z}_n)/p(\mathbf{X})$$

// Pseudocode for posterior decoding

Compute α[1..K][1..N] and β[1..K][1..N]

pX = α[1][N] + α[2][N] + … + α[K][N]

z[1..N] = undef

for n = 1 to N:
    z[n] = arg max$_k$ (α[k][n] * β[k][n] / pX)

print z[1..N]

…g

…d being in state $\mathbf{z}_n$

…ation $\mathbf{x}_{n+1},\ldots,\mathbf{x}_N$ assuming being in state $\mathbf{z}_n$

$$\beta(\mathbf{z}_n) \equiv p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{z}_n)$$

Using $\alpha(\mathbf{z}_n)$ and $\beta(\mathbf{z}_n)$ we get the likelihood of the observations as:

$$p(\mathbf{X}) = \sum_{\mathbf{z}_n} \alpha(\mathbf{z}_n)\beta(\mathbf{z}_n) \qquad p(\mathbf{X}) = \sum_{\mathbf{z}_N} \alpha(\mathbf{z}_N)$$

$$\mathbf{z}_n^* = \arg\max_{\mathbf{z}_n} p(\mathbf{z}_n|\mathbf{x}_1,\ldots,\mathbf{x}_N) = \arg\max_{\mathbf{z}_n} \alpha(\mathbf{z}_n)\beta(\mathbf{z}_n)/p(\mathbf{X})$$

Viterbi vs. Posterior decoding

A sequence of states $\mathbf{z}_1,\ldots,\mathbf{z}_N$ where $p(\mathbf{x}_1,\ldots,\mathbf{x}_N, \mathbf{z}_1,\ldots,\mathbf{z}_N) > 0$ is a legal (or syntactically correct) decoding of $\mathbf{X}$.

Viterbi finds the most likely syntactically correct decoding of $\mathbf{X}$.

What does Posterior decoding find?

Does it always find a syntactically correct decoding of $\mathbf{X}$?

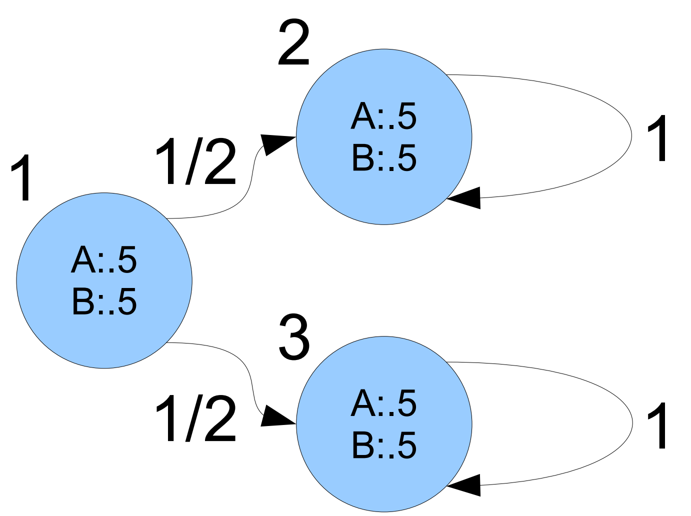
Viterbi vs. Posterior decoding

A sequence of states $\mathbf{z}_1,\ldots,\mathbf{z}_N$ where $p(\mathbf{x}_1,\ldots,\mathbf{x}_N, \mathbf{z}_1,\ldots,\mathbf{z}_N) > 0$ is a legal (or syntactically correct) decoding of **X**.

Viterbi finds the most likely syntactically correct decoding of **X**.

What does Posterior decoding find?

Does it always find a syntactically correct decoding of **X**?



Emits a sequence of A and Bs following either the path 12....2 or 13....3 with equal probability

I.e. Viterbi finds either 12...2 or 13...3, while Posterior finds that 2 and 3 are equally likely for $n>1$.

Recall: Using HMMs

- Determine the likelihood of a sequence of observations.
- Find a plausible underlying explanation (or decoding) of a sequence of observations.

$$p(\mathbf{X}|\mathbf{\Theta}) = \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\mathbf{\Theta}) = \sum_{\mathbf{z}_N} \alpha(\mathbf{z}_N)$$

The sum has K^N terms, but it turns out that it can be computed in $O(K^2N)$ time by computing the α-table using the forward algorithm and summing the last column:

$$p(\mathbf{X}) = \alpha[1][N] + \alpha[2][N] + \ldots + \alpha[K][N]$$

Summary

- **Viterbi-** and **Posterior decoding** for finding a plausible underlying explanation (sequence of hidden states) of a sequence of observations.
- **forward-backward algorithms** for computing the likelihood of being in a given state in the n'th step, and for determining the likelihood of a sequence of observations.

Viterbi

Recursion: $$\omega(\mathbf{z}_n) = p(\mathbf{x}_n|\mathbf{z}_n) \max_{\mathbf{z}_{n-1}} \omega(\mathbf{z}_{n-1}) p(\mathbf{z}_n|\mathbf{z}_{n-1})$$

Basis: $$\omega(\mathbf{z}_1) = p(\mathbf{x}_1, \mathbf{z}_1) = p(\mathbf{z}_1) p(\mathbf{x}_1|\mathbf{z}_1)$$

Forward

Recursion: $$\alpha(\mathbf{z}_n) = p(\mathbf{x}_n|\mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} \alpha(\mathbf{z}_{n-1}) p(\mathbf{z}_n|\mathbf{z}_{n-1})$$

Basis: $$\alpha(\mathbf{z}_1) = p(\mathbf{x}_1, \mathbf{z}_1) = p(\mathbf{z}_1) p(\mathbf{x}_1|\mathbf{z}_1)$$

Backward

Recursion: $$\beta(\mathbf{z}_n) = \sum_{\mathbf{z}_{n+1}} \beta(\mathbf{z}_{n+1}) p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1}) p(\mathbf{z}_{n+1}|\mathbf{z}_n)$$

Basis: $$\beta(\mathbf{z}_N) = 1$$

Problem: The values in the ω-, α-, and β-tables can come very close to zero, by multiplying them we potentially exceed the precision of double precision floating points and get underflow

Next: How to implement the basic algorithms (forward, backward, and Viterbi) in a "numerically" sound manner.

Recursion: $$\alpha(\mathbf{z}_n) = p(\mathbf{x}_n|\mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} \alpha(\mathbf{z}_{n-1}) p(\mathbf{z}_n|\mathbf{z}_{n-1})$$

Basis: $$\alpha(\mathbf{z}_1) = p(\mathbf{x}_1, \mathbf{z}_1) = p(\mathbf{z}_1)p(\mathbf{x}_1|\mathbf{z}_1)$$

Backward

Recursion: $$\beta(\mathbf{z}_n) = \sum_{\mathbf{z}_{n+1}} \beta(\mathbf{z}_{n+1}) p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1}) p(\mathbf{z}_{n+1}|\mathbf{z}_n)$$

Basis: $$\beta(\mathbf{z}_N) = 1$$