Hidden Markov Models
Terminology and Basic Algorithms
The next two weeks

Hidden Markov models (HMMs):

Tuesday 3/11: Terminology and basic algorithms
Thursday 5/11: Implementing the basic algorithms
Tuesday 10/11: Selecting model parameters and training. Introduction to hand-in project.
Thursday 12/11: Extensions and application.

We use Chapter 13 from Bishop's book "Pattern Recognition and Machine Learning". Rabiner's paper "A Tutorial on Hidden Markov Models [...]" might also be useful to read.

Blackboard and http://cs.au.dk/~cstorm/courses/ML_e15
What is machine learning?

Machine learning means different things to different people, and there is no general agreed upon core set of algorithms that must be learned.

For me, the core of machine learning is:

**Building a mathematical model** that captures some desired structure of the data that you are working on.

**Training the model** (i.e. set the parameters of the model) based on existing data to optimize it as well as we can.

**Making predictions** by using the model on new data.
Motivation

We make predictions based on models of observed data (machine learning). A simple model is that observations are assumed to be independent and identically distributed ...

but this assumption is not always the best, fx (1) measurements of weather patterns, (2) daily values of stocks, (3) the composition of DNA or proteins, or ...
Markov Models

If the $n$‘th observation in a chain of observations is influenced only by the $n-1$'th observation, i.e.

$$p(x_n | x_1, \ldots, x_{n-1}) = p(x_n | x_{n-1})$$

then the chain of observations is a 1st-order Markov chain, and the joint-probability of a sequence of $N$ observations is

$$p(x_1, \ldots, x_N) = \prod_{n=1}^{N} p(x_n | x_1, \ldots, x_{n-1}) = p(x_1) \prod_{n=2}^{N} p(x_n | x_{n-1})$$

If the distributions $p(x_n | x_{n-1})$ are the same for all $n$, then the chain of observations is an homogeneous 1st-order Markov chain ...
The model, i.e. $p(x_n | x_{n-1})$:

A sequence of observations:

If the $n$th observation in a chain of observations is influenced only by the $n-1$th observation, i.e.

$$p(x_n | x_1, \ldots, x_{n-1}) = p(x_n | x_{n-1})$$

then the chain of observations is a **1st-order Markov chain**, and the joint-probability of a sequence of $N$ observations is

$$p(x_1, \ldots, x_N) = \prod_{n=1}^{N} p(x_n | x_1, \ldots, x_{n-1}) = p(x_1) \prod_{n=2}^{N} p(x_n | x_{n-1})$$

If the distributions $p(x_n | x_{n-1})$ are the same for all $n$, then the chain of observations is an **homogeneous 1st-order Markov chain**...
What if the $n$'th observation in a chain of observations is influenced by a corresponding hidden variable?

If the hidden variables are discrete and form a Markov chain, then it is a hidden Markov model (HMM).
Hidden Markov Models

What if the $n$'th observation in a chain of observations is influenced by a corresponding hidden variable?

If the hidden variables are discrete and form a Markov chain, then it is a hidden Markov model (HMM)
Hidden Markov Models

What if the \( n \)th observation in a chain of observations is influenced by a corresponding hidden variable?

The joint distribution

\[
p(x_1, \ldots, x_N, z_1, \ldots, z_N) = p(z_1) \prod_{n=2}^{N} p(z_n | z_{n-1}) \prod_{n=1}^{N} p(x_n | z_n)
\]

If the hidden variables are discrete and form a Markov chain, then it is a hidden Markov model (HMM)
What if the $n$'th observation in a chain of observations is influenced by a corresponding hidden variable?

Hidden Markov Models

If the hidden variables are discrete and form a Markov chain, then it is a hidden Markov model (HMM).
Transition probabilities

**Notation:** In Bishop, the hidden variables $z_n$ are positional vectors, e.g. if $z_n = (0,0,1)$ then the model in step $n$ is in state $k=3$ ...

**Transition probabilities:** If the hidden variables are discrete with $K$ states, the conditional distribution $p(z_n | z_{n-1})$ is a $K \times K$ table $A$, and the marginal distribution $p(z_1)$ describing the initial state is a $K$ vector $\pi$ ...

The probability of going from state $j$ to state $k$ is:

$$A_{jk} \equiv p(z_{nk} = 1 | z_{n-1,k} = 1)$$

$$\sum_k A_{jk} = 1$$

The probability of state $k$ being the initial state is:

$$\pi_k \equiv p(z_{1k} = 1)$$

$$\sum_k \pi_k = 1$$
Transition probabilities

If the hidden variables are discrete with $K$ states, the conditional distribution $p(z_n | z_{n-1})$ is a $K \times K$ table $A$, and the marginal distribution $p(z_1)$ describing the initial state is a $K$ vector $\pi$ ...

The probability of going from state $j$ to state $k$ is:

$$A_{jk} \equiv p(z_{nk} = 1 | z_{n-1}, j = 1)$$

$$\sum_k A_{jk} = 1$$

The probability of state $k$ being the initial state is:

$$\pi_k \equiv p(z_{1k} = 1)$$

$$\sum_k \pi_k = 1$$
The transition probabilities:

\[ p(z_n | z_{n-1}, A) = \prod_{k=1}^{K} \prod_{j=1}^{K} A_{jk}^{z_{n-1,j,z_{n,k}}} \]

\[ p(z_1 | \pi) = \prod_{k=1}^{K} \pi_{1k}^{z_{1,k}} \]

The probability of going from state \( j \) to state \( k \) is:

\[ A_{jk} \equiv p(z_{nk} = 1 | z_{n-1,j} = 1) \]

\[ \sum_{k} A_{jk} = 1 \]

The probability of state \( k \) being the initial state is:

\[ \pi_k \equiv p(z_{1k} = 1) \]

\[ \sum_{k} \pi_k = 1 \]
Emission probabilities: The conditional distributions of the observed variables $p(x_n | z_n)$ from a specific state

If the observed values $x_n$ are discrete (e.g. $D$ symbols), the emission probabilities $\Phi$ is a $K \times D$ table of probabilities which for each of the $K$ states specifies the probability of emitting each observable ...

$$p(x_n | z_n, \phi) = \prod_{k=1}^{K} p(x_n | \phi_k)^{z_{nk}}$$
Emission probabilities: The conditional distributions of the observed variables $p(x_n \mid z_n)$ from a specific state $z_n$.

If the observed values $x_n$ are discrete (e.g. $D$ symbols), the emission probabilities $\Phi$ is a $K \times D$ table of probabilities which for each of the $K$ states specifies the probability of emitting each observable symbol.

$$p(x_n \mid z_n, \phi) = \prod_{k=1}^{K} p(x_n \mid \phi_k)^{z_{nk}}$$
HMM joint probability distribution

\[ p(X, Z|\Theta) = p(z_1|\pi) \left[ \prod_{n=2}^{N} p(z_n|z_{n-1}, A) \right] \prod_{n=1}^{N} p(x_n|z_n, \phi) \]

Observables: \( X = \{x_1, \ldots, x_N\} \)

Latent states: \( Z = \{z_1, \ldots, z_N\} \)

Model parameters: \( \Theta = \{\pi, A, \phi\} \)

If \( A \) and \( \phi \) are the same for all \( n \) then the HMM is **homogeneous**
HMM joint probability distribution

\[ p(X, Z | \Theta) = p(z_1 | \pi) \left[ \prod_{n=2}^{N} p(z_n | z_{n-1}, A) \right] \prod_{n=1}^{N} p(x_n | z_n, \phi) \]

If \( A \) and \( \Phi \) are the same for all \( n \) then the HMM is *homogeneous*.
HMMs as a generative model

A HMM *generates a sequence of observables* by moving from latent state to latent state according to the transition probabilities and *emitting an observable* (from a discrete set of observables, i.e. a finite alphabet) from each latent state visited *according to the emission probabilities* of the state ...
Using HMMs

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

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

$$p(X|\Theta) = \sum_Z p(X, Z|\Theta)$$
Using HMMs

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

\[ p(X|\Theta) = \sum_Z p(X, Z|\Theta) \]

The sum has \( K^N \) terms, but it turns out that it can be computed in \( O(K^2N) \) time, but first we will consider **decoding**
Decoding using HMMs

Given a HMM $\Theta$ and a sequence of observations $X = x_1, \ldots, x_N$, find a plausible explanation, i.e. a sequence $Z^* = z^*_1, \ldots, z^*_N$ of values of the hidden variable.
Decoding using HMMs

Given a HMM \( \Theta \) and a sequence of observations \( X = x_1, \ldots, x_N \), find a plausible explanation, i.e. a sequence \( Z^* = z^*_1, \ldots, z^*_N \) of values of the hidden variable.

**Viterbi decoding**

\( Z^* \) is the overall most likely explanation of \( X \): 

\[
Z^* = \arg \max_Z p(X, Z|\Theta)
\]
Decoding using HMMs

Given a HMM \( \Theta \) and a sequence of observations \( X = x_1, \ldots, x_N \), find a plausible explanation, i.e. a sequence \( Z^* = z_1^*, \ldots, z_N^* \) of values of the hidden variable.

**Viterbi decoding**

\( Z^* \) is the overall most likely explanation of \( X \):

\[
Z^* = \arg \max_Z p(X, Z | \Theta)
\]

**Posterior decoding**

\( z_n^* \) is the most likely state to be in the \( n \)'th step:

\[
z_n^* = \arg \max_{z_n} p(z_n | x_1, \ldots, x_N)
\]
Viterbi decoding

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

$$
p(X, Z^*) = \max_Z p(X, Z) = \max_{z_1, \ldots, z_N} p(x_1, \ldots, x_N, z_1, \ldots, z_N)
$$

$$
= \max_{z_N} \max_{z_1, \ldots, z_{N-1}} p(x_1, \ldots, x_N, z_1, \ldots, z_N)
$$

$$
= \max_{z_N} \omega(z_N)
$$

$Z_N^* = \operatorname{arg\ max}_{z_N} \omega(z_N)$

Where $\omega(z_n) \equiv \max_{z_1, \ldots, z_{n-1}} p(x_1, \ldots, x_n, z_1, \ldots, z_n)$ is the probability of the most likely sequence of states $z_1, \ldots, z_n$ ending in $z_n$ generating the observations $x_1, \ldots, x_n$. 
Viterbi decoding

Given $X$, find $Z^*$ such that: $Z^* = \arg\max_Z p(X, Z|\Theta)$

\[
p(X, Z^*) = \max_Z p(X, Z) = \max_{z_1, \ldots, z_N} p(x_1, \ldots, x_N)
\]

\[
= \max_{z_N} \max_{z_1, \ldots, z_{N-1}} p(x_1, \ldots, x_N)
\]

\[
= \max_{z_N} \omega(z_N)
\]

\[
z_N^* = \arg\max_{z_N} \omega(z_N)
\]

Where $\omega(z_n) \equiv \max_{z_1, \ldots, z_{n-1}} p(x_1, \ldots, x_n, z_1, \ldots, z_n)$ is the probability of the most likely sequence of states $z_1, \ldots, z_n$ ending in $z_n$ generating the observations $x_1, \ldots, x_n$. 
The $\omega$-recursion

$$
\omega(z_n) = \max_{z_1, \ldots, z_{n-1}} p(x_1, \ldots, x_n, z_1, \ldots, z_n)
$$

$$
= \max_{z_1, \ldots, z_{n-1}} p(z_1) \prod_{i=2}^{n} p(z_i | z_{i-1}) \prod_{i=1}^{n} p(x_i | z_i)
$$

$$
= p(x_n | z_n) \max_{z_1, \ldots, z_{n-1}} p(z_1) \prod_{i=2}^{n} p(z_i | z_{i-1}) \prod_{i=1}^{n-1} p(x_i | z_i)
$$

$$
= p(x_n | z_n) \max_{z_1, \ldots, z_{n-1}} p(z_1) p(z_n | z_{n-1}) \prod_{i=2}^{n-1} p(z_i | z_{i-1}) \prod_{i=1}^{n-1} p(x_i | z_i)
$$

$$
= p(x_n | z_n) \max_{z_{n-1}} \max_{z_1, \ldots, z_{n-2}} p(z_1) p(z_n | z_{n-1}) \prod_{i=2}^{n-1} p(z_i | z_{i-1}) \prod_{i=1}^{n-1} p(x_i | z_i)
$$

$$
= p(x_n | z_n) \max_{z_{n-1}} p(z_n | z_{n-1}) \max_{z_1, \ldots, z_{n-2}} p(z_1) \prod_{i=2}^{n-1} p(z_i | z_{i-1}) \prod_{i=1}^{n-1} p(x_i | z_i)
$$

$$
= p(x_n | z_n) \max_{z_{n-1}} p(z_n | z_{n-1}) \omega(z_{n-1})
$$
The $\omega$-recursion

$$
\omega(z_n) = \max_{z_1, \ldots, z_{n-1}} p(x_1, \ldots, x_n, z_1, \ldots, z_n)
$$

Using HMM

$$
= \max_{z_1, \ldots, z_{n-1}} p(z_1) \prod_{i=2}^{n} p(z_i | z_{i-1}) \prod_{i=1}^{n} p(x_i | z_i)
$$

$$
= p(x_n | z_n) \max_{z_1, \ldots, z_{n-1}} p(z_1) \prod_{i=2}^{n} p(z_i | z_{i-1}) \prod_{i=1}^{n} p(x_i | z_i)
$$

$$
= p(x_n | z_n) \max_{z_{n-1}, z_1, \ldots, z_{n-2}} p(z_1) p(z_n | z_{n-1}) \prod_{i=2}^{n-1} p(z_i | z_{i-1}) \prod_{i=1}^{n-1} p(x_i | z_i)
$$

$$
= p(x_n | z_n) \max_{z_{n-1}} p(z_n | z_{n-1}) \max_{z_1, \ldots, z_{n-2}} p(z_1) \prod_{i=2}^{n-1} p(z_i | z_{i-1}) \prod_{i=1}^{n-1} p(x_i | z_i)
$$

$$
= p(x_n | z_n) \max_{z_{n-1}} p(z_n | z_{n-1}) \omega(z_{n-1})
$$
The $\omega$-recursion

$\omega(z_n)$ is the probability of the most likely sequence of states $z_1,...,z_n$ ending in $z_n$ generating the observations $x_1,...,x_n$.

$$\omega(z_n) \equiv \max_{z_1,...,z_{n-1}} p(x_1, \ldots, x_n, z_1, \ldots, z_n)$$

Recursion:

$$\omega(z_n) = p(x_n | z_n) \max_{z_{n-1}} \omega(z_{n-1}) p(z_n | z_{n-1})$$

Basis:

$$\omega(z_1) = p(x_1, z_1) = p(z_1) p(x_1 | z_1)$$
The $\omega$-recursion

// Pseudo code for computing $\omega[k][n]$ for some $n>1$

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

if $p(x[n] | k) \neq 0$:  
    for $j = 1$ to $K$:
        if $p(k | j) \neq 0$:
            $\omega[k][n] = \max( \omega[k][n], p(x[n] | k) \cdot \omega[j][n-1] \cdot p(k | j) )$

Recursion:

$\omega(z_n) = p(x_n | z_n) \max_{z_{n-1}} \omega(z_{n-1}) p(z_n | z_{n-1})$

Basis:

$\omega(z_1) = p(x_1, z_1) = p(z_1) p(x_1 | z_1)$
The \( \omega \)-recursion

\[
\omega(k)[n] = 0
\]

if \( p(x[n] \mid k) \neq 0 \):
  
  for \( j = 1 \) to \( K \):
    
    if \( p(k \mid j) \neq 0 \):
      
      \[
      \omega(k)[n] = \max( \omega(k)[n], p(x[n] \mid k) \cdot \omega(j)[n-1] \cdot p(k \mid j) )
      \]

\[ \omega(z_n) = p(x_n \mid z_n) \max_{z_{n-1}} \omega(z_{n-1}) p(z_n \mid z_{n-1}) \]

**Basis:**

\[ \omega(z_1) = p(x_1, z_1) = p(z_1) p(x_1 \mid z_1) \]

Computing \( \omega \) takes time \( O(K^2N) \) and space \( O(KN) \) using memorization.

\[ \omega[k][n] = \omega(z_n) \text{ if } z_n \text{ is state } k \]
\( \omega(z_n) \) is the probability of the most likely sequence of states \( z_1, \ldots, z_n \) ending in \( z_n \) generating the observations \( x_1, \ldots, x_n \). We find \( Z^* \) by backtracking:

\[
\begin{align*}
Z_N^* &= \arg \max_{Z_N} \omega(Z_N) = \arg \max_{Z_N} \max_{Z_{N-1}} \left( p(x_N | Z_N) \omega(Z_{n-1}) p(Z_N | Z_{N-1}) \right) \\
Z_{N-1}^* &= \arg \max_{Z_{N-1}} \left( p(x_N | Z_N^*) \omega(Z_{N-1}) p(Z_N^* | Z_{N-1}) \right) \\
Z_{N-2}^* &= \arg \max_{Z_{N-2}} \left( p(x_{N-1} | Z_{N-1}^*) \omega(Z_{N-2}) p(Z_{N-1}^* | Z_{N-2}) \right) \\
& \vdots
\end{align*}
\]
\( \omega(\mathbf{z}_n) \) 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 \( \mathbf{Z}^* \) by backtracking:

**Viterbi decoding – Retrieving \( \mathbf{Z}^* \)**

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

// Pseudocode for backtracking

\[
z[1..N] = \text{undef}
\]

\[
z[N] = \arg\max_k \omega[k][N]
\]

for \( n = N-1 \) to 1:

\[
z[n] = \arg\max_k \left( p(\mathbf{x}[n+1] | \mathbf{z}[n+1]) \cdot \omega[k][n] \cdot p(\mathbf{z}[n+1] | k) \right)
\]

print \( z[1..N] \)

**Pseudocode for backtracking**

\[
\mathbf{z}^*_N = \arg\max_{\mathbf{z}_N} \omega(\mathbf{z}_N) = \arg\max_{\mathbf{z}_N} \max_{\mathbf{z}_{N-1}} \left( p(\mathbf{x}_N | \mathbf{z}_N) \omega(\mathbf{z}_{n-1}) p(\mathbf{z}_N | \mathbf{z}_{N-1}) \right)
\]

\[
\mathbf{z}^*_{N-1} = \arg\max_{\mathbf{z}_{N-1}} \left( p(\mathbf{x}_N | \mathbf{z}^*_N) \omega(\mathbf{z}_{N-1}) p(\mathbf{z}_N | \mathbf{z}_{N-1}) \right)
\]

\[
\mathbf{z}^*_{N-2} = \arg\max_{\mathbf{z}_{N-2}} \left( p(\mathbf{x}_{N-1} | \mathbf{z}^*_{N-1}) \omega(\mathbf{z}_{N-2}) p(\mathbf{z}^*_{N-1} | \mathbf{z}_{N-2}) \right)
\]

\[\vdots\]

\[
\omega[k][n] = \omega(\mathbf{z}_n) \text{ if } \mathbf{z}_n \text{ is state } k
\]
\[ \omega(z^n) \] is the probability of the most likely sequence of states \( z_1, \ldots, z_n \) ending in \( z_n \) generating the observations \( x_1, \ldots, x_n \). We find \( Z^* \) by backtracking:

\[
\begin{align*}
\text{Pseudocode for backtracking} \\
z[1..N] &= \text{undef} \\
z[N] &= \arg \max_k \omega[k][N] \\
\text{for } n = N-1 \text{ to } 1: \\
& \quad z[n] = \arg \max_k \left( p(x[n+1] \mid z[n+1]) \ast \omega[k][n] \ast p(z[n+1] \mid k) \right) \\
\text{print } z[1..N]
\end{align*}
\]

Backtracking takes time \( O(KN) \) and space \( O(KN) \) using \( \omega \):

\[ \omega[k][n] = \omega(z^n) \text{ if } z_n \text{ is state } k \]
Decoding using HMMs

Given a HMM $\Theta$ and a sequence of observations $X = x_1, \ldots, x_N$, find a plausible explanation, i.e. a sequence $Z^* = z^*_1, \ldots, z^*_N$ of values of the hidden variable.

**Viterbi decoding**

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

$$Z^* = \arg \max_Z p(X, Z|\Theta)$$

**Posterior decoding**

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

$$z^*_n = \arg \max_{z_n} p(z_n|x_1, \ldots, x_N)$$
Posterior decoding

Given $X$, find $Z^*$, where $z^*_n = \arg\max_{z_n} p(z_n|x_1, \ldots, x_N)$ is the most likely state to be in the $n$'th step.

\[
p(z_n|x_1, \ldots, x_N) = \frac{p(z_n, x_1, \ldots, x_N)}{p(x_1, \ldots, x_N)}
\]
\[
= \frac{p(x_1, \ldots, x_n, z_n)p(x_{n+1}, \ldots, x_N|z_n)}{p(x_1, \ldots, x_N)}
\]
\[
= \frac{\alpha(z_n)\beta(z_n)}{p(X)}
\]

\[
z^*_n = \arg\max_{z_n} p(z_n|x_1, \ldots, x_N) = \arg\max_{z_n} \frac{\alpha(z_n)\beta(z_n)}{p(X)}
\]
Posterior decoding

\( \alpha(z_n) \) is the joint probability of observing \( x_1, \ldots, x_n \) and being in state \( z_n \)

\[
\alpha(z_n) \equiv p(x_1, \ldots, x_n, z_n)
\]

\( \beta(z_n) \) is the conditional probability of future observation \( x_{n+1}, \ldots, x_N \) assuming being in state \( z_n \)

\[
\beta(z_n) \equiv p(x_{n+1}, \ldots, x_N | z_n)
\]

\( \alpha[k][n] = \alpha(z_n) \) if \( z_n \) is state \( k \)

\( \beta[k][n] = \beta(z_n) \) if \( z_n \) is state \( k \)
Posterior decoding

$\alpha(z_n)$ is the joint probability of observing $x_1, \ldots, x_n$ and being in state $z_n$

$$\alpha(z_n) \equiv p(x_1, \ldots, x_n, z_n)$$

$\beta(z_n)$ is the conditional probability of future observation $x_{n+1}, \ldots, x_N$ assuming being in state $z_n$

$$\beta(z_n) \equiv p(x_{n+1}, \ldots, x_N | z_n)$$

Using $\alpha(z_n)$ and $\beta(z_n)$ we get the likelihood of the observations as:

$$p(X) = \sum_{z_n} \alpha(z_n) \beta(z_n)$$

$$p(X) = \sum_{z_n} \alpha(z_N)$$

$$z_n^* = \arg \max_{z_n} p(z_n | x_1, \ldots, x_N) = \arg \max_{z_n} \alpha(z_n) \beta(z_n) / p(X)$$
The forward algorithm

$\alpha(z_n)$ is the joint probability of observing $x_1, \ldots, x_n$ and being in state $z_n$

$$\alpha(z_n) \equiv p(x_1, \ldots, x_n, z_n)$$

$\alpha[k][n] = \alpha(z_n)$ if $z_n$ is state $k$
The $\alpha$-recursion

\[
\begin{align*}
\alpha(z_n) &= p(x_1, \ldots, x_n, z_n) \\
&= \sum_{z_1, \ldots, z_{n-1}} p(x_1, \ldots, x_n, z_1, \ldots, z_n) \\
&= \sum_{z_1, \ldots, z_{n-1}} p(z_1) \prod_{i=2}^{n} p(z_i | z_{i-1}) \prod_{i=1}^{n} p(x_i | z_i) \\
&= p(x_n | z_n) \sum_{z_1, \ldots, z_{n-1}} p(z_1) \prod_{i=2}^{n} p(z_i | z_{i-1}) \prod_{i=1}^{n-1} p(x_i | z_i) \\
&= p(x_n | z_n) \sum_{z_1, \ldots, z_{n-1}} p(z_1) \prod_{i=2}^{n-1} p(z_i | z_{i-1}) \prod_{i=1}^{n-1} p(x_i | z_i) \\
&= p(x_n | z_n) \sum_{z_{n-1} z_1, \ldots, z_{n-2}} p(z_1) \prod_{i=2}^{n-1} p(z_i | z_{i-1}) \prod_{i=1}^{n-1} p(x_i | z_i) \\
&= p(x_n | z_n) \sum_{z_{n-1}} p(z_n | z_{n-1}) \sum_{z_1, \ldots, z_{n-2}} p(z_1) \prod_{i=2}^{n-1} p(z_i | z_{i-1}) \prod_{i=1}^{n-1} p(x_i | z_i) \\
&= p(x_n | z_n) \sum_{z_{n-1}} p(z_n | z_{n-1}) \alpha(z_{n-1})
\end{align*}
\]
The $\alpha$-recursion

$$\alpha(z_n) = p(x_1, \ldots, x_n, z_n)$$

$$= \sum_{z_1, \ldots, z_{n-1}} p(x_1, \ldots, x_n, z_1, \ldots, z_n)$$

Using HMM

$$= \sum_{z_1, \ldots, z_{n-1}} p(z_1) \prod_{i=2}^{n} p(z_i | z_{i-1}) \prod_{i=1}^{n} p(x_i | z_i)$$

$$= p(x_n | z_n) \sum_{z_1, \ldots, z_{n-1}} p(z_1) \prod_{i=2}^{n} p(z_i | z_{i-1}) \prod_{i=1}^{n-1} p(x_i | z_i)$$

$$= p(x_n | z_n) \sum_{z_1, \ldots, z_{n-1}} p(z_1) p(z_n | z_{n-1}) \prod_{i=2}^{n-1} p(z_i | z_{i-1}) \prod_{i=1}^{n-1} p(x_i | z_i)$$

$$= p(x_n | z_n) \sum_{z_{n-1}} \sum_{z_1, \ldots, z_{n-2}} p(z_1) p(z_n | z_{n-1}) \prod_{i=2}^{n-1} p(z_i | z_{i-1}) \prod_{i=1}^{n-1} p(x_i | z_i)$$

$$= p(x_n | z_n) \sum_{z_{n-1}} p(z_n | z_{n-1}) \sum_{z_1, \ldots, z_{n-2}} p(z_1) \prod_{i=2}^{n-1} p(z_i | z_{i-1}) \prod_{i=1}^{n-1} p(x_i | z_i)$$

$$= p(x_n | z_n) \sum_{z_{n-1}} p(z_n | z_{n-1}) \alpha(z_{n-1})$$
The forward algorithm

\( \alpha(z_n) \) is the joint probability of observing \( x_1, \ldots, x_n \) and being in state \( z_n \)

\[
\alpha(z_n) \equiv p(x_1, \ldots, x_n, z_n)
\]

Recursion:

\[
\alpha(z_n) = p(x_n | z_n) \sum_{z_{n-1}} \alpha(z_{n-1}) p(z_n | z_{n-1})
\]

Basis:

\[
\alpha(z_1) = p(x_1, z_1) = p(z_1) p(x_1 | z_1)
\]

\( \alpha[k][n] = \alpha(z_n) \) if \( z_n \) is state \( k \)
The forward algorithm

// Pseudo code for computing $\alpha[k][n]$ for some $n>1$

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

if $p(x[n] \mid k) \neq 0$:
    for $j = 1$ to $K$:
        if $p(k \mid j) \neq 0$:
            $\alpha[k][n] = \alpha[k][n] + p(x[n] \mid k) \ast \alpha[j][n-1] \ast p(k \mid j)$

Recursion:

$\alpha(z_n) = p(x_n \mid z_n) \sum_{z_{n-1}} \alpha(z_{n-1}) p(z_n \mid z_{n-1})$

Basis:

$\alpha(z_1) = p(x_1, z_1) = p(z_1) p(x_1 \mid z_1)$

$\alpha[k][n] = \alpha(z_n)$ if $z_n$ is state $k$
The forward algorithm

// Pseudo code for computing $\alpha[k][n]$ for some $n>1$

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

if $p(x[n] | k) \neq 0$:
    for $j = 1$ to $K$:
        if $p(k | j) \neq 0$:
            $\alpha[k][n] = \alpha[k][n] + p(x[n] | k) \times \alpha[j][n-1] \times p(k | j)$

Recursion:

$\alpha(z_n) = p(x_n | z_n) \sum_{z_{n-1}} \alpha(z_{n-1}) p(z_n | z_{n-1})$

Basis:

$\alpha(z_1) = p(x_1, z_1) = p(z_1) p(x_1 | z_1)$

Computing $\alpha$ takes time $O(K^2 N)$ and space $O(KN)$ using memorization.
The backward algorithm

\( \beta(z_n) \) is the conditional probability of future observation \( x_{n+1}, \ldots, x_N \) assuming being in state \( z_n \)

\[
\beta(z_n) \equiv p(x_{n+1}, \ldots, x_N | z_n)
\]

\( \beta[k][n] = \beta(z_n) \) if \( z_n \) is state \( k \)
The $\beta$-recursion

$$
\beta(z_n) = p(x_{n+1}, \ldots, x_N | z_n)
= \sum_{z_{n+1}, \ldots, z_N} p(x_{n+1}, \ldots, x_N, z_{n+1}, \ldots, z_N | z_n)
= \sum_{z_{n+1}, \ldots, z_N} p(x_{n+1}, \ldots, x_N, z_n, z_{n+1}, \ldots, z_N) / p(z_n)
= \sum_{z_{n+1}, \ldots, z_N} p(z_n) \prod_{i=n+1}^{N} p(z_i | z_{i-1}) \prod_{i=n+1}^{N} p(x_i | z_i) / p(z_n)
= \sum_{z_{n+1}, \ldots, z_N} \prod_{i=n+1}^{N} p(z_i | z_{i-1}) \prod_{i=n+1}^{N} p(x_i | z_i)
= \sum_{z_n} \sum_{z_{n+1}, z_{n+2}, \ldots, z_N} p(z_{n+1} | z_n) p(x_{n+1} | z_{n+1}) \prod_{i=n+2}^{N} p(z_i | z_{i-1}) \prod_{i=n+2}^{N} p(x_i | z_i)
= \sum_{z_n} p(z_{n+1} | z_n) p(x_{n+1} | z_{n+1}) \sum_{z_{n+2}, \ldots, z_N} \prod_{i=n+2}^{N} p(z_i | z_{i-1}) \prod_{i=n+2}^{N} p(x_i | z_i)
= \sum_{z_{n+1}} p(z_{n+1} | z_n) p(x_{n+1} | z_{n+1}) p(x_{n+2}, \ldots, x_N | z_{n+1})
= \sum_{z_{n+1}} p(z_{n+1} | z_n) p(x_{n+1} | z_{n+1}) \beta(z_{n+1})
$$
The $\beta$-recursion

$$
\beta(z_n) = p(x_{n+1}, \ldots, x_N | z_n)
$$

$$
= \sum_{z_{n+1}, \ldots, z_N} p(x_{n+1}, \ldots, x_N, z_{n+1}, \ldots, z_N | z_n)
$$

$$
= \sum_{z_{n+1}, \ldots, z_N} \frac{p(x_{n+1}, \ldots, x_N, z_n, z_{n+1}, \ldots, z_N)}{p(z_n)}
$$

Using HMM

$$
= \sum_{z_{n+1}, \ldots, z_N} \frac{p(z_n)}{p(z_n)} \prod_{i=n+1}^{N} p(z_i | z_{i-1}) \prod_{i=n+1}^{N} \frac{p(x_i | z_i)}{p(z_n)}
$$

$$
= \sum_{z_{n+1}, \ldots, z_N} \prod_{i=n+1}^{N} p(z_i | z_{i-1}) \prod_{i=n+1}^{N} p(x_i | z_i)
$$

$$
= \sum_{z_{n+1}} \sum_{z_{n+2}, \ldots, z_N} p(z_{n+1} | z_n) p(x_{n+1} | z_{n+1}) \prod_{i=n+2}^{N} p(z_i | z_{i-1}) \prod_{i=n+2}^{N} p(x_i | z_i)
$$

$$
= \sum_{z_{n+1}} p(z_{n+1} | z_n) p(x_{n+1} | z_{n+1}) \sum_{z_{n+2}, \ldots, z_N} \prod_{i=n+2}^{N} p(z_i | z_{i-1}) \prod_{i=n+2}^{N} p(x_i | z_i)
$$

$$
= \sum_{z_{n+1}} p(z_{n+1} | z_n) p(x_{n+1} | z_{n+1}) p(x_{n+2}, \ldots, x_N | z_{n+1})
$$

$$
= \sum_{z_{n+1}} p(z_{n+1} | z_n) p(x_{n+1} | z_{n+1}) \beta(z_{n+1})
$$
The backward algorithm

$\beta(z_n)$ is the conditional probability of future observation $x_{n+1}, \ldots, x_N$ assuming being in state $z_n$

$$\beta(z_n) \equiv p(x_{n+1}, \ldots, x_N | z_n)$$

Recursion:

$$\beta(z_n) = \sum_{z_{n+1}} \beta(z_{n+1}) p(x_{n+1} | z_{n+1}) p(z_{n+1} | z_n)$$

Basis:

$$\beta(z_N) = 1$$

$\beta[k][n] = \beta(z_n)$ if $z_n$ 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 $K$:
    if $p(j \mid k) \neq 0$:
        $\beta[k][n] = \beta[k][n] + p(j \mid k) \cdot p(x[n+1] \mid j) \cdot \beta[j][n+1]$

Recursion:

$\beta(z_n) = \sum_{z_{n+1}} \beta(z_{n+1}) p(x_{n+1} \mid z_{n+1}) p(z_{n+1} \mid z_n)$

Basis:

$\beta(z_N) = 1$

$\beta[k][n] = \beta(z_n)$ if $z_n$ 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 $K$:
    if $p(j | k) \neq 0$:
        $\beta[k][n] = \beta[k][n] + p(j | k) \cdot p(x[n+1] | j) \cdot \beta[j][n+1]$

Recursion:

$\beta(z_n) = \sum_{z_{n+1}} \beta(z_{n+1}) p(x_{n+1} | z_{n+1}) p(z_{n+1} | z_n)$

Basis:

$\beta(z_N) = 1$

Computing $\beta$ takes time $O(K^2N)$ and space $O(KN)$ using memorization.
Posterior decoding

$\alpha(z_n)$ is the joint probability of observing $x_1, \ldots, x_n$ and being in state $z_n$

$$\alpha(z_n) \equiv p(x_1, \ldots, x_n, z_n)$$

$\beta(z_n)$ is the conditional probability of future observation $x_{n+1}, \ldots, x_N$ assuming being in state $z_n$

$$\beta(z_n) \equiv p(x_{n+1}, \ldots, x_N | z_n)$$

Using $\alpha(z_n)$ and $\beta(z_n)$ we get the likelihood of the observations as:

$$p(X) = \sum_{z_n} \alpha(z_n) \beta(z_n)$$

$$p(X) = \sum_{z_N} \alpha(z_N)$$

$$z^*_n = \arg\max_{z_n} p(z_n | x_1, \ldots, x_N) = \arg\max_{z_n} \alpha(z_n) \beta(z_n) / p(X)$$
Posterior decoding $\alpha(z_n)$ is the joint probability of observing $x_1, \ldots, x_n$ and being in state $z_n$. $\beta(z_n)$ is the conditional probability of future observation $x_{n+1}, \ldots, x_N$ assuming being in state $z_n$.

Using $\alpha(z_n)$ and $\beta(z_n)$ we get the likelihood of the observations as:

$$p(X) = \sum_{z_n} \alpha(z_n) \beta(z_n)$$

The optimal state sequence $z^*_n = \arg \max_{z_n} p(z_n | x_1, \ldots, x_N) = \arg \max_{z_n} \alpha(z_n) \beta(z_n) / p(X)$.
A sequence of states $z_1, \ldots, z_N$ where $p(x_1, \ldots, x_N, z_1, \ldots, z_N) > 0$ 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 $X$?
Viterbi vs. Posterior decoding

A sequence of states \( z_1, \ldots, z_N \) where \( p(x_1, \ldots, x_N, z_1, \ldots, z_N) > 0 \) 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\(\ldots\)2 or 13\(\ldots\)3 with equal probability

I.e. Viterbi finds either 12\(\ldots\)2 or 13\(\ldots\)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.

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

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

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

- Terminology of hidden Markov models (HMMs)

- Viterbi- and Posterior decoding for finding a plausible underlying explanation (sequence of hidden states) of a sequence of observation

- 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: \( \omega(z_n) = p(x_n | z_n) \max_{z_{n-1}} \omega(z_{n-1}) p(z_n | z_{n-1}) \)

Basis: \( \omega(z_1) = p(x_1, z_1) = p(z_1)p(x_1 | z_1) \)

Forward

Recursion: \( \alpha(z_n) = p(x_n | z_n) \sum z_{n-1} \alpha(z_{n-1}) p(z_n | z_{n-1}) \)

Basis: \( \alpha(z_1) = p(x_1, z_1) = p(z_1)p(x_1 | z_1) \)

Backward

Recursion: \( \beta(z_n) = \sum z_{n+1} \beta(z_{n+1}) p(x_{n+1} | z_{n+1}) p(z_{n+1} | z_n) \)

Basis: \( \beta(z_N) = 1 \)
**Problem:** The values in the $\omega$-, $\alpha$-, and $\beta$-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:**

$$\alpha(z_n) = p(x_n | z_n) \sum_{z_{n-1}} \alpha(z_{n-1}) p(z_n | z_{n-1})$$

**Basis:**

$$\alpha(z_1) = p(x_1, z_1) = p(z_1) p(x_1 | z_1)$$

**Backward**

**Recursion:**

$$\beta(z_n) = \sum_{z_{n+1}} \beta(z_{n+1}) p(x_{n+1} | z_{n+1}) p(z_{n+1} | z_n)$$

**Basis:**

$$\beta(z_N) = 1$$