Hidden Markov Models
Terminology and Basic Algorithms
Machine learning, a branch of artificial intelligence, is about the construction and study of systems that can learn from data. For example, a machine learning system could be trained on email messages to learn to distinguish between spam and non-spam messages. After learning, it can then be used to classify new email messages into spam and non-spam folders.

[...]

Machine learning focuses on prediction, based on known properties learned from the training data.
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.

To us, the core of machine learning (in bioinformatics) boils down to three things:

**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** using the model on new data.
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 (iid) ...

but this assumption is not always the best, fx (1) measurements of weather patterns, (2) daily values of stocks, (3) acoustic features in successive time frames used for speech recognition, (4) the composition of texts, (5) the composition of DNA, 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 ...
Markov Models

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

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

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...
If the distributions \( p(x_n | x_{n-1}) \) are the same for all \( n \), then the chain of observations is an \textit{homogeneous} 1st-order Markov chain...

The model, i.e. \( p(x_n | x_{n-1}) \):

A sequence of observations:

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

Extension – A higher order Markov chain

\[
p(x_1, \ldots, x_N) = p(x_1) p(x_2 | x_1) \prod_{n=3}^{N} p(x_n | x_{n-1}, x_{n-2})
\]

...observations is an \textit{homogeneous} 1st-order Markov chain...
Hidden Markov Models

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

If the latent 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 latent (i.e. hidden) variable?

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

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

Computational problems

- Determine the likelihood of the sequence of observations
- Predict the next observation in the sequence of observations
- Find the most likely underlying explanation of the sequence of observation (i.e. most likely sequence of hidden values).
Hidden Markov Models

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

The predictive distribution

\[
p(x_{n+1} | x_1, \ldots, x_n)
\]

for observation \( x_{n+1} \) can be shown to depend on all previous observations, i.e. the sequence of observations is not a Markov chain of any order ...

If the latent 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 latent 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 latent 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 latent variable?

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

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

**Transition probabilities:** If the latent 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:

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

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

In Bishop, the latent variables $z_n$ are discrete variables, e.g. if $z_n = (0,0,1)$ then the model in step $n$ is in state $k = 3$ ...

If the latent 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:

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

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

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

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

\[ \sum_k A_{jk} = 1 \]

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

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

\[ \sum_k \pi_k = 1 \]
Transition probabilities: If the latent 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 $\pi$ ...
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 ...
Emission probabilities: The conditional distributions of the observed variables \( p(x_n \mid z_n) \) 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.

\[
p(x_n \mid z_n, \phi) = \prod_{k=1}^{K} p(x_n \mid \phi_k)^{z_{nk}}
\]
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...

$$z_{nk} = 1 \text{ iff the } n\text{'th latent variable in the sequence is in state } k, \text{ otherwise it is } 0, \text{ i.e. the product just “picks” the emission probabilities corresponding to state } k ...$$

$$p(x_n | z_n, \phi) = \prod_{k=1}^{K} p(x_n | \phi_k)^{z_{nk}}$$
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 ...

Model $M$:

A run follows a sequence of states:

```
H   H   L   L   L   H
```

And emits a sequence of symbols:

```
\[\text{sun} \quad \text{sun} \quad \text{cloud} \quad \text{sun} \quad \text{sun} \quad \text{sun}\]
```
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 ...

A special **End**-state can be added to generate finite output.

Model $M$:

A run follows a sequence of states:

$$
\begin{array}{ccccccccc}
H & H & L & L & L & L & H \\
\end{array}
$$

And emits a sequence of symbols:

$$
\begin{array}{cccccc}
\text{Sun} & \text{Sun} & \text{Rain} & \text{Sun} & \text{Sun} & \text{Sun} \\
\end{array}
$$
Using HMMs

- Determine the likelihood of the sequence of observations
- Predict the next observation in the sequence of observations
- Find the most likely underlying explanation of the sequence of observation
Using HMMs

- Determine the likelihood of the sequence of observations
- Predict the next observation in the sequence of observations
- Find the most likely underlying explanation of the sequence of observation

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

- Determine the likelihood of the sequence of observations
- Predict the next observation in the sequence of observations
- Find the most likely underlying explanation of the sequence of observations

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

The sum has \(K^N\) terms, but it can be computed in \(O(K^2N)\) time ...
The forward-backward algorithm

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

$$\alpha(z_n) \equiv p(x_1, \ldots, x_n, z_n)$$
The forward-backward algorithm

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

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

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

\[ \beta(z_n) \equiv p(x_{n+1}, \ldots, x_N | z_n) \]
The forward-backward algorithm

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

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

\( \beta(z_n) \) 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:

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

\[
p(X) = \sum_{z_N} \alpha(z_N)
\]
The forward algorithm

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

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

\[
\alpha(z_n) = p(x_1, \ldots, x_n, z_n) = p(x_1, \ldots, x_n | z_n) p(z_n) = p(x_n | z_n) p(x_1, \ldots, x_{n-1} | z_n) p(z_n) = p(x_n | z_n) \sum_{z_{n-1}} p(x_1, \ldots, x_{n-1}, z_n, z_{n-1}, z_n) = p(x_n | z_n) \sum_{z_{n-1}} p(x_1, \ldots, x_{n-1}, z_n | z_{n-1}) p(z_n | z_{n-1}) p(z_n) = p(x_n | z_n) \sum_{z_{n-1}} p(x_1, \ldots, x_{n-1}, z_{n-1}, z_n | z_{n-1}) p(z_n | z_{n-1}) p(z_n) = p(x_n | z_n) \sum_{z_{n-1}} \alpha(z_{n-1}) p(z_n | z_{n-1})
\]
The $\alpha$-recursion

$$\alpha(z_n) = p(x_1, \ldots, x_n, z_n)$$
$$= p(x_1, \ldots, x_n | z_n) p(z_n)$$
$$= p(x_n | z_n) p(x_1, \ldots, x_{n-1} | z_n) p(z_n)$$
$$= p(x_n | z_n) p(x_1, \ldots, x_{n-1}, z_n)$$
$$= p(x_n | z_n) \sum_{z_{n-1}} p(x_1, \ldots, x_{n-1}, z_{n-1}, z_n)$$
$$= p(x_n | z_n) \sum_{z_{n-1}} p(x_1, \ldots, x_{n-1}, z_n | z_{n-1}) p(z_{n-1})$$
$$= p(x_n | z_n) \sum_{z_{n-1}} p(x_1, \ldots, x_{n-1} | z_{n-1}) p(z_n | z_{n-1}) p(z_{n-1})$$
$$= p(x_n | z_n) \sum_{z_{n-1}} p(x_1, \ldots, x_{n-1}, z_{n-1}) p(z_n | z_{n-1})$$
$$= p(x_n | z_n) \sum_{z_{n-1}} \alpha(z_{n-1}) p(z_n | z_{n-1})$$
The $\alpha$-recursion

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

Prod. rule

\[ = p(x_1, \ldots, x_n | z_n)p(z_n) \]

Using HMM

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

Prod. rule

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

Sum rule

\[ = p(x_n | z_n) \sum_{z_{n-1}} p(x_1, \ldots, x_{n-1}, z_n, z_{n-1}) \]

Prod. rule

\[ = p(x_n | z_n) \sum_{z_{n-1}} p(x_1, \ldots, x_{n-1}, z_n | z_{n-1})p(z_n | z_{n-1})p(z_{n-1}) \]

Using HMM

\[ = p(x_n | z_n) \sum_{z_{n-1}} p(x_1, \ldots, x_{n-1} | z_{n-1})p(z_n | z_{n-1})p(z_{n-1}) \]

Prod. rule

\[ = p(x_n | z_n) \sum_{z_{n-1}} p(x_1, \ldots, x_{n-1}, z_{n-1})p(z_n | z_{n-1}) \]

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

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

\[ \alpha(z_1) = p(x_1, z_1) = p(z_1) p(x_1 | z_1) = \prod_{k=1}^{K} \{ \pi_k p(x_1 | \phi_k) \}^{z_{1k}} \]
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:

\[
\alpha(z_1) = p(x_1, z_1) = p(z_1) p(x_1 | z_1) = \prod_{k=1}^{K} \{ \pi_k p(x_1 | \phi_k) \}^{z_{1,k}}
\]

Takes time \( O(K^2N) \) 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 \( z_n \)

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

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

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

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

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

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

\[
= \sum_{z_{n+1}} \beta(z_{n+1}) p(x_{n+1} | z_{n+1}) p(z_{n+1} | z_n)
\]
The $\beta$-recursion

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

Sum rule
\[
= \sum_{z_{n+1}} p(x_{n+1}, \ldots, x_N, z_{n+1} | z_n)
\]

Prod. rule
\[
= \sum_{z_{n+1}} p(x_{n+1}, \ldots, x_N | z_n, z_{n+1}) p(z_{n+1} | z_n)
\]

Using HMM
\[
= \sum_{z_{n+1}} p(x_{n+1}, \ldots, x_N | z_{n+1}) p(z_{n+1} | z_n)
\]

Using HMM
\[
= \sum_{z_{n+1}} p(x_{n+2}, \ldots, x_N | z_{n+1}) p(x_{n+1} | z_{n+1}) p(z_{n+1} | z_n)
\]

\[
= \sum_{z_{n+1}} \beta(z_{n+1}) p(x_{n+1} | z_{n+1}) p(z_{n+1} | z_n)
\]
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:**

\[
\beta(z_n) = \sum_{z_{n+1}} \beta(z_{n+1}) p(x_{n+1} | z_{n+1}) p(z_{n+1} | z_n)
\]
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:**

$$\beta(z_N) = 1$$
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:

\[
\beta(z_N) = 1
\]

Takes time \( O(K^2N) \) and space \( O(KN) \) using memorization
Using HMMs

- Determine the likelihood of the sequence of observations
- Predict the next observation in the sequence of observations
- Find the most likely underlying explanation of the sequence of observation

\[ p(\mathbf{x}_{N+1} | \mathbf{X}) \]
Predicting the next observation

\[ p(x_{N+1}|X) = \sum_{z_{N+1}} p(x_{N+1}, z_{N+1}|X) \]

\[ = \sum_{z_{N+1}} p(x_{N+1}|z_{N+1}) p(z_{N+1}|X) \]

\[ = \sum_{z_{N+1}} p(x_{N+1}|z_{N+1}) \sum_{z_N} p(z_{N+1}, z_N|X) \]

\[ = \sum_{z_{N+1}} p(x_{N+1}|z_{N+1}) \sum_{z_N} p(z_{N+1}|z_N) p(z_N|X) \]

\[ = \sum_{z_{N+1}} p(x_{N+1}|z_{N+1}) \sum_{z_N} p(z_{N+1}|z_N) \frac{p(z_N, X)}{p(X)} \]

\[ = \frac{1}{p(X)} \sum_{z_{N+1}} p(x_{N+1}|z_{N+1}) \sum_{z_N} p(z_{N+1}|z_N) \alpha(z_N) \]
Predicting the next observation

\[ p(x_{N+1}|X) = \sum_{z_{N+1}} p(x_{N+1}, z_{N+1}|X) \]

Prod. rule and HMM
\[ = \sum_{z_{N+1}} p(x_{N+1}|z_{N+1}) p(z_{N+1}|X) \]

Sum rule
\[ = \sum_{z_{N+1}} p(x_{N+1}|z_{N+1}) \sum_{z_N} p(z_{N+1}, z_N|X) \]

Prod. rule and HMM
\[ = \sum_{z_{N+1}} p(x_{N+1}|z_{N+1}) \sum_{z_N} p(z_{N+1}|z_N) p(z_N|X) \]

Prod. rule
\[ = \sum_{z_{N+1}} p(x_{N+1}|z_{N+1}) \sum_{z_N} p(z_{N+1}|z_N) \frac{p(z_N, X)}{p(X)} \]
\[ = \frac{1}{p(X)} \sum_{z_{N+1}} p(x_{N+1}|z_{N+1}) \sum_{z_N} p(z_{N+1}|z_N) \alpha(z_N) \]
Predicting the next observation

$$p(x_{N+1}|X) = \sum_{z_{N+1}} p(x_{N+1}, z_{N+1}|X)$$

Prod. rule and HMM

$$= \sum_{z_{N+1}} p(x_{N+1}|z_{N+1}) p(z_{N+1}|X)$$

Sum rule

$$= \sum_{z_{N+1}} p(x_{N+1}|z_{N+1}) \sum_{z_N} p(z_{N+1}, z_N|X)$$

Prod. rule and HMM

$$= \sum_{z_{N+1}} p(x_{N+1}|z_{N+1}) \sum_{z_N} p(z_{N+1}|z_N) p(z_N|X)$$

Prod. rule

$$= \sum_{z_{N+1}} p(x_{N+1}|z_{N+1}) \sum_{z_N} p(z_{N+1}|z_N) \frac{p(z_N, X)}{p(X)}$$

$$= \frac{1}{p(X)} \sum_{z_{N+1}} p(x_{N+1}|z_{N+1}) \sum_{z_N} p(z_{N+1}|z_N) \alpha(z_N)$$

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

- Determine the likelihood of the sequence of observations
- Predict the next observation in the sequence of observations
- Find the most likely underlying explanation of the sequence of observation

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

The Viterbi algorithm: Finds the most probable sequence of states generating the observations ...
The Viterbi algorithm

\[ \omega(z_n) \] is the probability of the most likely sequence of states \( z_1, \ldots, z_n \) generating the observations \( x_1, \ldots, x_n \)

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

**Intuition:** Find the “longest path” from column 1 to column n, where “length” is its total probability, i.e. the probability of the transitions and emissions along the path ...
The Viterbi algorithm

ω(\(z_n\)) is the probability of the most likely sequence of states \(z_1, \ldots, z_n\) generating the observations \(x_1, \ldots, x_n\)

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

Recursion:

\[\omega(z_n) = p(x_n | z_n) \max_{z_{n-1}} \omega(z_{n-1}) p(z_n | z_{n-1})\]
The Viterbi algorithm

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

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

Recursion:

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

Basis:

$\omega(z_1) = p(x_1, z_1) = p(z_1) p(x_1 | z_1) = \prod_{k=1}^{K} \{\pi_k p(x_1 | \phi_k)\}^{z_{1k}}$

Takes time $O(K^2 N)$ and space $O(KN)$ using memorization
The Viterbi algorithm

\( \omega(z_n) \) is the probability of the most likely sequence of states \( z_1, \ldots, z_n \) generating the observations \( x_1, \ldots, x_n \)

Recursion:

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

Basis:

\[
\omega(z_1) = p(x_1 | z_1) \max_{z_{1-1}} \omega(z_{n-1}) p(z_1 | z_{n-1})
\]

The path itself can be retrieved in time \( O(KN) \) by backtracking.

Takes time \( O(K^2N) \) and space \( O(KN) \) using memorization.
Another kind of decoding

$\alpha(z_n)$ is the joint probability of observing $x_1,...,x_n$ 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 $x_{n+1},...,x_N$ 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 can also find the most likely state to be in the $n$'th step:

$$z^*_n = \arg \max_{z_n} P(z_n | x_1, \ldots, x_n) = \arg \max_{z_n} \alpha(z_n) \beta(z_n) / P(X)$$

Finding $z^*_1,...,z^*_N$ is called posterior decoding.
Another kind of decoding

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

Using \(\alpha(z_n)\) and \(\beta(z_n)\) we can also find the most likely state to be in the \(n\)'th step:

\[
z_n^* = \arg \max_{z_n} P(z_n | x_1, \ldots, x_n) = \arg \max_{z_n} \frac{\alpha(z_n) \beta(z_n)}{P(X)}
\]

Finding \(z_1^*, \ldots, z_N^*\) is called **posterior decoding**.
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 $X$?

Using $\alpha(z_n)$ and $\beta(z_n)$ we can also find the most likely state to be in the $n$'th step:

$$z_n^* = \arg \max_{z_n} P(z_n | x_1, \ldots, x_n) = \arg \max_{z_n} \alpha(z_n) \beta(z_n) / P(X)$$

Finding $z_1^*, \ldots, z_N^*$ is called **posterior decoding**.
Summary

- Introduced hidden Markov models (HMMs)
- The forward-backward algorithms for determining the likelihood of a sequence of observations, and predicting the next observation in a sequence of observations.
- The Viterbi-algorithm for finding the most likely underlying explanation (sequence of latent states) of a sequence of observation
- Next: How to implement the basic algorithms (forward, backward, and Viterbi) in a “numerically” sound manner.