Probabilistic Modelling of Sequences

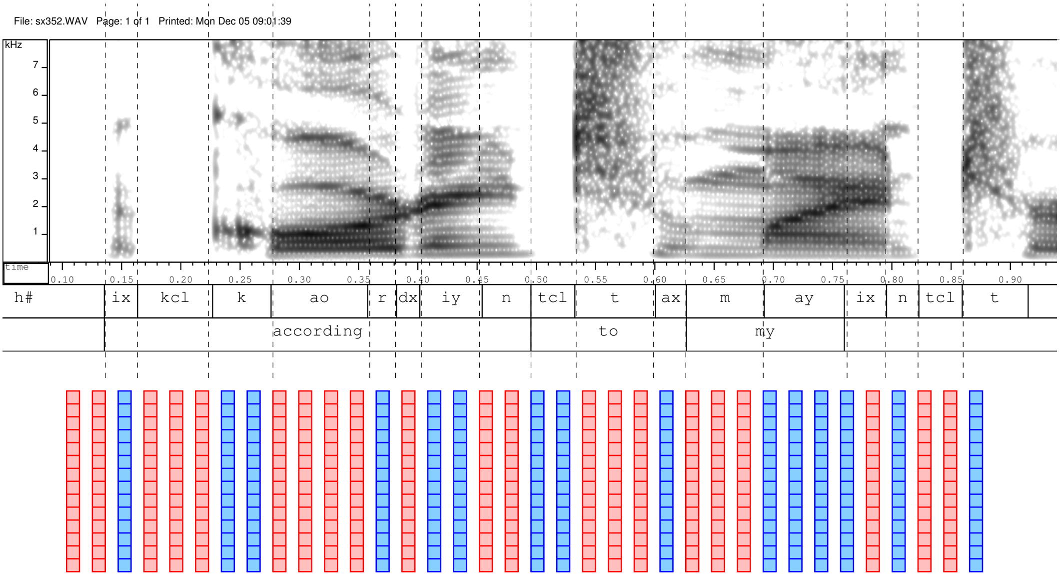
TTT4185 Machine Learning for Signal Processing

Giampiero Salvi

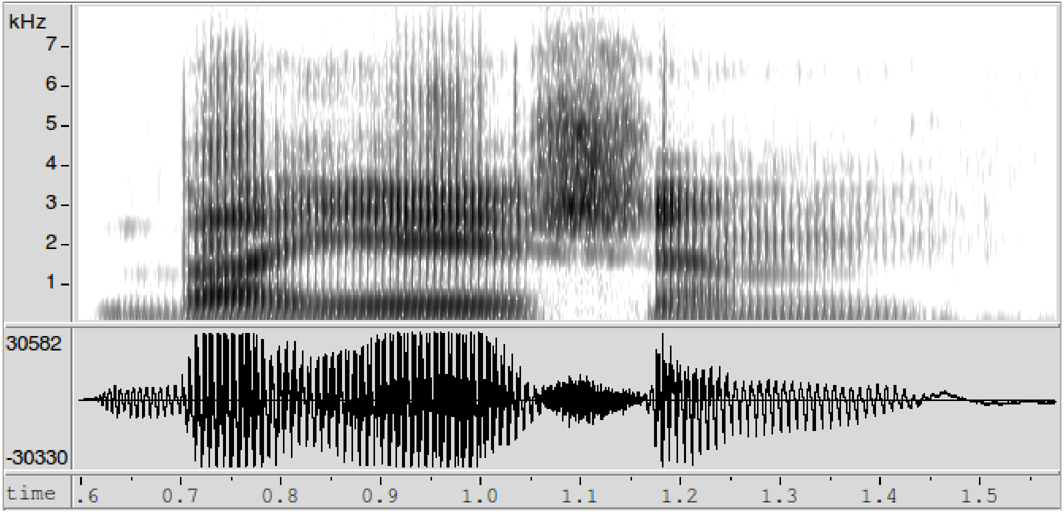
Department of Electronic Systems
NTNU

HT2020

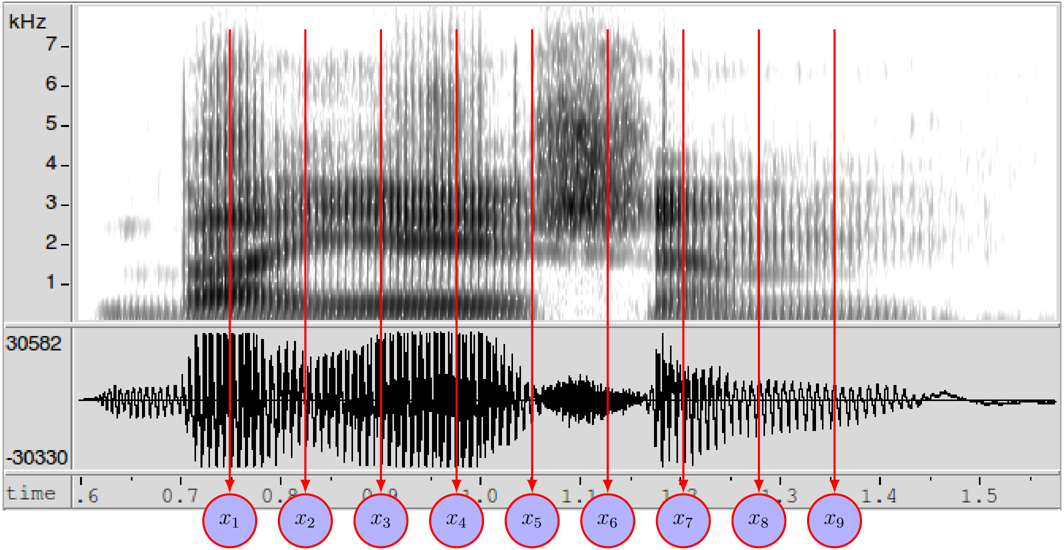
Frame-Based Processing



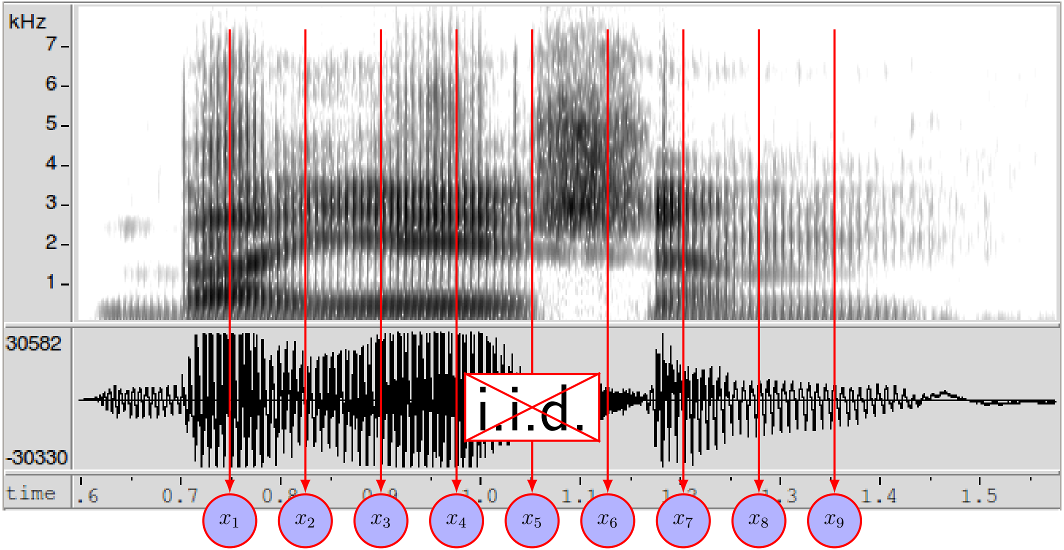
Sequences in Statistical Terms



Sequences in Statistical Terms



Sequences in Statistical Terms

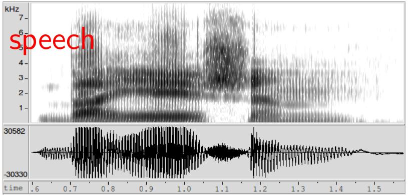


Sequential Data: Not Only Speech

Time sequences

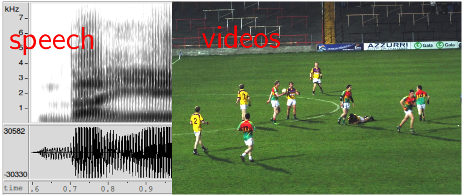
Sequential Data: Not Only Speech

Time sequences



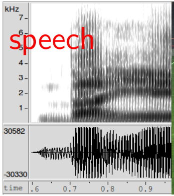
Sequential Data: Not Only Speech

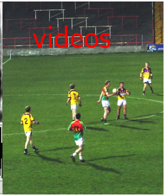
Time sequences



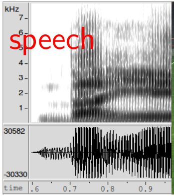
Sequential Data: Not Only Speech

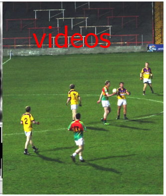
Time sequences

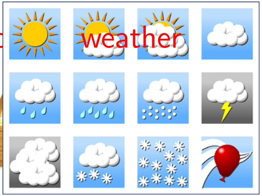




Time sequences





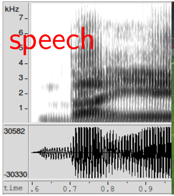


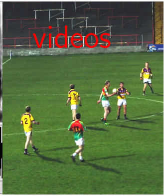
Sequential Data: Not Only Speech

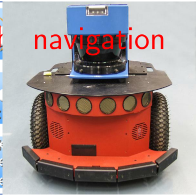
Time sequences

Sequential Data: Not Only Speech

Time sequences



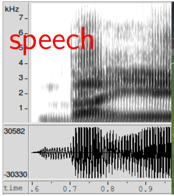


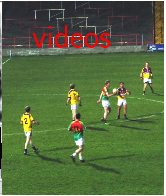


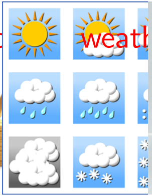
Timeless sequences

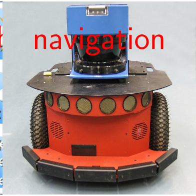
Sequential Data: Not Only Speech

Time sequences

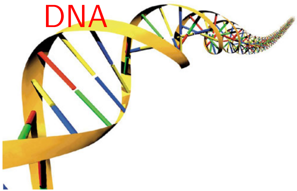








Timeless sequences



text

Nel mezzo del cammin di nostra vita
mi ritrovai per una selva oscura,
ché la diritta via era smarrita.
Ahi quanto a dir qual era è cosa dura
esta selva selvaggia e aspra e forte
che nel pensier rinova la paura!
Tant' è amara che poco è più morte;
ma per trattar del ben ch'i' vi trovai,
dirò de l'altre cose ch'i' v'ho scorte.
Io non so ben ridir com' i' v'intrai,
tant' era pien di sonno a quel punto
che la verace via abbandonai.

Historical Perspective

- Hidden Markov Models first studied in the '60s[1][2]
- applied to ASR in the mid '70s[3]
- later seen as special case of Bayesian Networks[4]

---

[1] R. Stratonovich. "Conditional Markov Processes". In: *Theory of Probability and its Applications* 5.2 (1960), pp. 156–178.

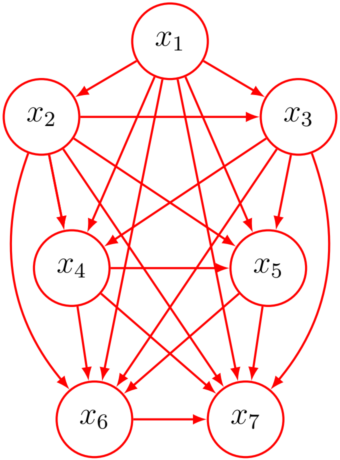
[2] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains". In: *Ann. Math. Statist.* 41.1 (1970), pp. 164–171.

[3] J. Baker. "The DRAGON system—An overview". In: *IEEE Trans. Acoust., Speech, Signal Process.* 23 (1975), pp. 24–29.

[4] J. Pearl. "Bayesian networks: a model of self-activated memory for evidential reasoning". In: *Proceedings of the 7th Conference of the Cognitive Science Society*. University of California, Irvine, Aug. 1985, pp. 329–334.

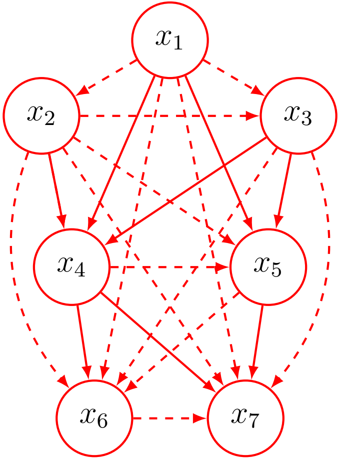
Bayesian Networks (reminder)

$$
\begin{aligned}
p(x_1, \ldots, x_7) = \\
& p(x_1) \\
& p(x_2|x_1) \\
& p(x_3|x_1, x_2) \\
& p(x_4|x_1, x_2, x_3) \\
& p(x_5|x_1, x_2, x_3, x_4) \\
& p(x_6|x_1, x_2, x_3, x_4, x_5) \\
& p(x_7|x_1, x_2, x_3, x_4, x_5, x_6)
\end{aligned}
$$



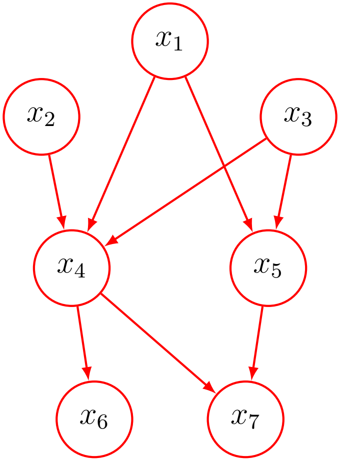
Bayesian Networks (reminder)

$$
\begin{aligned}
p(x_1, \ldots, x_7) = \\
&p(x_1) \\
&p(x_2 | x_1) \\
&p(x_3 | x_1, x_2) \\
&p(x_4 | x_1, x_2, x_3) \\
&p(x_5 | x_1, x_2, x_3, x_4) \\
&p(x_6 | x_1, x_2, x_3, x_4, x_5) \\
&p(x_7 | x_1, x_2, x_3, x_4, x_5, x_6)
\end{aligned}
$$



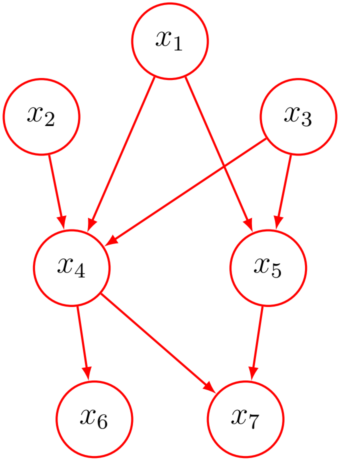
Bayesian Networks (reminder)

$$
\begin{aligned}
p(x_1, \ldots, x_7) = \\
& p(x_1) \\
& p(x_2) \\
& p(x_3) \\
& p(x_4 | x_1, x_2, x_3) \\
& p(x_5 | x_1, x_3) \\
& p(x_6 | x_4) \\
& p(x_7 | x_4, x_5)
\end{aligned}
$$



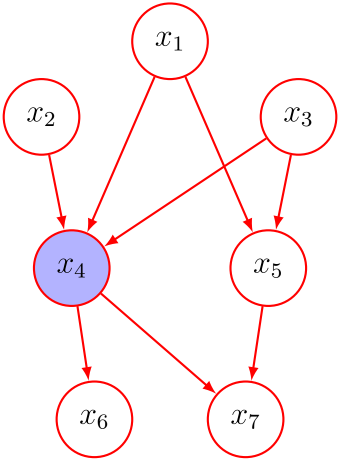
Bayesian Networks (reminder)

$$p(x_1, \ldots, x_7) = \prod_{k=1}^{K} p(x_k | \mathsf{pa}_k)$$



Bayesian Networks (reminder)

If we observe x_4...



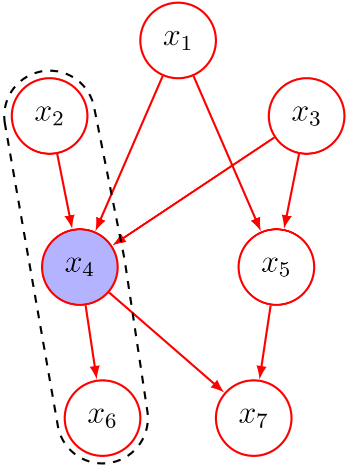
Bayesian Networks (reminder)

If we observe $x_4 \ldots$

d-separation:

Head-to-tail:

x_2 and x_6 conditionally independent



Bayesian Networks (reminder)

If we observe $x_4 \ldots$

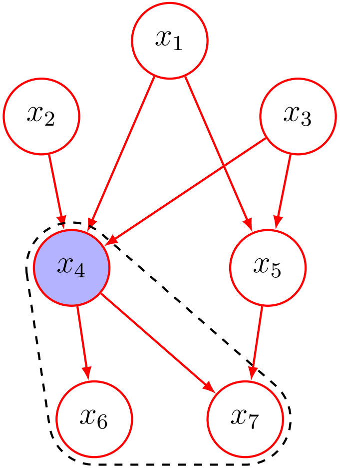
d-separation:

Head-to-tail:

x_2 and x_6 conditionally independent

Tail-to-tail:

x_6 and x_7 conditionally independent



Bayesian Networks (reminder)

If we observe $x_4 \ldots$

d-separation:

Head-to-tail:

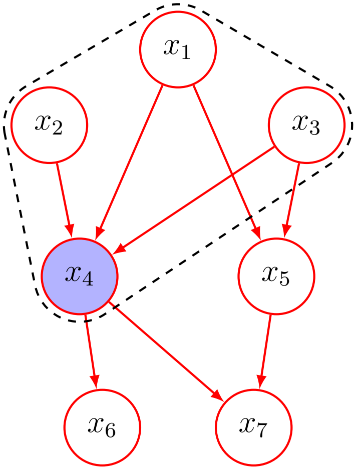
x_2 and x_6 conditionally independent

Tail-to-tail:

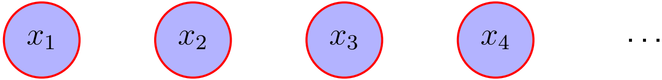
x_6 and x_7 conditionally independent

Head-to-head:

x_1, x_2 and x_3 dependent (explaining away)

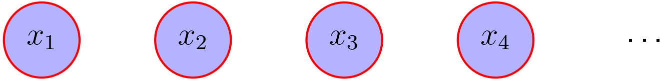


(Dynamic) Bayesian Networks



independence assumption (e.g. i.i.d) not satisfactory

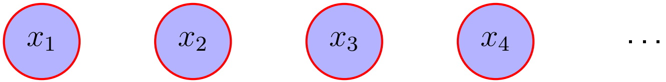
(Dynamic) Bayesian Networks



Most general case, applying chain rule recursively ($p(a, b) = p(a)p(b|a)$)

$$p(x_1, \ldots, x_N) = p(x_1)p(x_2, \ldots, x_N|x_1)$$

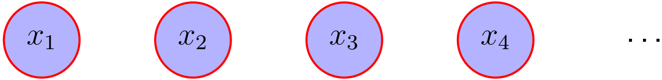
(Dynamic) Bayesian Networks



Most general case, applying chain rule recursively ($p(a, b) = p(a)p(b|a)$)

$$p(x_1, \ldots, x_N) = p(x_1)p(x_2|x_1)p(x_3, \ldots, x_N|x_1, x_2)$$

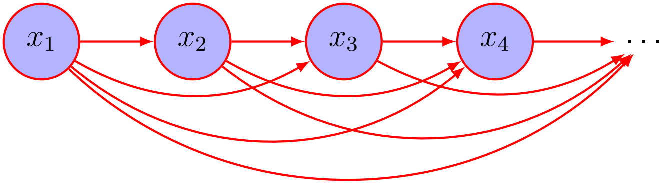
(Dynamic) Bayesian Networks



Most general case, applying chain rule recursively ($p(a, b) = p(a)p(b|a)$)

$$\begin{aligned} p(x_1, \ldots, x_N) =& p(x_1)p(x_2|x_1)p(x_3|x_1, x_2)\cdots \\ & \cdots p(x_N|x_1, \ldots, x_{N-1}) \end{aligned}$$

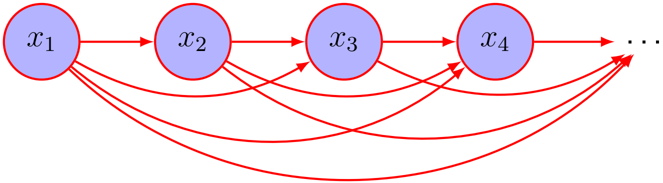
(Dynamic) Bayesian Networks



Most general case, applying chain rule recursively ($p(a, b) = p(a)p(b|a)$)

$$
\begin{aligned}
p(x_1, \ldots, x_N) =& p(x_1)p(x_2|x_1)p(x_3|x_1, x_2) \cdots \\
& \cdots p(x_N|x_1, \ldots, x_{N-1})
\end{aligned}
$$

(Dynamic) Bayesian Networks



Most general case, applying chain rule recursively ($p(a, b) = p(a)p(b|a)$)

$$
\begin{aligned}
p(x_1, \ldots, x_N) = & p(x_1)p(x_2|x_1)p(x_3|x_1, x_2)\cdots \\
& \cdots p(x_N|x_1, \ldots, x_{N-1})
\end{aligned}
$$

Grows quadratically with sequence length (N)!!!

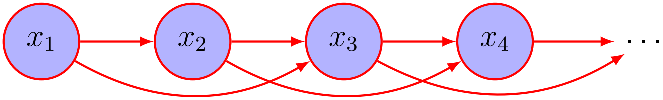
Markov assumption

$$x_1 \to x_2 \to x_3 \to x_4 \to \cdots$$

First order Markov assumption: $p(x_n|x_1,\ldots,x_{n-1}) \approx p(x_n|x_{n-1})$

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

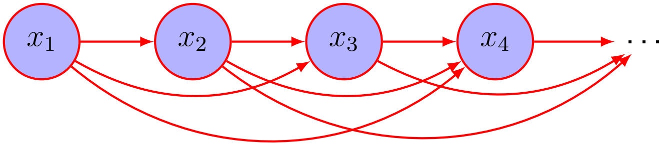
Markov assumption



Second order Markov assumption:

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

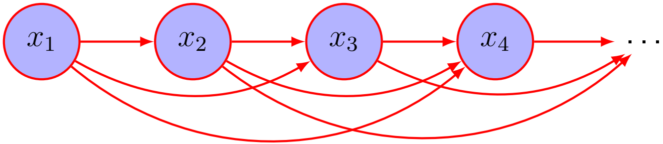
Markov assumption



Third order Markov assumption:

$$p(x_1, \ldots, x_N) \quad = \quad p(x_1)p(x_2|x_1)p(x_3|x_1, x_2)$$
$$\prod_{n=4}^{N} p(x_n|x_{n-3}, x_{n-2}, x_{n-1})$$

Markov assumption



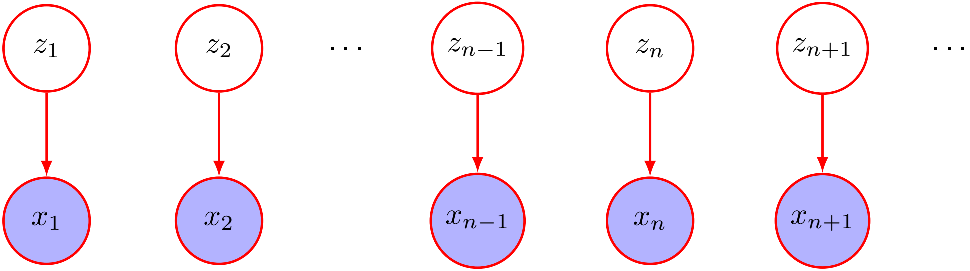
Third order Markov assumption:

$$p(x_1, \ldots, x_N) = p(x_1)p(x_2|x_1)p(x_3|x_1, x_2) \prod_{n=4}^{N} p(x_n|x_{n-3}, x_{n-2}, x_{n-1})$$

Grows quadratically with order!!!

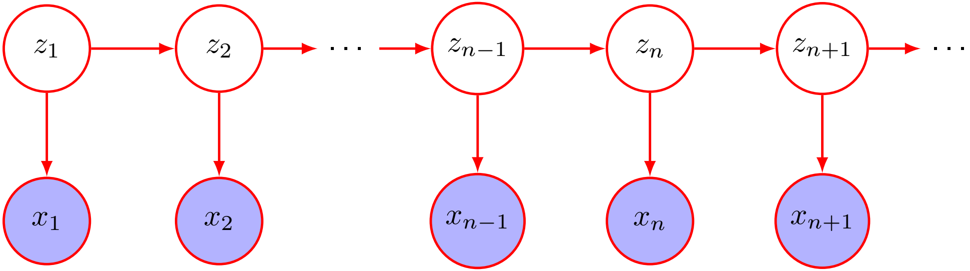
Mixture Models

Adding latent variables z_n

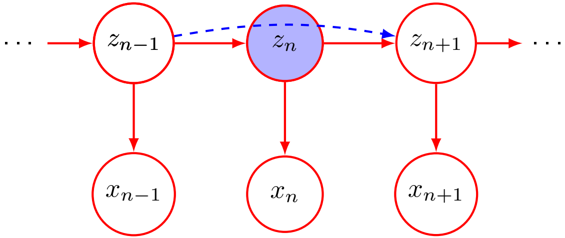


State Space Models

Adding latent variables z_n

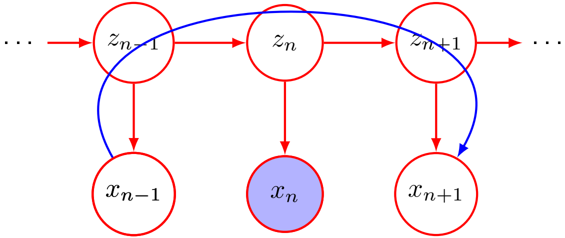


State Space Models: Properties



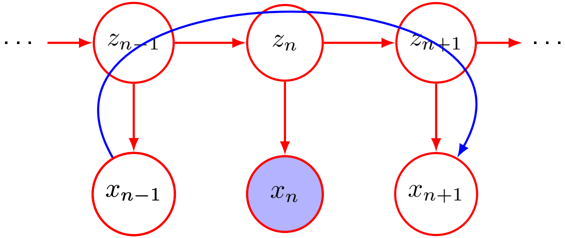
- given z_n, z_{n+1} is independent of $z_1, \ldots, z_{n-1}$
  $p(z_{n+1}|z_1, \ldots, z_n) = p(z_{n+1}|z_n)$

State Space Models: Properties



- given z_n, z_{n+1} is independent of $z_1, \ldots, z_{n-1}$
  $p(z_{n+1}|z_1, \ldots, z_n) = p(z_{n+1}|z_n)$
- $p(x_{n+1}|x_1, \ldots, x_n)$ does not simplify

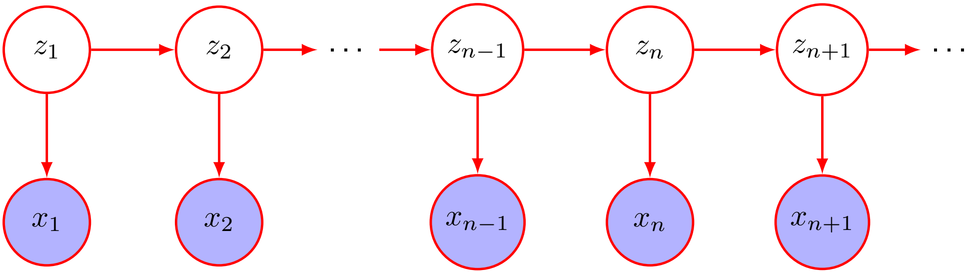
State Space Models: Properties



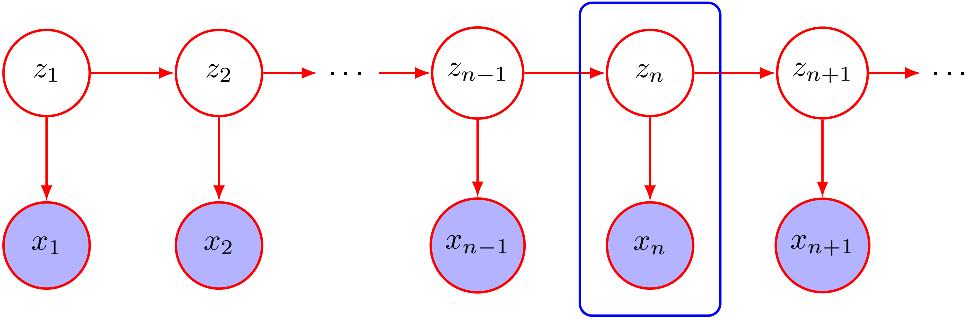
- given z_n, z_{n+1} is independent of $z_1, \ldots, z_{n-1}$
  $p(z_{n+1}|z_1, \ldots, z_n) = p(z_{n+1}|z_n)$
- $p(x_{n+1}|x_1, \ldots, x_n)$ does not simplify

We have modelled indefinitely long dependencies with a limited set of parameters!

Stationary State Space Models

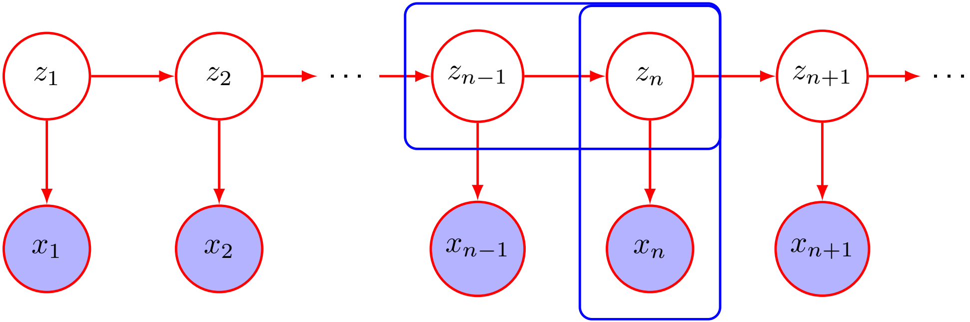


Stationary State Space Models



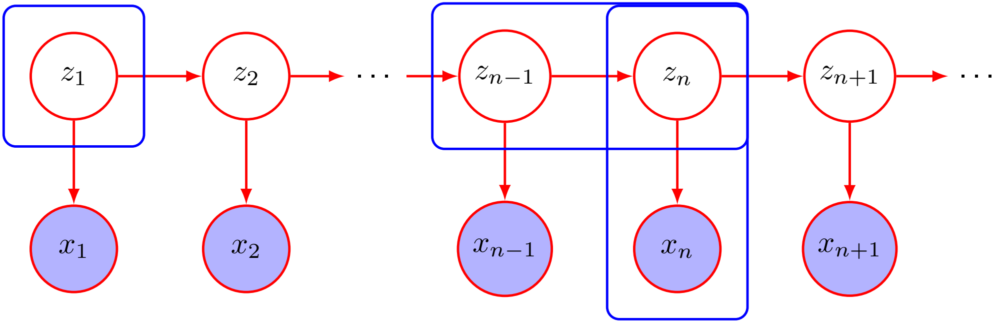
- Emission: $p(x_n|z_n)$

Stationary State Space Models



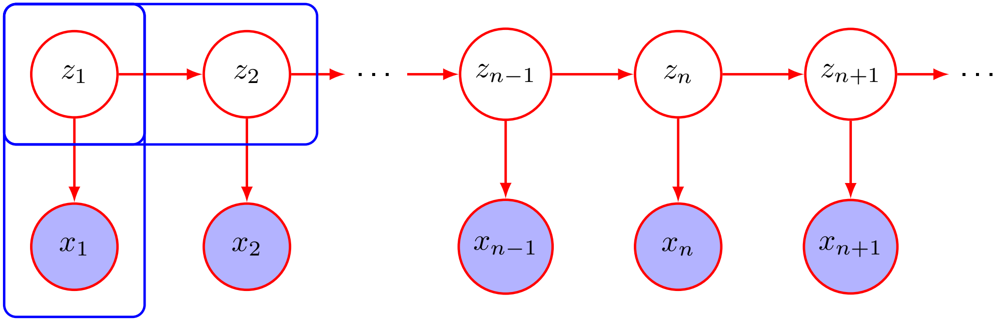
- Emission: $p(x_n|z_n)$
- Transition: $p(z_n|z_{n-1})$

Stationary State Space Models



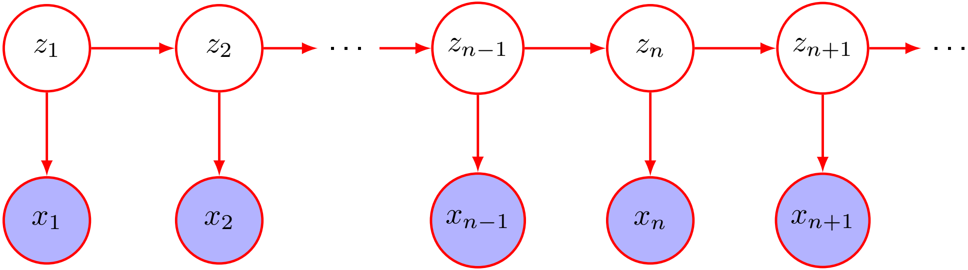
- Emission: $p(x_n|z_n)$
- Transition: $p(z_n|z_{n-1})$
- Initial: $p(z_1)$

Stationary State Space Models



- Emission: $p(x_n|z_n)$
- Transition: $p(z_n|z_{n-1})$
- Initial: $p(z_1)$

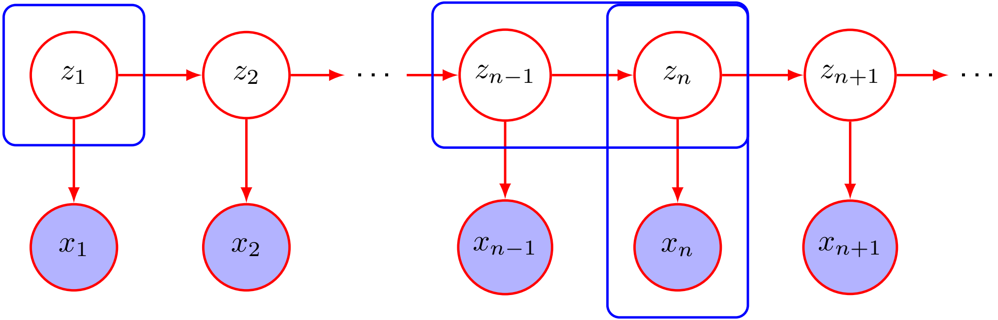
State Space Models Instances



- if z_n are discrete: Hidden Markov Models
- if z_n are continuous: Linear Dynamical Systems

Hidden Markov Models

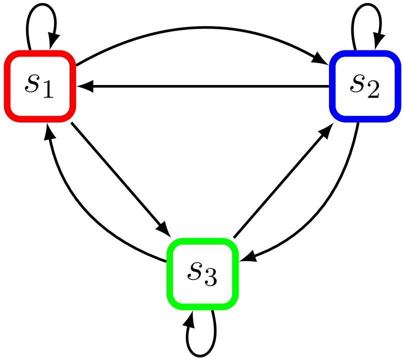
State space models with discrete z_n



- Emission: $p(x_n|z_n) = p(x_n|z_n, \phi)$
  equivalent to Mixture Model
- Transition: $p(z_n|z_{n-1}) = p(z_n|z_{n-1}, A)$
- Initial: $p(z_1) = p(z_1|\pi)$

Hidden Markov Models (HMMs)

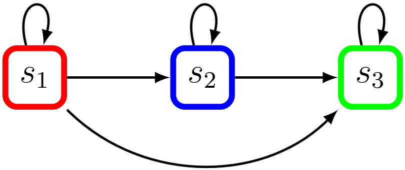
Ergodic HMM



Elements:	
set of states:	$S = \{s_1, s_2, s_3\}$
transition probabilities:	$A(s_a, s_b) = P(s_b, t \mid s_a, t-1)$
prior probabilities:	$\pi(s_a) = P(s_a, t_0)$
state to observation probs:	$\phi(o, s_a) = P(o \mid s_a)$

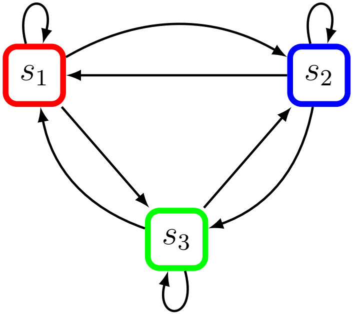
Hidden Markov Models (HMMs)

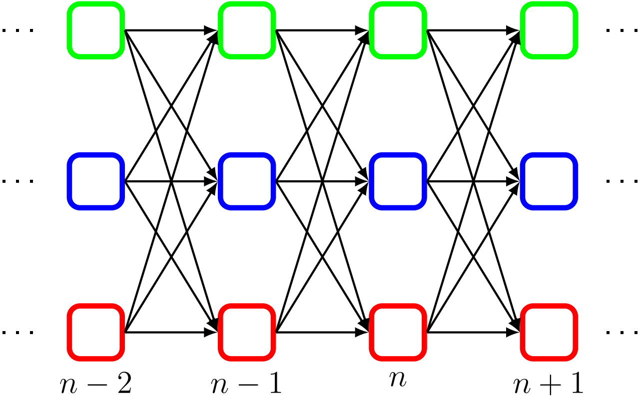
Left-to-right HMM



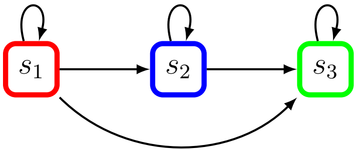
Elements:	
set of states:	$S = \{s_1, s_2, s_3\}$
transition probabilities:	$A(s_a, s_b) = P(s_b, t \mid s_a, t-1)$
prior probabilities:	$\pi(s_a) = P(s_a, t_0)$
state to observation probs:	$\phi(o, s_a) = P(o \mid s_a)$

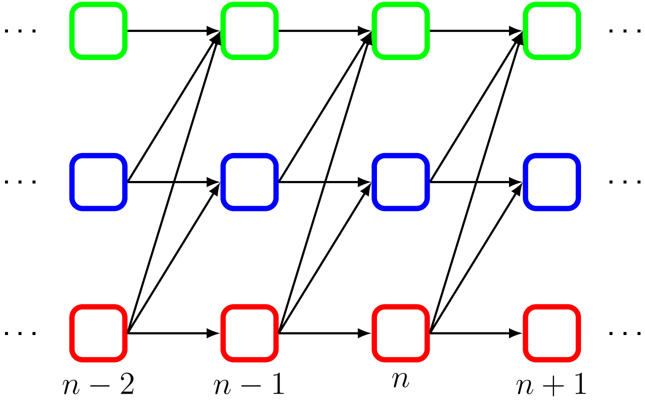
HMMs: Trellis (Lattice)





HMMs: Trellis (Lattice)





A probabilistic perspective: Bayes' rule

$$P(\text{words}|\text{sounds}) = \frac{P(\text{sounds}|\text{words})P(\text{words})}{P(\text{sounds})}$$

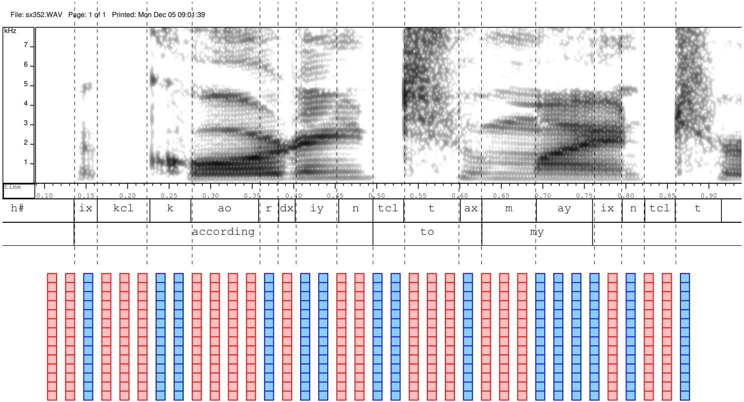
- $P(\text{sounds}|\text{words})$ can be estimated from training data and transcriptions
- $P(\text{words})$: *a priori* probability of the words (Language Model)
- $P(\text{sounds})$: *a priori* probability of the sounds (constant, can be ignored)

Probabilistic Modelling

Problem: How do we model $P(\text{sounds}|\text{words})$?

Probabilistic Modelling

Problem: How do we model $P(\text{sounds}|\text{words})$?

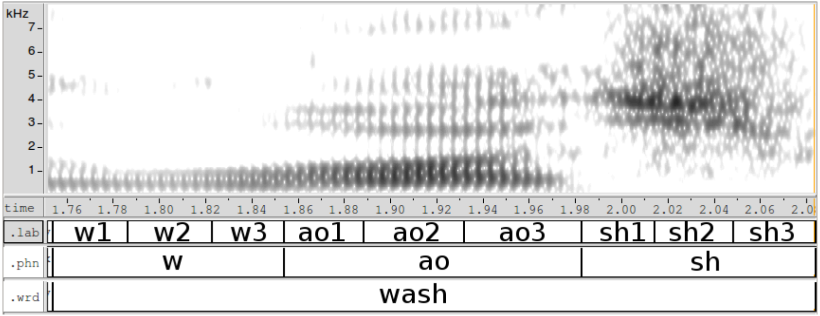


Every feature vector (observation at time t) is a continuous stochastic variable (e.g. MFCC)

Stationarity

Problem: speech is not stationary

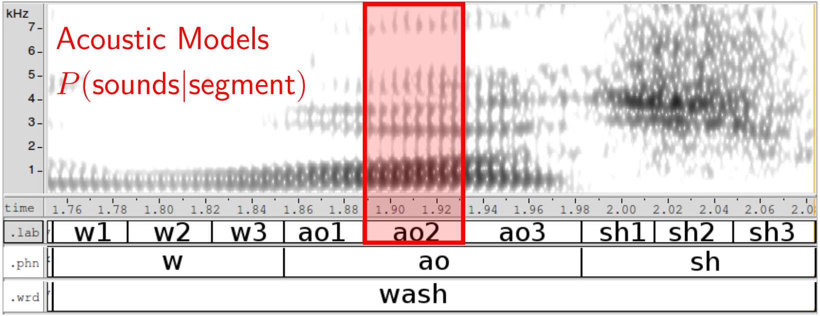
- we need to model short segments independently
- the fundamental unit can not be the word, but must be shorter
- usually we model three segments for each phoneme



Stationarity

Problem: speech is not stationary

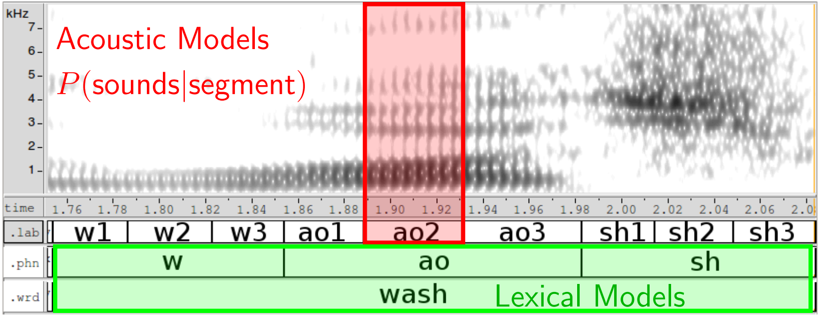
- we need to model short segments independently
- the fundamental unit can not be the word, but must be shorter
- usually we model three segments for each phoneme



Stationarity

Problem: speech is not stationary

- we need to model short segments independently
- the fundamental unit can not be the word, but must be shorter
- usually we model three segments for each phoneme



Local probabilities (frame-wise)

If segment sufficiently short

$$P(\text{sounds}|\text{segment})$$

can be modelled with standard probability distributions

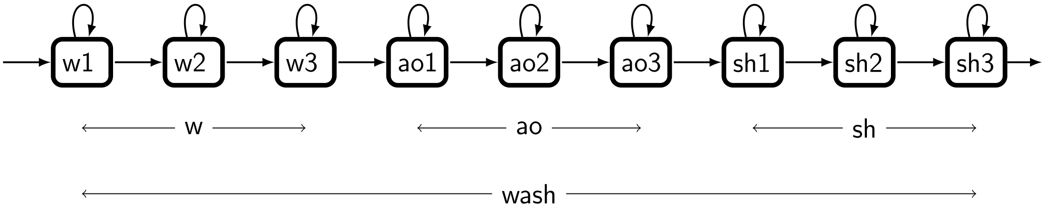
$$\phi_{s_a}(x) = P(x|s_a)$$

Usually Gaussian or Gaussian Mixture

Global Probabilities (utterance)

Problem: How do we combine the different $P(\text{sounds}|\text{segment})$ to form $P(\text{sounds}|\text{words})$?

Answer: Hidden Markov Model (HMM)



HMM-questions (Inference)

1. what is the probability that the model has generated the sequence of observations? (isolated word recognition)

[5] A. J. Viterbi. "Error Bounds for Convolutional Codes and an Asymtotically optimum decoding algorithm". In: *IEEE Trans. Inf. Theory* IT-13 (Apr. 1967), pp. 260–269.

[6] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains". In: *Ann. Math. Statist.* 41.1 (1970), pp. 164–171.

HMM-questions (Inference)

1. what is the probability that the model has generated the sequence of observations? (isolated word recognition) forward algorithm

[5] A. J. Viterbi. "Error Bounds for Convolutional Codes and an Asymtotically optimum decoding algorithm". In: *IEEE Trans. Inf. Theory* IT-13 (Apr. 1967), pp. 260–269.

[6] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains". In: *Ann. Math. Statist.* 41.1 (1970), pp. 164–171.

HMM-questions (Inference)

1. what is the probability that the model has generated the sequence of observations? (isolated word recognition) forward algorithm
2. what is the most likely state sequence given the observation sequence? (continuous speech recognition)

[5] A. J. Viterbi. "Error Bounds for Convolutional Codes and an Asymtotically optimum decoding algorithm". In: *IEEE Trans. Inf. Theory* IT-13 (Apr. 1967), pp. 260–269.

[6] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains". In: *Ann. Math. Statist.* 41.1 (1970), pp. 164–171.

HMM-questions (Inference)

1. what is the probability that the model has generated the sequence of observations? (isolated word recognition) forward algorithm
2. what is the most likely state sequence given the observation sequence? (continuous speech recognition) Viterbi algorithm[5]

[5] A. J. Viterbi. "Error Bounds for Convolutional Codes and an Asymtotically optimum decoding algorithm". In: *IEEE Trans. Inf. Theory* IT-13 (Apr. 1967), pp. 260–269.

[6] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains". In: *Ann. Math. Statist.* 41.1 (1970), pp. 164–171.

HMM-questions (Inference)

1. what is the probability that the model has generated the sequence of observations? (isolated word recognition) forward algorithm
2. what is the most likely state sequence given the observation sequence? (continuous speech recognition) Viterbi algorithm[5]
3. how can the model parameters be estimated from examples? (training)

---

[5] A. J. Viterbi. "Error Bounds for Convolutional Codes and an Asymtotically optimum decoding algorithm". In: *IEEE Trans. Inf. Theory* IT-13 (Apr. 1967), pp. 260–269.

[6] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains". In: *Ann. Math. Statist.* 41.1 (1970), pp. 164–171.

HMM-questions (Inference)

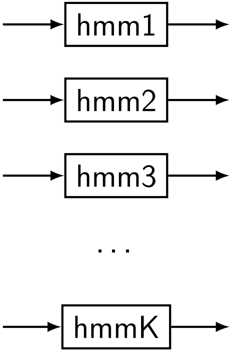
1. what is the probability that the model has generated the sequence of observations? (isolated word recognition) forward algorithm
2. what is the most likely state sequence given the observation sequence? (continuous speech recognition) Viterbi algorithm[5]
3. how can the model parameters be estimated from examples? (training) Baum-Welch[6]

---

[5] A. J. Viterbi. "Error Bounds for Convolutional Codes and an Asymtotically optimum decoding algorithm". In: *IEEE Trans. Inf. Theory* IT-13 (Apr. 1967), pp. 260–269.

[6] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains". In: *Ann. Math. Statist.* 41.1 (1970), pp. 164–171.

Isolated Words Recognition

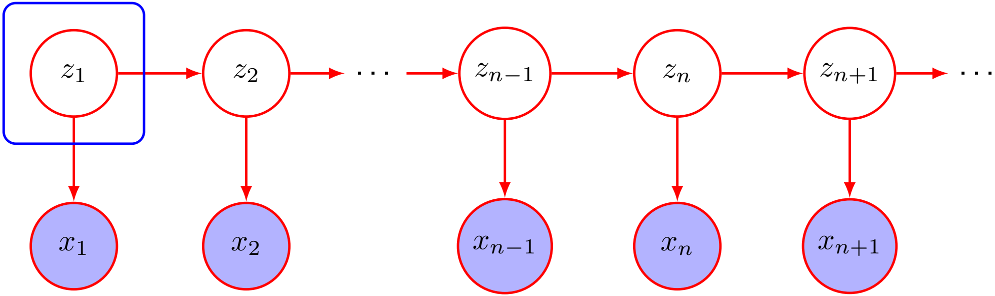


Compare Likelihoods (forward algorithm)

HMM Inference: Joint Distribution

$$X = \{x_1, \ldots, x_N\}$$
$$Z = \{z_1, \ldots, z_N\}$$

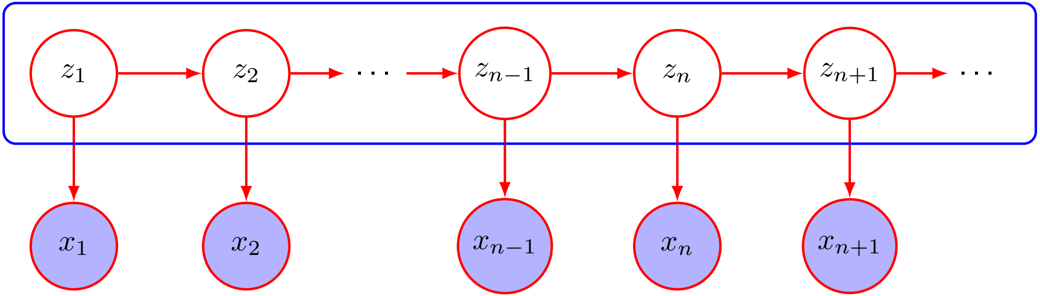
$$P(X, Z|\theta) = p(z_1|\pi) \left[\prod_{n=2}^{N} p(z_n|z_{n-1}, A)\right] \prod_{m=1}^{N} p(x_m|z_m, \phi)$$



HMM Inference: Joint Distribution

$$X = \{x_1, \ldots, x_N\}$$
$$Z = \{z_1, \ldots, z_N\}$$

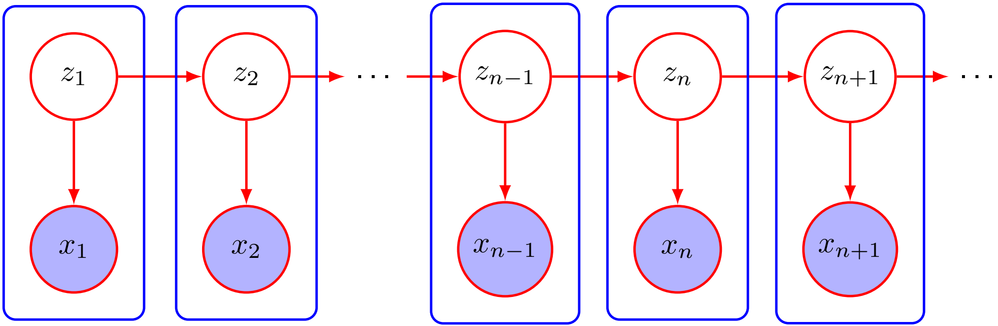
$$P(X, Z|\theta) = p(z_1|\pi) \left[\prod_{n=2}^{N} p(z_n|z_{n-1}, A) \right] \prod_{m=1}^{N} p(x_m|z_m, \phi)$$



HMM Inference: Joint Distribution

$$X = \{x_1, \ldots, x_N\}$$
$$Z = \{z_1, \ldots, z_N\}$$

$$P(X, Z|\theta) = p(z_1|\pi) \left[\prod_{n=2}^{N} p(z_n|z_{n-1}, A) \right] \prod_{m=1}^{N} p(x_m|z_m, \phi)$$

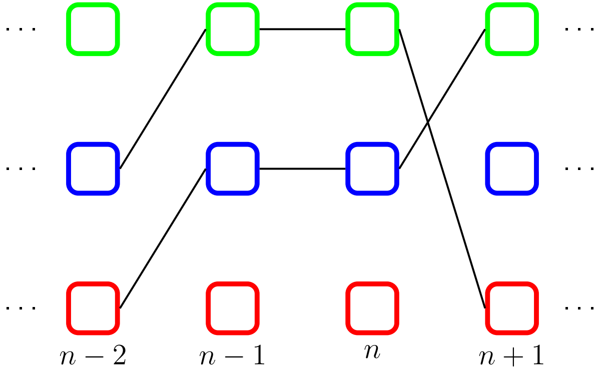


HMM Inference: Likelihood Function

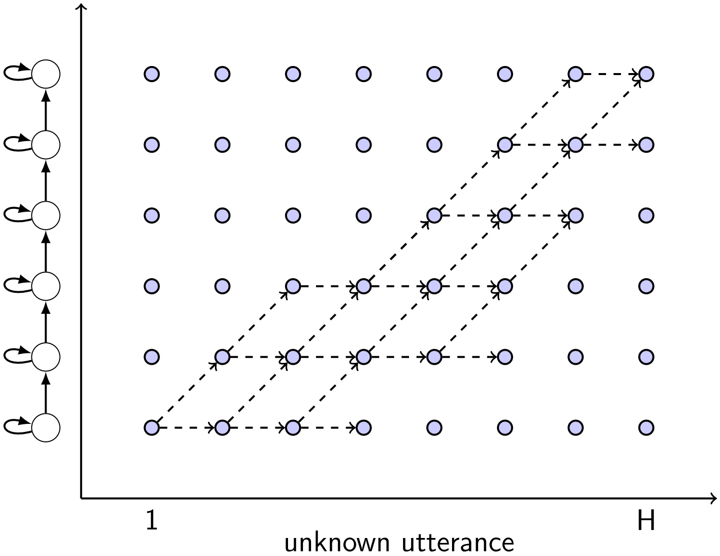
marginalise joint distribution over Z:

$$P(X|\theta) = \sum_Z p(X, Z|\theta)$$

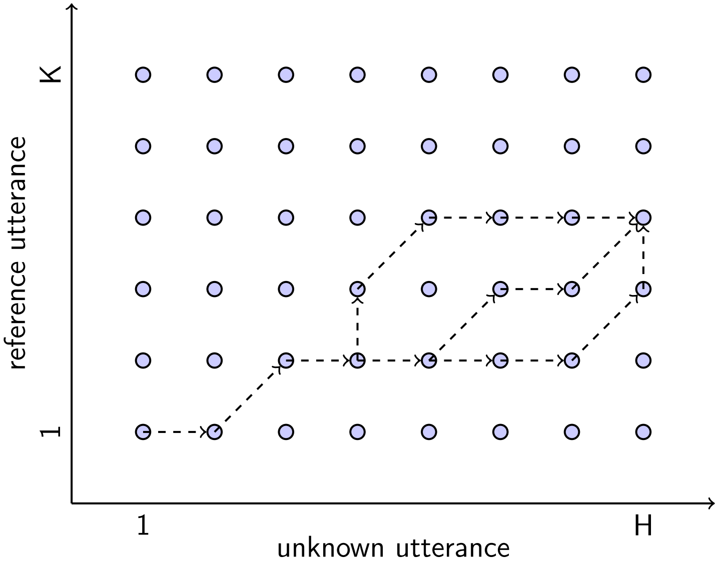
Problem: there are K^N possible sequences for Z



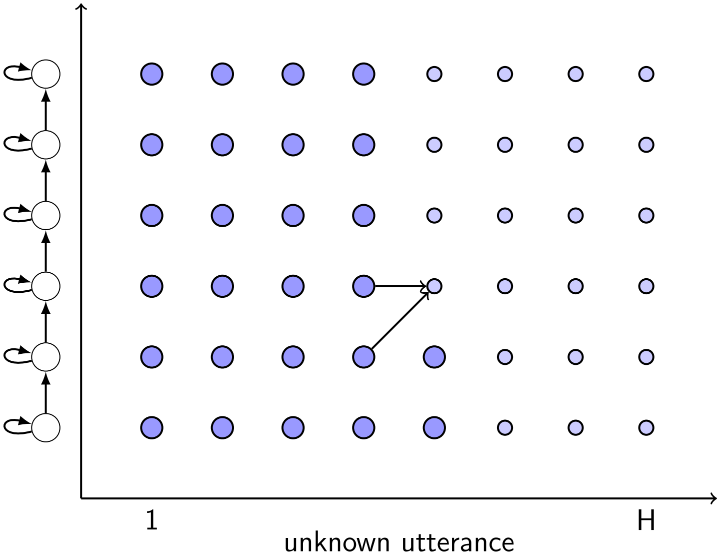
HMM Likelihood



Very Similar to Template Matching



Same Solution: Dynamic Programming



Solution: Forward algorithm

Instead of AccD[h,k] (Template Matching)

$$\alpha_n(j) \equiv p(x_1, \ldots, x_n, z_n = s_j | \theta)$$

At the end, instead of AccD[H,K]:

$$P(X|\theta) = \sum_{i=1}^{M} \alpha_N(i)$$

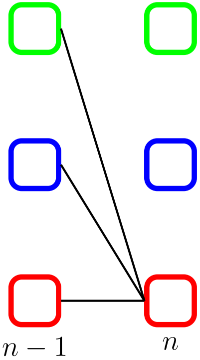
Forward Probability

Initialization:

$$\alpha_1(j) = \pi_j \phi_j(x_1)$$

Recursion:

$$\alpha_n(j) = \left[\sum_{i=1}^{M} \alpha_{n-1}(i) a_{ij} \right] \phi_j(x_n)$$



Forward Probability

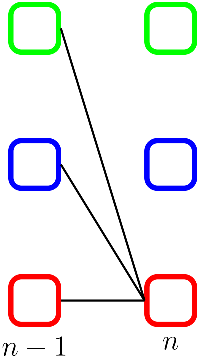
Initialization:

$$\alpha_1(j) = \pi_j \phi_j(x_1)$$

Recursion:

$$\alpha_n(j) = \left[\sum_{i=1}^{M} \alpha_{n-1}(i) a_{ij} \right] \phi_j(x_n)$$

equivalent to sum-product in Bayesian Networks



Backward probability

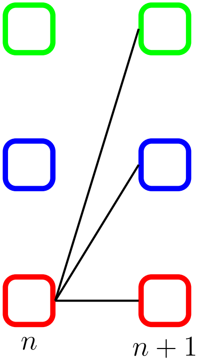
$$\beta_n(i) \equiv p(x_{n+1}, \ldots, x_N | z_n = s_i)$$

Initialization:

$$\beta_N(i) \equiv p(?|z_n = s_i) \equiv 1$$

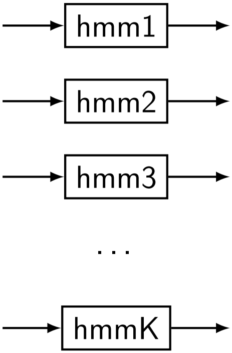
Recursion:

$$\beta_n(i) = \left[\sum_{j=1}^{M} a_{ij} \phi_j(x_{n+1}) \beta_{n+1}(j) \right]$$



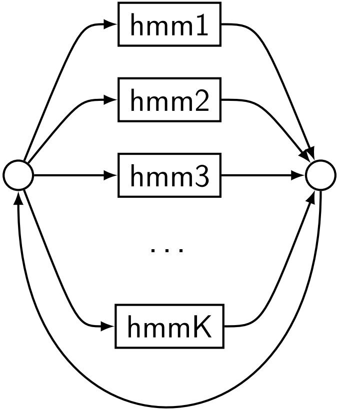
Find best sequence of states: why?

Isolated Words



(Likelihood)

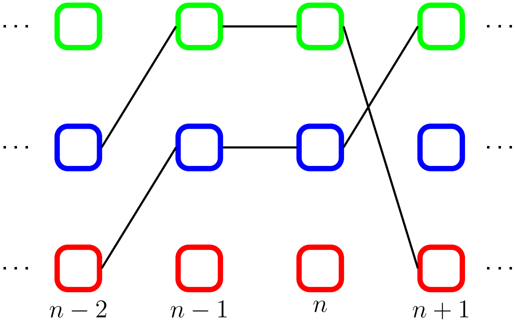
Continuous Speech



(best path)

Find best sequence of states: how?

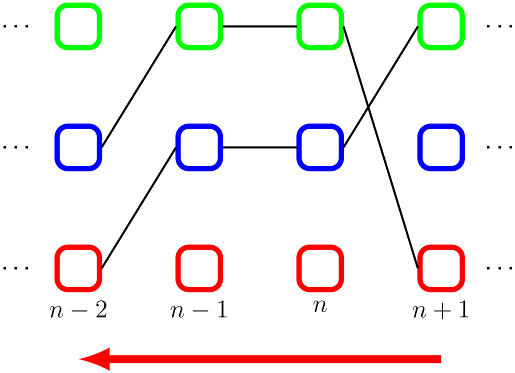
- Viterbi algorithm[7]
- equivalent to max-sum in Bayesian Networks



[7] A. J. Viterbi. "Error Bounds for Convolutional Codes and an Asymtotically optimum decoding algorithm". In: *IEEE Trans. Inf. Theory* IT-13 (Apr. 1967), pp. 260–269.

Find best sequence of states: how?

- Viterbi algorithm[7]
- equivalent to max-sum in Bayesian Networks



[7] A. J. Viterbi. "Error Bounds for Convolutional Codes and an Asymtotically optimum decoding algorithm". In: *IEEE Trans. Inf. Theory* IT-13 (Apr. 1967), pp. 260–269.

Summary: update rules

Forward algorithm (sum-product):

$$\alpha_n(j) = \left[\sum_{i=1}^{M} \alpha_{n-1}(i) a_{ij}\right] \phi_j(x_n)$$

Viterbi algorithm (max-sum):

$$V_n(j) = \max_{i=1}^{M} \left[V_{n-1}(i) a_{ij}\right] \phi_j(x_n)$$

$$B_n(j) = \arg\max_{i=1}^{M} \left[V_{n-1}(i) a_{ij}\right]$$