Hidden Markov Models

Sargur N. Srihari

srihari@cedar.buffalo.edu

Machine Learning Course:

http://www.cedar.buffalo.edu/~srihari/CSE574/index.html

HMM Topics

1. What is an HMM?
2. State-space Representation
3. HMM Parameters
4. Generative View of HMM
5. Determining HMM Parameters Using EM
6. Forward-Backward or $\alpha-\beta$ algorithm
7. HMM Implementation Issues:
   a) Length of Sequence
   b) Predictive Distribution
   c) Sum-Product Algorithm
   d) Scaling Factors
   e) Viterbi Algorithm

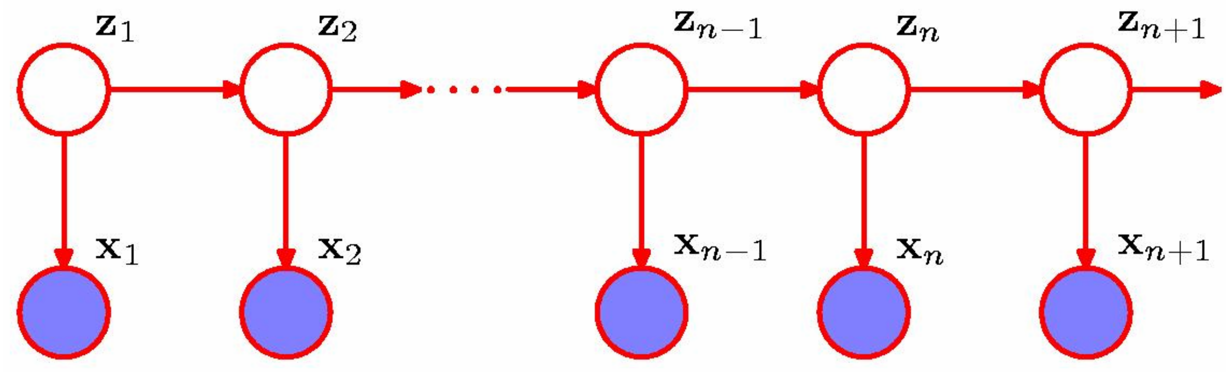
1. What is an HMM?

- Ubiquitous tool for modeling time series data
- Used in
  - Almost all speech recognition systems
  - Computational molecular biology
    - Group amino acid sequences into proteins
  - Handwritten word recognition
- It is a tool for representing probability distributions over sequences of observations
- HMM gets its name from two defining properties:
  - Observation x_t at time t was generated by some process whose state z_t is hidden from the observer
  - Assumes that state at z_t is dependent only on state z_{t-1} and independent of all prior states (First order)
- Example: z are phoneme sequences

  x are acoustic observations

Graphical Model of HMM

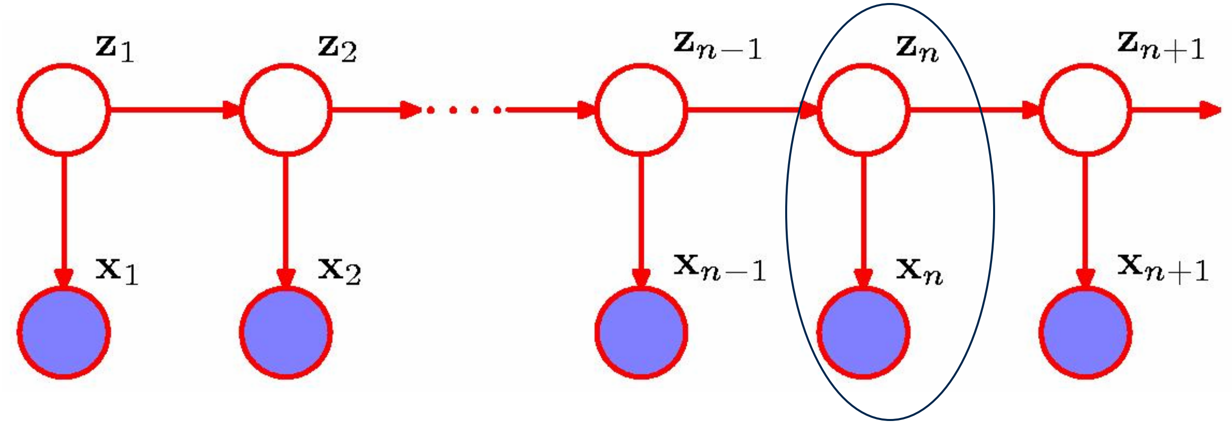
- Has the graphical model shown below and latent variables are discrete



- Joint distribution has the form:

$$p(x_1,..x_N,z_1,..z_n) = p(z_1)\left[\prod_{n=2}^{N} p(z_n \mid z_{n-1})\right]\prod_{n=1}^{N} p(x_n \mid z_n)$$

HMM Viewed as Mixture



- A single time slice corresponds to a mixture distribution with component densities $p(\mathrm{x}|\mathrm{z})$
  - Each state of discrete variable z represents a different component
- An extension of mixture model
  - Choice of mixture component depends on choice of mixture component for previous distribution
- Latent variables are multinomial variables z_n
  - That describe component responsible for generating x_n
- Can use *one-of-K* coding scheme

2. State-Space Representation

- Probability distribution of z_n depends on state of previous latent variable z_{n-1} through probability distribution $p(z_n|z_{n-1})$
- One-of K coding
  - Since latent variables are K-dimensional binary vectors

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

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

- These are known as *Transition Probabilities*
- $K(K-1)$ independent parameters

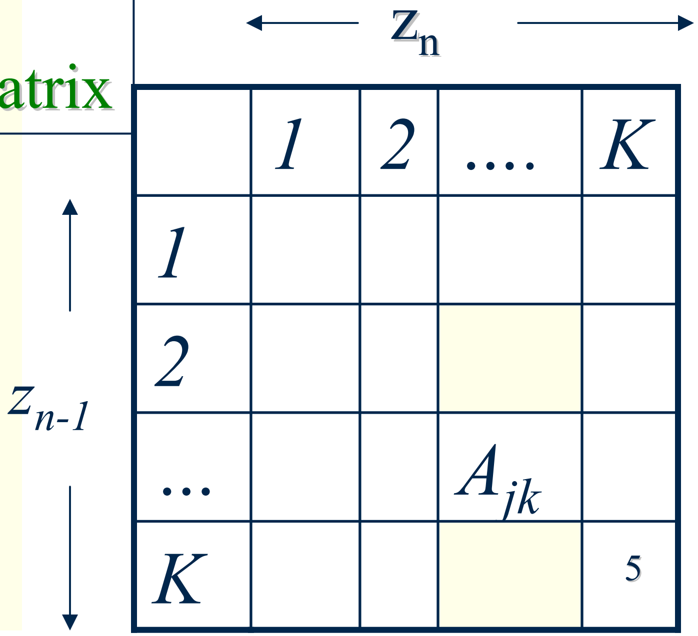
State of z_n

k	1	2	.	K
z_{nk}	0	1	.	0

State of z_{n-1}

j	1	2	.	K
$z_{n-1,j}$	1	0	.	0

A matrix



	1	2	....	K
1				
2				
...			A_{jk}	
K				

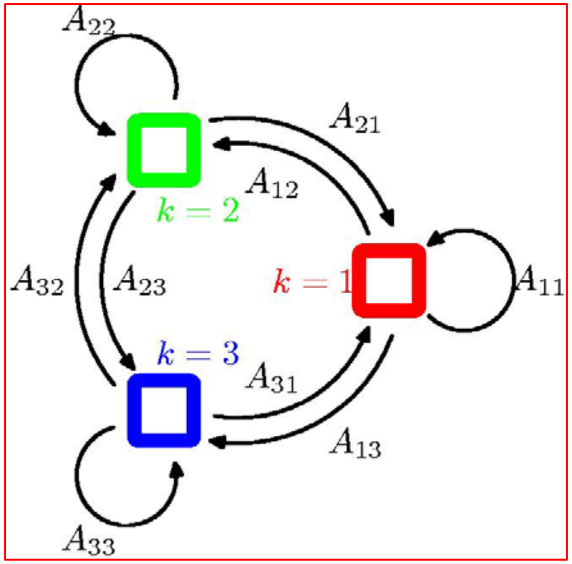
Transition Probabilities

Example with 3-state latent variable

	$z_n=1$ $z_{n1}=1$ $z_{n2}=0$ $z_{n3}=0$	$z_n=2$ $z_{n1}=0$ $z_{n2}=1$ $z_{n3}=0$	$z_n=3$ $z_{n1}=0$ $z_{n2}=0$ $z_{n3}=1$
$z_{n-1}=1$ $z_{n-1,1}=1$ $z_{n-1,2}=0$ $z_{n-1,3}=0$	A_{11}	A_{12}	A_{13}
$z_{n-1}=2$ $z_{n-1,1}=0$ $z_{n-1,2}=1$ $z_{n-1,3}=0$	A_{21}	A_{22}	A_{23}
$z_{n-3}= 3$ $z_{n-1,1}=0$ $z_{n-1,2}=0$ $z_{n-1,3}=1$	A_{31}	A_{32}	A_{33}

$$A_{11}+A_{12}+A_{13}=1$$

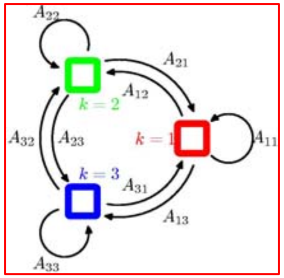
State Transition Diagram



- Not a graphical model since nodes are not separate variables but states of a single variable
- Here ***K*=3**

Conditional Probabilities

- Transition probabilities A_{jk} represent state-to-state probabilities for each variable
- Conditional probabilities are variable-to-variable probabilities
  - can be written in terms of transition probabilities as



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

  - Note that exponent $z_{n-1,j}\, z_{n,k}$ is a product that evaluates to 0 or 1
  - Hence the overall product will evaluate to a single A_{jk} for each setting of values of z_n and z_{n-1}
    - E.g., $z_{n-1}=2$ and $z_n=3$ will result in only $z_{n-1,2}=1$ and $z_{n,3}=1$. Thus $p(z_n=3|z_{n-1}=2)=A_{23}$
- A is a global HMM parameter

Initial Variable Probabilities

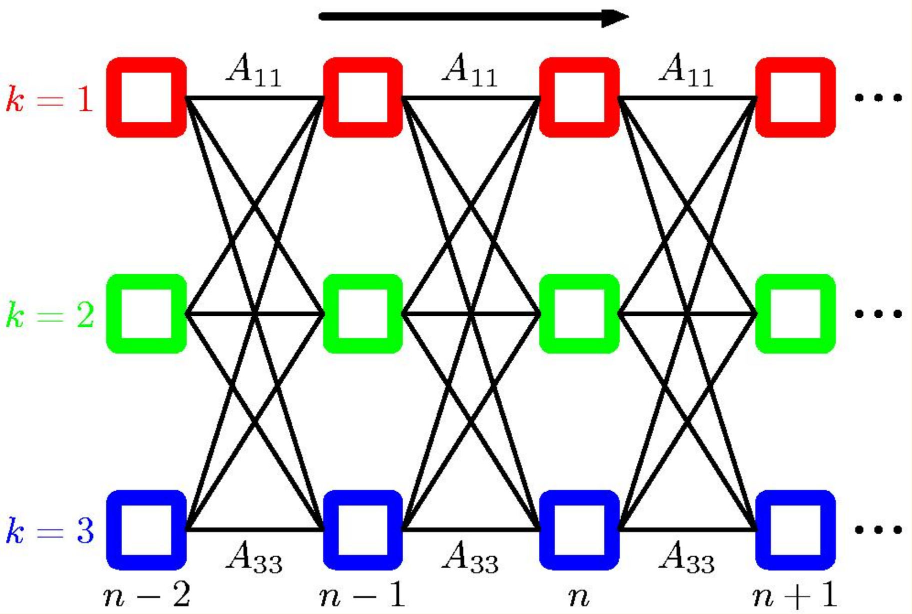
- Initial latent node z_1 is a special case without parent node
- Represented by vector of probabilities π with elements $\pi_k = p(z_{1k} = 1)$ so that

$$p(z_1 \mid \pi) = \prod_{k=1}^{K} \pi_k^{z_{1,k}} \text{ where } \Sigma_k \pi_k = 1$$

- Note that π is an HMM parameter
  - representing probabilities of each state for the first variable

Lattice or Trellis Diagram

- State transition diagram unfolded over time



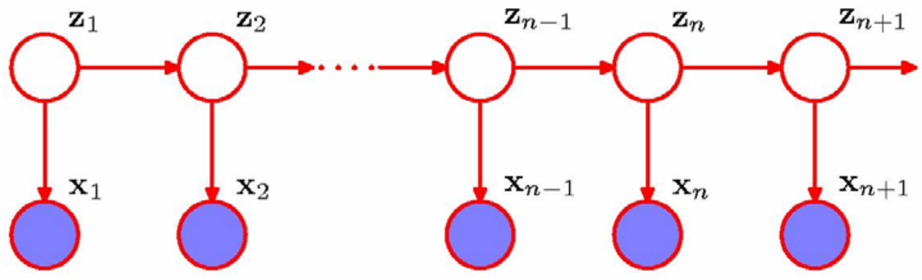
- Representation of latent variable states
- Each column corresponds to one of latent variables z_n

Emission Probabilities $p(\mathrm{x}_n|\mathrm{z}_n)$

- We have so far only specified $p(\mathrm{z}_n|\mathrm{z}_{n-1})$ by means of transition probabilities
- Need to specify probabilities $p(\mathrm{x}_n|\mathrm{z}_n, \phi)$ to complete the model, where ϕ are parameters
- These can be continuous or discrete
- Because x_n is observed and z_n is discrete $p(\mathrm{x}_n|\mathrm{z}_n, \phi)$ consists of a table of K numbers corresponding to K states of z_n
  - Analogous to class-conditional probabilities
- Can be represented as

$$p(\mathrm{x}_n \mid \mathrm{z}_n, \varphi) = \prod_{k=1}^{K} p(\mathrm{x}_n \mid \varphi_k)^{z_{nk}}$$

3. HMM Parameters



- We have defined three types of HMM parameters: $\theta = (\pi, A, \phi)$
  1. *Initial Probabilities of first latent variable:*

     π is a vector of K probabilities of the states for latent variable z_1
  2. *Transition Probabilities (State-to-state for any latent variable):*

     A is a K x K matrix of transition probabilities A_{ij}
  3. *Emission Probabilities (Observations conditioned on latent):*

     ϕ are parameters of conditional distribution $p(x_k|z_k)$
- A and π parameters are often initialized uniformly
- Initialization of ϕ depends on form of distribution

Joint Distribution over Latent and Observed variables

- Joint can be expressed in terms of parameters:

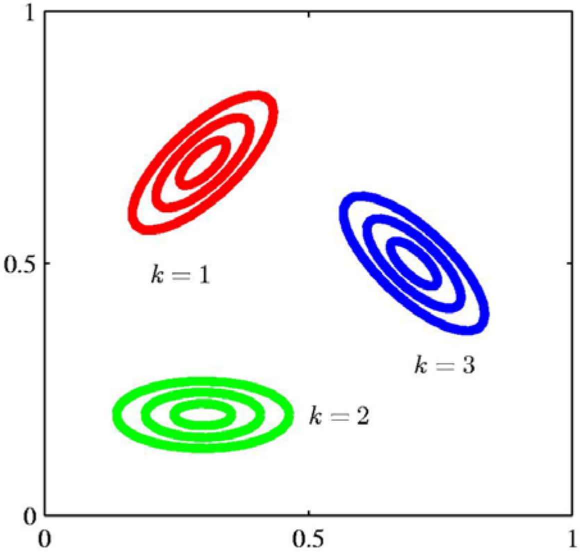
$$p(X,Z\,|\,\theta) = p(\mathrm{z}_1\,|\,\pi)\left[\prod_{n=2}^{N} p(\mathrm{z}_n\,|\,\mathrm{z}_{n-1},A)\right]\prod_{m=1}^{N} p(\mathrm{x}_m\,|\,\mathrm{z}_m,\varphi)$$

$$\text{where}\quad X=\{\mathrm{x}_1,..\mathrm{x}_N\},Z=\{\mathrm{z}_1,..\mathrm{z}_N\},\theta=\{\pi,A,\varphi\}$$

- Most discussion of HMM is independent of emission probabilities
  - Tractable for discrete tables, Gaussian, GMMs

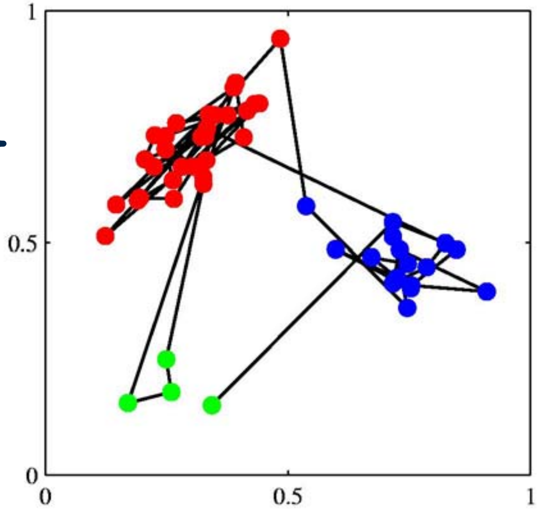
4. Generative View of HMM

- Sampling from an HMM
- HMM with 3-state latent variable z
  - Gaussian emission model $p(\mathrm{x}|\mathrm{z})$
  - Contours of constant density of emission distribution shown for each state
  - Two-dimensional x



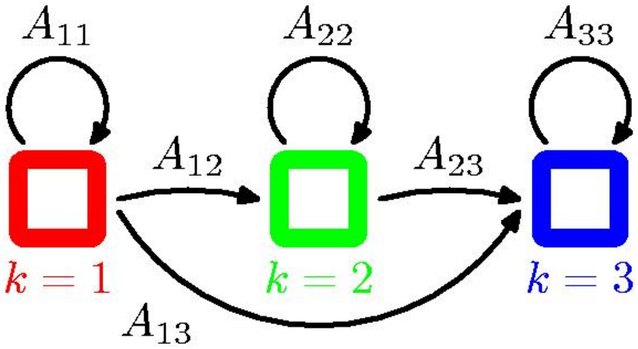
- 50 data points generated from HMM
- Lines show successive observations

- Transition probabilities fixed so that
  - 5% probability of making transition
  - 90% of remaining in same

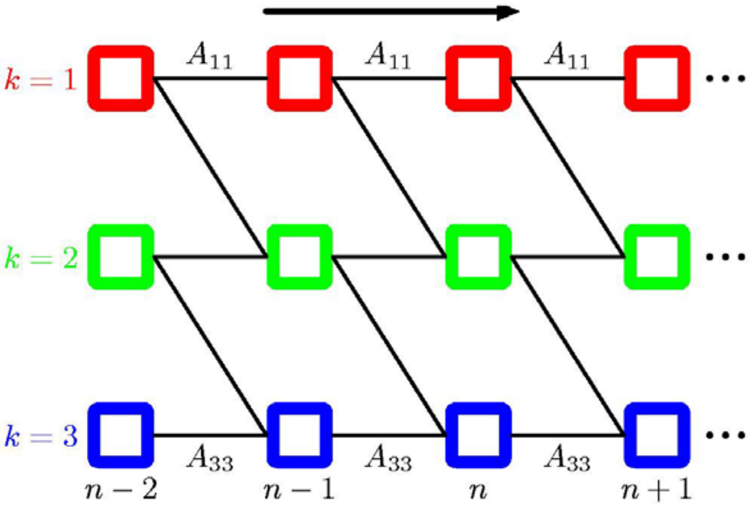


Left-to-Right HMM

- Setting elements of $A_{jk}=0$ if $k < j$

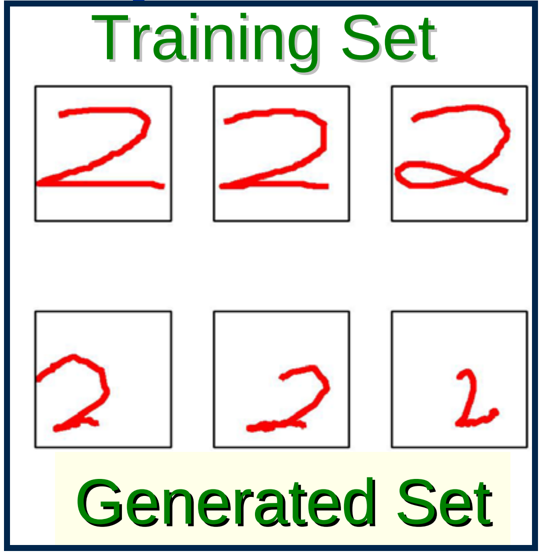


- Corresponding lattice diagram



Left-to-Right Applied Generatively to Digits

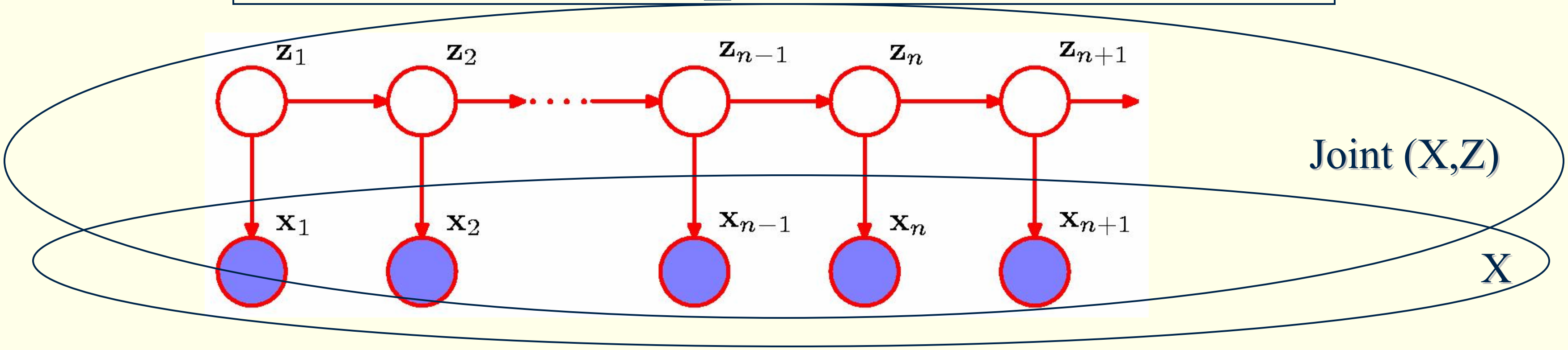
- Examples of on-line handwritten 2’s
- x is a sequence of pen coordinates
- There are16 states for z, or $K=16$
- Each state can generate a line segment of fixed length in one of 16 angles
  - Emission probabilities: 16 x 16 table
- Transition probabilities set to zero except for those that keep state index k the same or increment by one
- Parameters optimized by 25 EM iterations
- Trained on 45 digits
- Generative model is quite poor
  - Since generated don’t look like training
  - If classification is goal, can do better by using a discriminative HMM



5. Determining HMM Parameters

- Given data set $X=\{x_1,..x_n\}$ we can determine HMM parameters $\theta=\{\pi, A, \phi\}$ using maximum likelihood
- Likelihood function obtained from joint distribution by marginalizing over latent variables $Z=\{z_1,..z_n\}$

$$\boxed{p(X|\theta) \;=\; \Sigma_Z\, p(X,Z|\theta)}$$



Computational issues for Parameters

$$\boxed{p(X|\theta) = \Sigma_Z \, p(X,Z|\theta)}$$

- Computational difficulties
  - Joint distribution $p(X,Z|\theta)$ does not factorize over n, in contrast to mixture model
  - Z has exponential number of terms corresponding to trellis
- Solution
  - Use conditional independence properties to reorder summations
  - Use EM instead to maximize log-likelihood function of joint $\ln p(X,Z|\theta)$

Efficient framework for maximizing the likelihood function in HMMs

EM for MLE in HMM

1. Start with *initial selection for model parameters* θ^{old}
2. In E step take these parameter values and find *posterior distribution of latent variables* $p(Z|X, \theta^{old})$

   Use this posterior distribution to evaluate *expectation of the logarithm of the complete-data likelihood function* $\ln p(X,Z\,|\,\theta)$

   Which can be written as

$$Q(\theta,\theta^{old}) = \sum_{Z} \underline{p(Z\,|\,X,\theta^{old})} \ln p(X,Z\,|\,\theta)$$

   underlined portion independent of θ is evaluated

3. In M-Step maximize Q w.r.t. θ

Expansion of Q

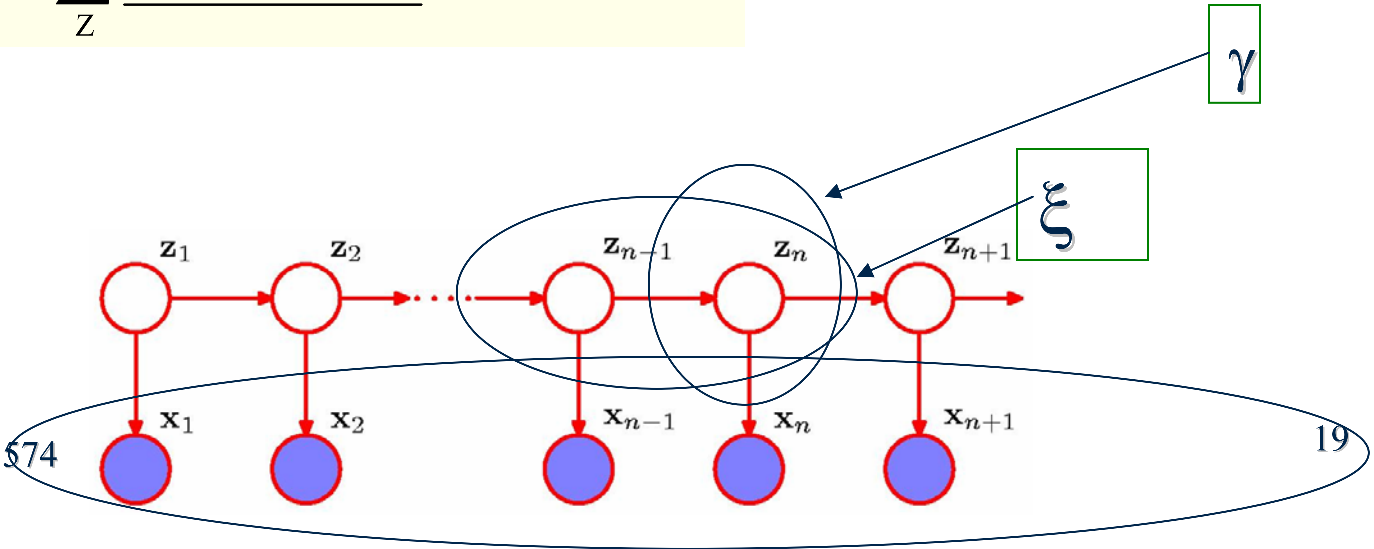
- Introduce notation

  $\gamma(z_n) = p(z_n|X,\theta^{old})$: Marginal posterior distribution of latent variable z_n

  $\xi(z_{n-1},z_n) = p(z_{n-1},z_n|X,\theta^{old})$: Joint posterior of two successive latent variables

- We will be re-expressing Q in terms of γ and ξ

$$Q(\theta,\theta^{old}) = \sum_Z \underline{p(Z|X,\theta^{old})} \ln p(X,Z|\theta)$$



Detail of γ and ξ

For each value of n we can store

$\gamma(z_n)$ using K non-negative numbers that sum to unity

$\xi(z_{n-1}, z_n)$ using a K x K matrix whose elements also sum to unity

- Using notation

  $\gamma(z_{nk})$ denotes conditional probability of $z_{nk}{}^{=1}$

  Similar notation for $\xi(z_{n-1,j}, z_{nk})$

- Because the expectation of a binary random variable is the probability that it takes value 1

$$\gamma(z_{nk}) = E[z_{nk}] = \Sigma_z \gamma(z) z_{nk}$$

$$\xi(z_{n-1,j}, z_{nk}) = E[z_{n-1,j}, z_{nk}] = \Sigma_z \gamma(z)\, z_{n-1,j} z_{nk}$$

Expansion of Q

- We begin with

$$Q(\theta,\theta^{old}) = \sum_{Z} \underline{p(\mathrm{Z} \mid X,\theta^{old})} \ln p(X,Z \mid \theta)$$

- Substitute

$$p(X,Z \mid \theta) = p(\mathrm{z}_1 \mid \pi)\left[\prod_{n=2}^{N} p(\mathrm{z}_n \mid \mathrm{z}_{n-1}, A)\right]\prod_{m=1}^{N} p(\mathrm{x}_m \mid \mathrm{z}_m, \varphi)$$

- And use definitions of γ and ξ to get:

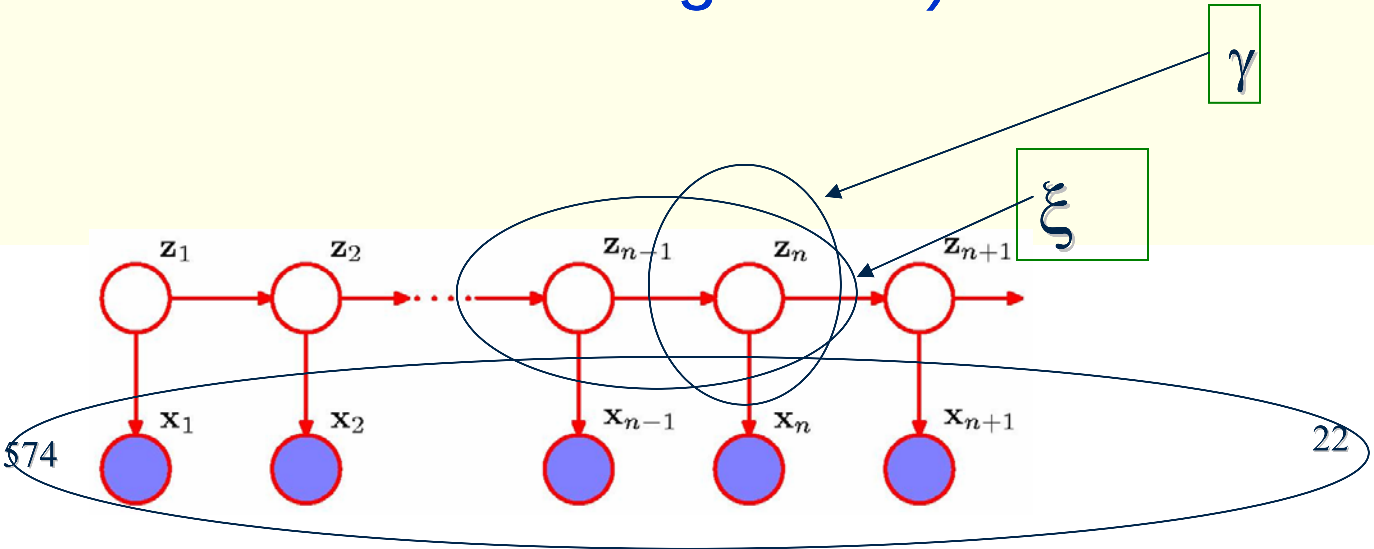
$$\boxed{\begin{aligned} Q(\theta,\theta^{old}) = \sum_{k=1}^{K}\gamma(z_{1k})\ln \pi_k + \sum_{n=2}^{N}\sum_{j=1}^{K}\sum_{k=1}^{K}\xi(z_{n\text{-}1,j},z_{nk})\ln A_{jk} \\ + \sum_{n=1}^{N}\sum_{k=1}^{K}\gamma(z_{nk})\ln p(x_n \mid \phi_k) \end{aligned}}$$

E-Step

$$Q(\theta,\theta^{old}) = \sum_{k=1}^{K} \gamma(z_{1k}) \ln \pi_k + \sum_{n=2}^{N} \sum_{j=1}^{K} \sum_{k=1}^{K} \xi(z_{n\text{-}1,j}, z_{nk}) \ln A_{jk}$$

$$+ \sum_{n=1}^{N} \sum_{k=1}^{K} \gamma(z_{nk}) \ln p(x_n \mid \phi_k)$$

- Goal of E step is to evaluate $\gamma(z_n)$ and $\xi(z_{n-1},z_n)$ efficiently (Forward-Backward Algorithm)



M-Step

- Maximize $Q(\theta, \theta^{old})$ with respect to parameters $\theta = \{\pi, A, \phi\}$
  - Treat $\gamma(z_n)$ and $\xi(z_{n-1}, z_n)$ as constant
- Maximization w.r.t. π and A
  - easily achieved (using Lagrangian multipliers)

$$\pi_k = \frac{\gamma(z_{1k})}{\sum_{j=1}^{K} \gamma(z_{1j})} \qquad A_{jk} = \frac{\sum_{n=2}^{N} \xi(z_{n-1,j}, z_{nk})}{\sum_{l=1}^{K} \sum_{n=2}^{N} \xi(z_{n-1,j}, z_{nl})}$$

- Maximization w.r.t. ϕ_k
  - Only last term of Q depends on ϕ_k $\longrightarrow$ $\boxed{\sum_{n=1}^{N} \sum_{k=1}^{K} \gamma(z_{nk}) \ln p(x_n \mid \phi_k)}$
  - Same form as in mixture distribution for i.i.d.

M-step for Gaussian emission

- Maximization of $Q(\theta, \theta^{old})$ wrt ϕ_k
- Gaussian Emission Densities

$$p(\mathrm{x}|\phi_k) \sim N(\mathrm{x}|\mu_k, \Sigma_k)$$

- Solution:

$$\boxed{\mu_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk})\mathrm{x}_n}{\sum_{n=1}^{N} \gamma(z_{nk})}}$$

$$\boxed{\Sigma_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk})(\mathrm{x}_n - \mu_k)(\mathrm{x}_n - \mu_k)^T}{\sum_{n=1}^{N} \gamma(z_{nk})}}$$

M-Step for Multinomial Observed

- Conditional Distribution of Observations have the form

$$p(\mathrm{x} \mid \mathrm{z}) = \prod_{i=1}^{D} \prod_{k=1}^{K} \mu_{ik}^{x_i z_k}$$

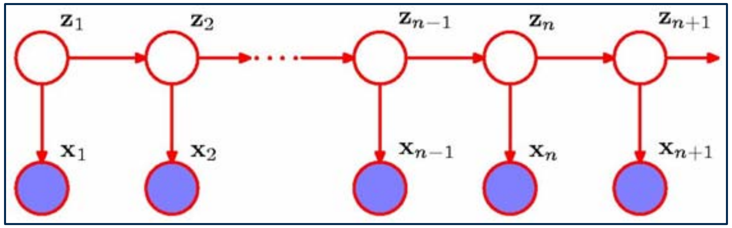
- M-Step equations:

$$\mu_{ik} = \frac{\sum_{n=1}^{N} \gamma(z_{nk}) x_{ni}}{\sum_{n=1}^{N} \gamma(z_{nk})}$$

- Analogous result holds for Bernoulli observed variables

6. Forward-Backward Algorithm

- E step: efficient procedure to evaluate $\gamma(z_n)$ and $\xi(z_{n-1},z_n)$
- Graph of HMM, a tree→



  - Implies that posterior distribution of latent variables can be obtained efficiently using message passing algorithm
- In HMM it is called *forward-backward* algorithm or *Baum-Welch Algorithm*
- Several variants lead to exact marginals
  - Method called *alpha-beta* discussed here

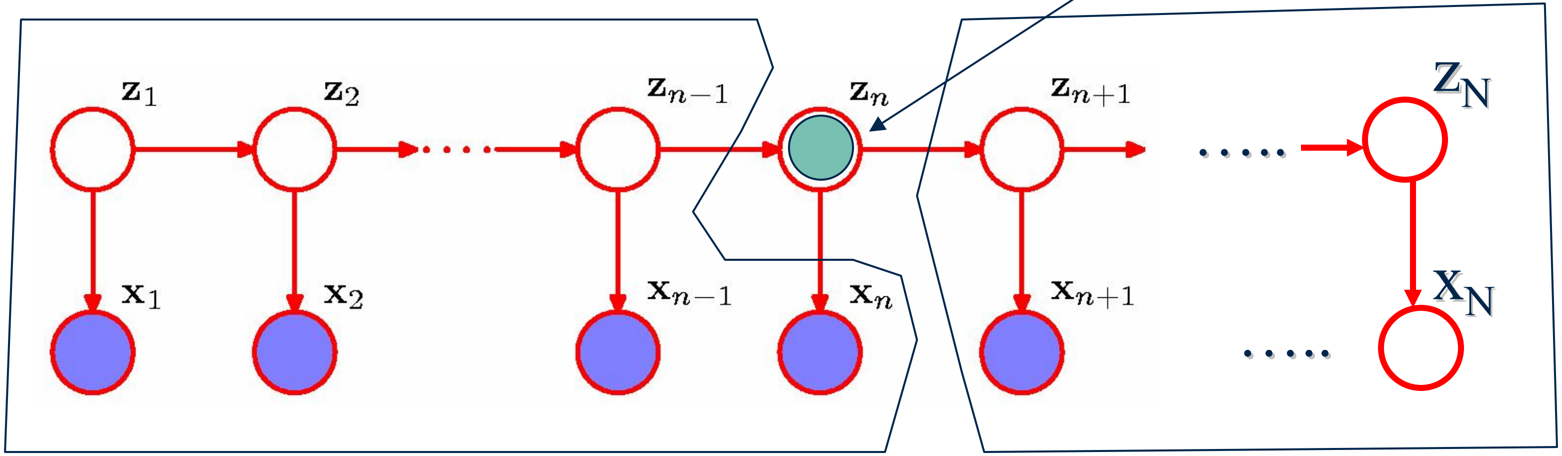
Derivation of Forward-Backward

- Several conditional-independences (A-H) hold

A. $$\boxed{p(X|z_n) = p(x_1,..x_n|z_n)\, p(x_{n+1},..x_N|z_n)}$$

- Proved using d-separation:

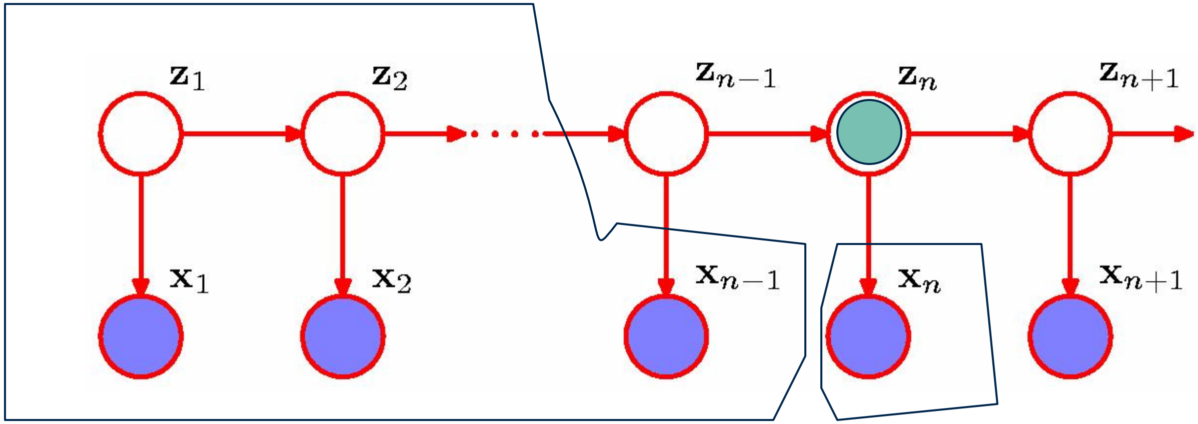
Path from x_1 to x_{n-1} passes through z_n which is observed.
Path is head-to-tail. Thus $(x_1,..x_{n-1}) \perp\!\!\!\perp x_n \mid z_n$
Similarly $(x_1,..x_{n-1},x_n) \perp\!\!\!\perp x_{n+1},..x_N \mid z_n$



Conditional independence B

- Since $(x_1,..x_{n-1}) \perp\!\!\!\perp x_n \mid z_n$ we have

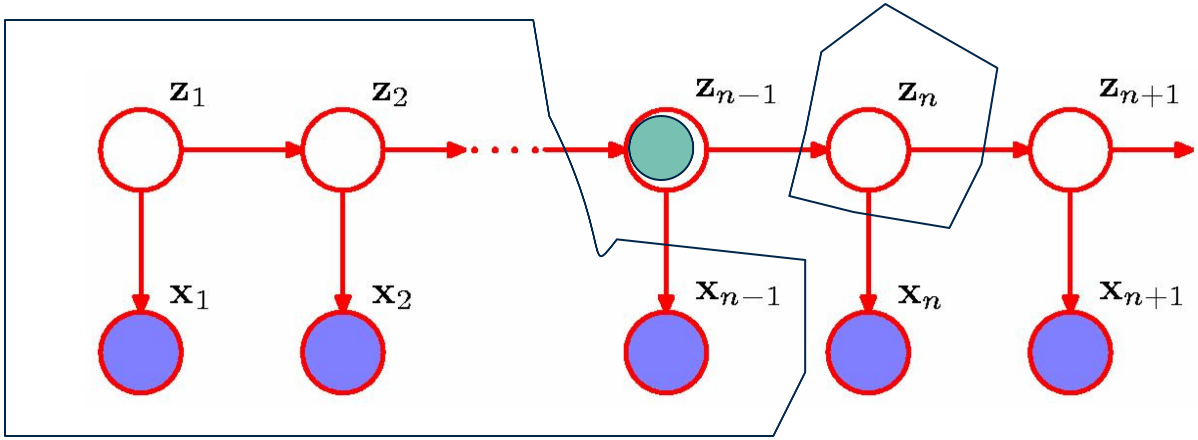
B. $\boxed{p(x_1,..x_{n-1}|x_n, z_n) = p(x_1,..x_{n-1}|z_n)}$



Conditional independence C

- Since $(x_1,..x_{n-1}) \perp\!\!\!\perp z_n \mid z_{n-1}$

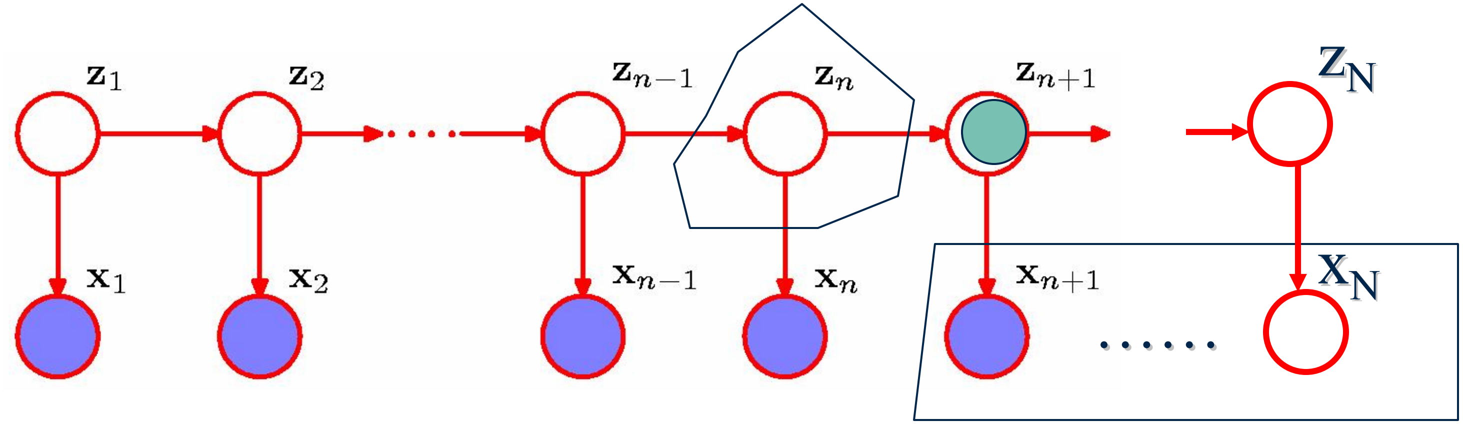
C. $\boxed{p(x_1,..x_{n-1}|z_{n-1},z_n) = p\,(\,x_1,..x_{n-1}|z_{n-1}\,)}$



Conditional independence D

- Since $(x_{n+1},..x_N) \perp\!\!\!\perp z_n \mid z_{n+1}$

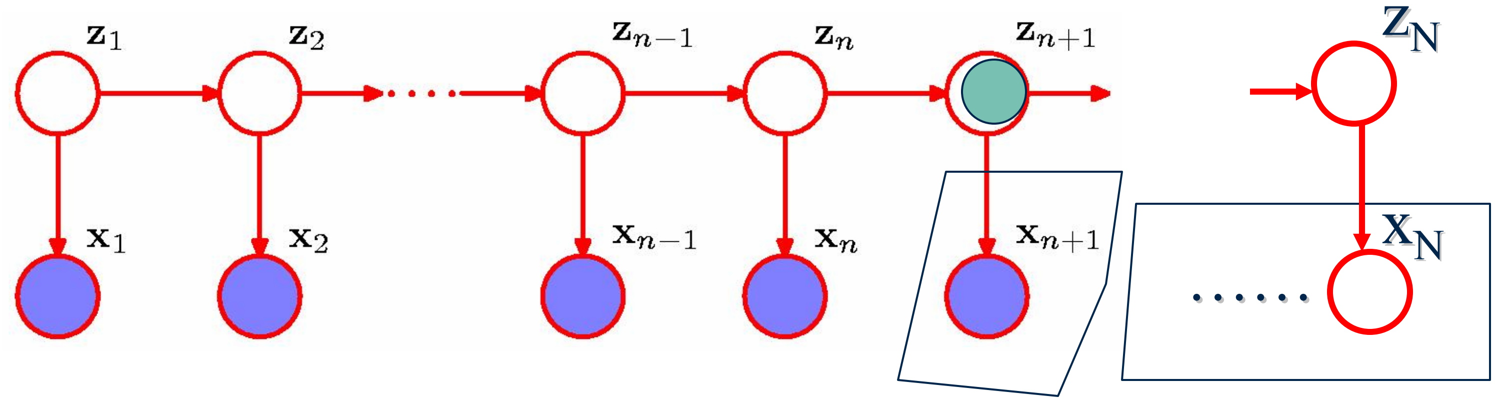
D. $\boxed{p(x_{n+1},..x_N|z_n,z_{n+1})=p\,(\,x_1,..x_N|z_{n+1}\,)}$



Conditional independence E

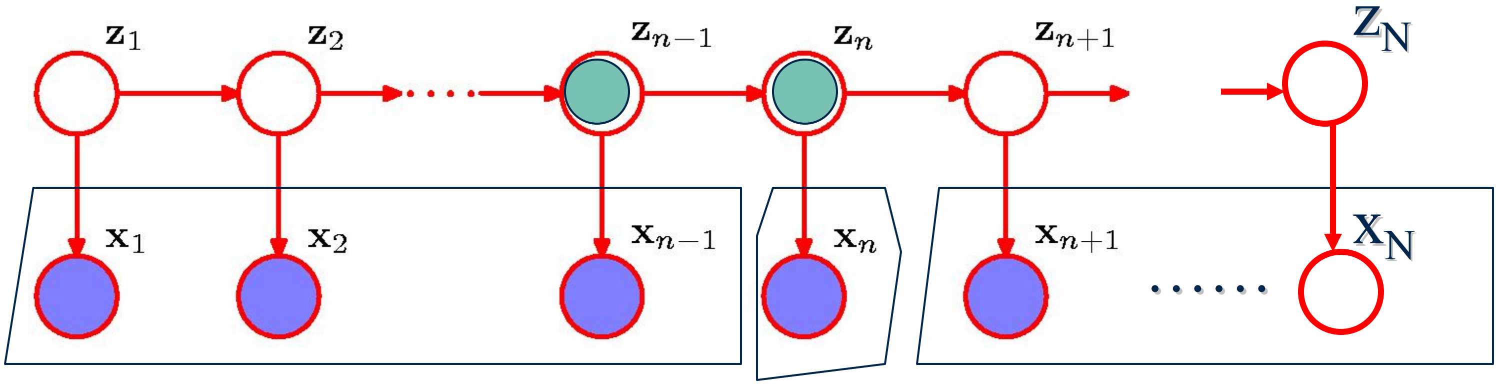
- Since $(x_{n+2},..x_N) \perp\!\!\!\perp z_n \mid z_{n+1}$

E. $\boxed{p(x_{n+2},..x_N|z_{n+1},x_{n+1}) = p\,(\,x_{n+2},..x_N|z_{n+1}\,)}$



Conditional independence F

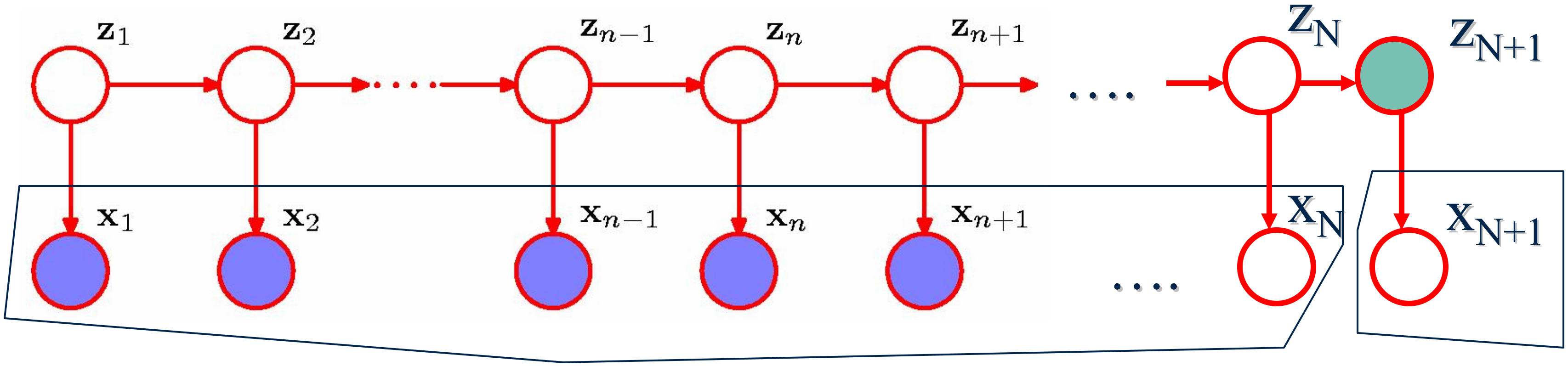
F. $p(X|z_{n-1},z_n) = p(x_1,..x_{n-1}|z_{n-1})p(x_n|z_n)$
$p(x_{n+1},..x_N|z_n)$



Conditional independence G

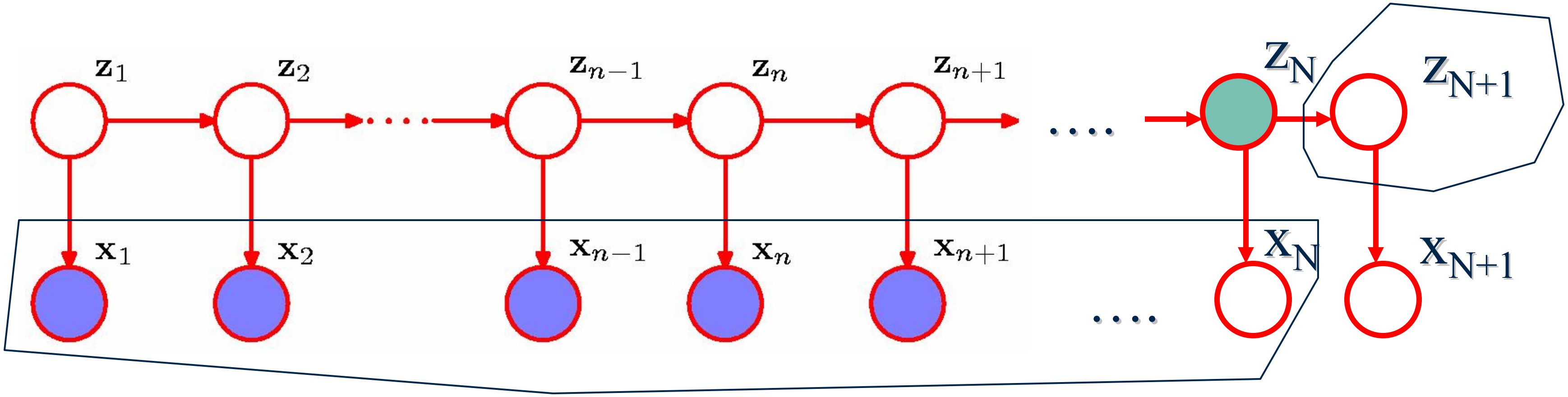
Since $(x_1,..x_N) \perp\!\!\!\perp x_{N+1} \mid z_{N+1}$

G. $p(x_{N+1}|X,z_{N+1})=p(x_{N+1}|z_{N+1})$



Conditional independence H

H. $p(z_{N+1}|z_N, X) = p(z_{N+1}|z_N)$



Evaluation of $\gamma(z_n)$

- Recall that this is to efficiently compute the E step of estimating parameters of HMM

  $\gamma(z_n) = p(z_n|X,\theta^{old})$: Marginal posterior distribution of latent variable z_n

- We are interested in finding posterior distribution $p(z_n|x_1,..x_N)$
- This is a vector of length K whose entries correspond to expected values of z_{nk}

Introducing α and β

- Using Bayes theorem $\gamma(\mathrm{z}_n) = p(z_n \mid \mathrm{X}) = \dfrac{p(\mathrm{X} \mid z_n)p(z_n)}{p(\mathrm{X})}$
- Using conditional independence A

$$\gamma(\mathrm{z}_n) = \frac{p(\mathrm{x}_1,..\mathrm{x}_n \mid \mathrm{z}_n)p(\mathrm{x}_{n+1},..\mathrm{x}_N \mid \mathrm{z}_n)p(\mathrm{z}_n)}{p(\mathrm{X})}$$

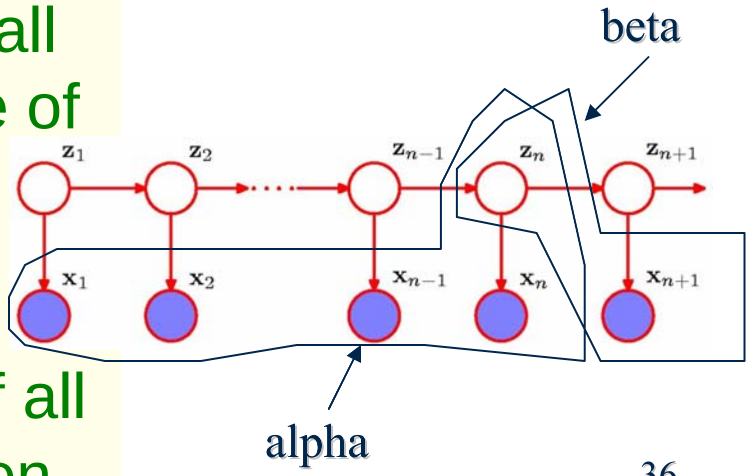
$$= \frac{p(\mathrm{x}_1,..\mathrm{x}_n,\mathrm{z}_n)p(\mathrm{x}_{n+1},..\mathrm{x}_N \mid \mathrm{z}_n)}{p(\mathrm{X})} = \boxed{\frac{\alpha(\mathrm{z}_n)\beta(\mathrm{z}_n)}{p(\mathrm{X})}}$$

- where $\boxed{\alpha(\mathrm{z}_n) \equiv p(\mathrm{x}_1,..,\mathrm{x}_n,\mathrm{z}_n)}$

  which is the probability of observing all given data up to time n and the value of z_n

$$\boxed{\beta(\mathrm{z}_n) \equiv p(\mathrm{x}_{n+1},..,\mathrm{x}_N \mid \mathrm{z}_n)}$$

  which is the conditional probability of all future data from time $n+1$ up to N given the value of z_n



Recursion Relation for α

$$
\begin{aligned}
\alpha(\mathrm{z}_n) &= p(\mathrm{x}_1,..,\mathrm{x}_n,\mathrm{z}_n) \\
&= \underline{p(\mathrm{x}_1,..,\mathrm{x}_n \mid \mathrm{z}_n)}p(\mathrm{z}_n) \text{ by Bayes rule} \\
&= \underline{p(\mathrm{x}_n \mid \mathrm{z}_n)\,p(\mathrm{x}_1,..,\mathrm{x}_{n-1} \mid \mathrm{z}_n)}p(\mathrm{z}_n) \text{ by conditional independence B} \\
&= p(\mathrm{x}_n \mid \mathrm{z}_n)\,p(\mathrm{x}_1,..,\mathrm{x}_{n-1},\mathrm{z}_n) \text{ by Bayes rule} \\
&= p(\mathrm{x}_n \mid \mathrm{z}_n)\sum_{\mathrm{z}_{n-1}} p(\mathrm{x}_1,..,\mathrm{x}_{n-1},\mathrm{z}_{n-1},\mathrm{z}_n) \text{ by Sum Rule} \\
&= p(\mathrm{x}_n \mid \mathrm{z}_n)\sum_{\mathrm{z}_{n-1}} p(\mathrm{x}_1,..,\mathrm{x}_{n-1},\mathrm{z}_n \mid \mathrm{z}_{n-1})\,p(\mathrm{z}_{n-1}) \text{ by Bayes rule} \\
&= p(\mathrm{x}_n \mid \mathrm{z}_n)\sum_{\mathrm{z}_{n-1}} p(\mathrm{x}_1,..,\mathrm{x}_{n-1} \mid \mathrm{z}_{n-1})\,p(\mathrm{z}_n \mid \mathrm{z}_{n-1})p(\mathrm{z}_{n-1}) \text{ by cond. ind. C} \\
&= p(\mathrm{x}_n \mid \mathrm{z}_n)\sum_{\mathrm{z}_{n-1}} p(\mathrm{x}_1,..,\mathrm{x}_{n-1},\mathrm{z}_{n-1})\,p(\mathrm{z}_n \mid \mathrm{z}_{n-1}) \text{ by Bayes rule} \\
&= p(\mathrm{x}_n \mid \mathrm{z}_n)\sum_{\mathrm{z}_{n-1}} \alpha(\mathrm{z}_{n-1})\,p(\mathrm{z}_n \mid \mathrm{z}_{n-1}) \text{ by definition of } \alpha
\end{aligned}
$$

Forward Recursion for α Evaluation

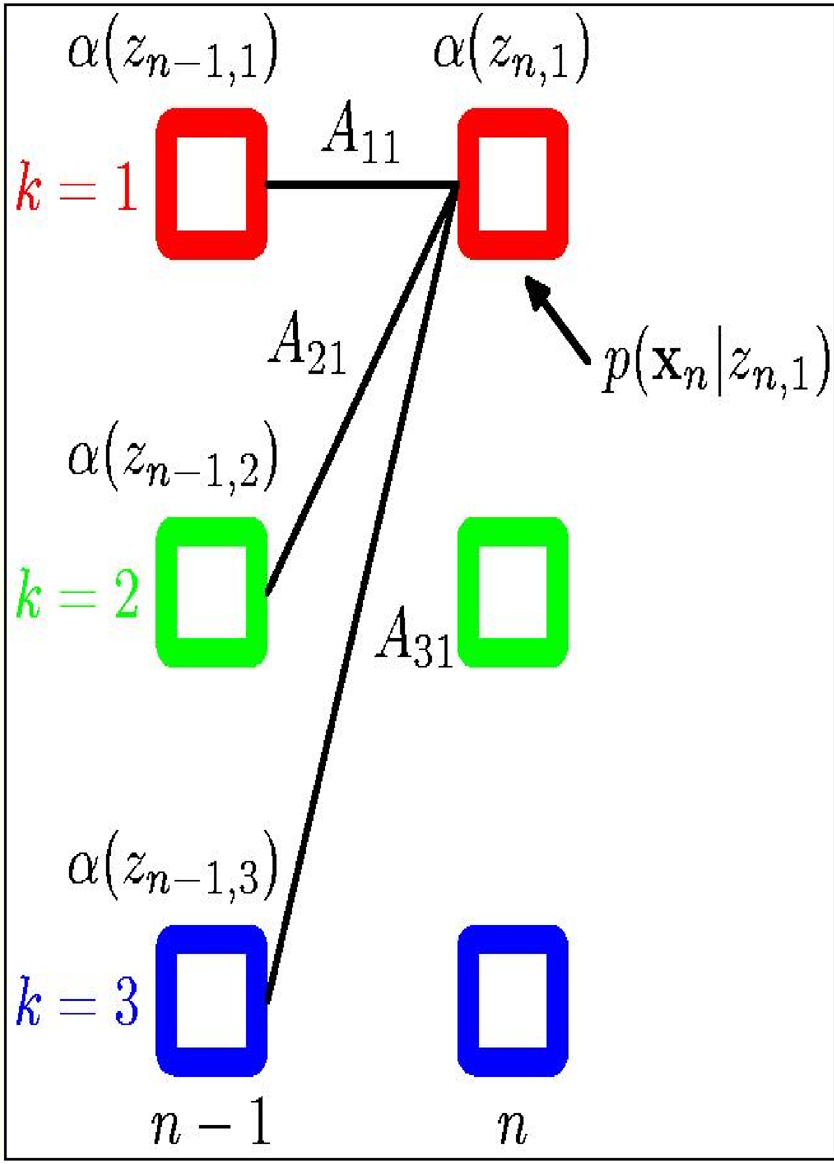
- Recursion Relation is

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

- There are *K* terms in the summation
  - Has to be evaluated for each of *K* values of z_n
  - Each step of recursion is $O(K^2)$
- Initial condition is

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

- Overall cost for the chain in $O(K^2N)$



Recursion Relation for β

$$\beta(\mathrm{z}_n) = p(\mathrm{x}_{n+1},..,\mathrm{x}_N \mid \mathrm{z}_n)$$

$$= \sum_{z_{n+1}} p(\mathrm{x}_{n+1},..,\mathrm{x}_N, \mathrm{z}_{n+1} \mid \mathrm{z}_n) \text{ by Sum Rule}$$

$$= \sum_{z_{n+1}} p(\mathrm{x}_{n+1},..,\mathrm{x}_N \mid \mathrm{z}_n,\mathrm{z}_{n+1}) p(\mathrm{z}_{n+1}|\mathrm{z}_n) \text{ by Bayes rule}$$

$$= \sum_{z_{n+1}} p(\mathrm{x}_{n+1},..,\mathrm{x}_N \mid \mathrm{z}_{n+1}) p(\mathrm{z}_{n+1}|\mathrm{z}_n) \text{ by Cond ind. D}$$

$$= \sum_{z_{n+1}} p(\mathrm{x}_{n+2},..,\mathrm{x}_N \mid \mathrm{z}_{n+1}) p(\mathrm{x}_{n+1}|\mathrm{z}_{n+1}) p(\mathrm{z}_{n+1}|\mathrm{z}_n) \text{ by Cond. ind E}$$

$$= \sum_{z_{n+1}} \beta(\mathrm{z}_{n+1}) p(\mathrm{x}_{n+1}|\mathrm{z}_{n+1}) p(\mathrm{z}_{n+1}|\mathrm{z}_n) \text{ by definition of } \beta$$

Backward Recursion for β

- Backward message passing

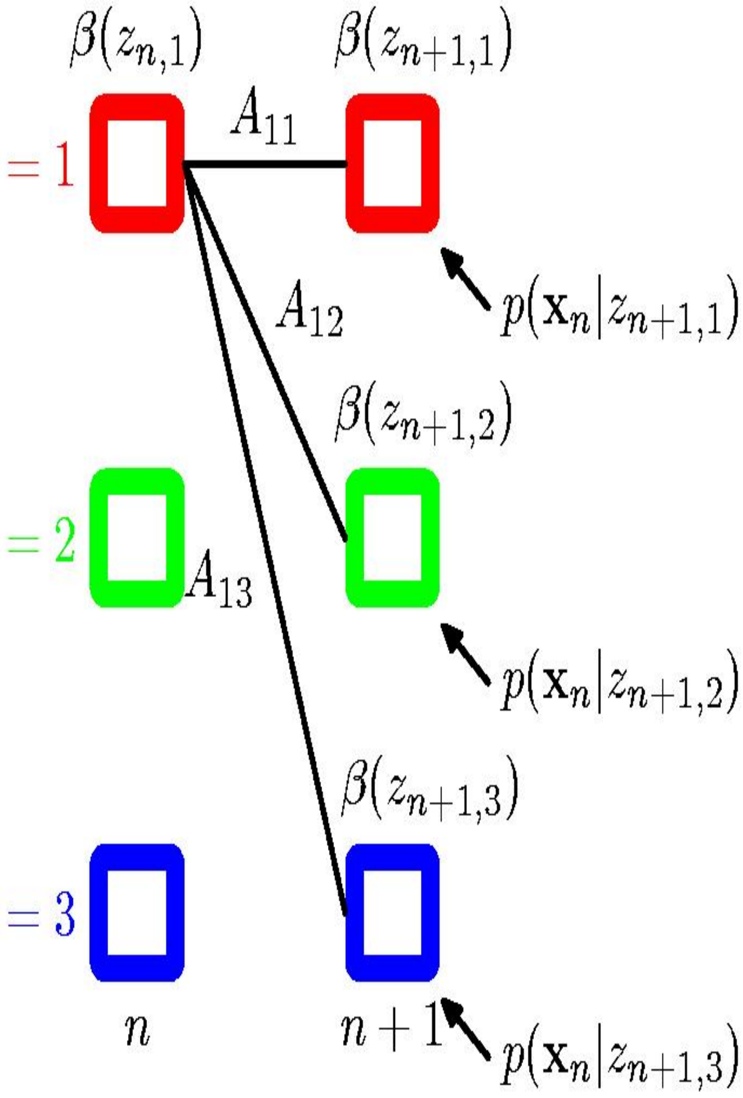
$$\beta(z_n) = \sum_{z_{n+1}} \beta(z_{n+1}) p(x_{n+1}|z_n) p(z_{n+1}|z_n)$$

- Evaluates $\beta(z_n)$ in terms of $\beta(z_{n+1})$

- Starting condition for recursion is

$$p(z_N \mid X) = \frac{p(X, z_N)\beta(z_N)}{p(X)}$$

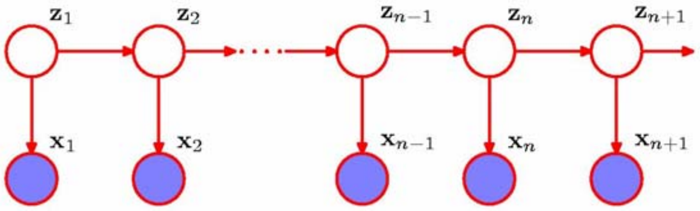
- Is correct provided we set $\beta(z_N) = 1$ for all settings of z_N
  - This is the initial condition for backward computation



M step Equations

- In the M-step equations $p(\mathrm{x})$ will cancel out

$$\boxed{\mu_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk})\mathrm{x}_n}{\sum_{n=1}^{N} \gamma(z_{nk})}}$$



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

Evaluation of Quantities $\xi(z_{n-1},z_n)$

- They correspond to the values of the conditional probabilities $p(z_{n-1},z_n|X)$ for each of the K x K settings for (z_{n-1},z_n)

$$\xi(\mathrm{z}_{\mathrm{n\text{-}1}},\mathrm{z}_{\mathrm{n}}) = \mathrm{p}(\mathrm{z}_{\mathrm{n\text{-}1}},\mathrm{z}_{\mathrm{n}} \mid \mathrm{X}) \;\; \text{by definition}$$

$$= \frac{p(X|z_{n\text{-}1},z_n)p(z_{n\text{-}1},z_n)}{p(X)} \;\; \text{by Bayes Rule}$$

$$= \frac{p(\mathrm{x}_1,..\mathrm{x}_{n-1}|z_{n\text{-}1})p(\mathrm{x}_n \mid \mathrm{z}_n)p(\mathrm{x}_{n+1},..,\mathrm{x}_N \mid z_n)p(\mathrm{z}_n \mid \mathrm{z}_{n\text{-}1})p(\mathrm{z}_{n\text{-}1})}{p(X)} \;\; \text{by cond ind F}$$

$$= \frac{\alpha(\mathrm{z}_{n-1})p(\mathrm{x}_n \mid \mathrm{z}_n)p(\mathrm{z}_n \mid \mathrm{z}_{n\text{-}1})\beta(\mathrm{z}_n)}{p(X)}$$

- Thus we calculate $\xi(z_{n-1},z_n)$ directly by using results of the α and β recursions

Summary of EM to train HMM

Step 1: Initialization

- Make an initial selection of parameters θ^{old} where $\theta = (\pi, A, \phi)$
  1. π is a vector of K probabilities of the states for latent variable z_1
  2. A is a K x K matrix of transition probabilities A_{ij}
  3. ϕ are parameters of conditional distribution $p(x_k|z_k)$
- A and π parameters are often initialized uniformly
- Initialization of ϕ depends on form of distribution
  - For Gaussian:
    - parameters μ_k initialized by applying K-means to the data, Σ_k corresponds to covariance matrix of cluster

Summary of EM to train HMM

Step 2: E Step

- Run both forward α recursion and backward β recursion
- Use results to evaluate $\gamma(z_n)$ and $\xi(z_{n-1},z_n)$ and the likelihood function

Step 3: M Step

- Use results of E step to find revised set of parameters θ^{new} using M-step equations

Alternate between E and M

until convergence of likelihood function

Values for $p(\mathrm{x}_n|\mathrm{z}_n)$

- In recursion relations, observations enter through conditional distributions $p(\mathrm{x}_\mathrm{n}|\mathrm{z}_\mathrm{n})$
- Recursions are independent of
  - Dimensionality of observed variables
  - Form of conditional distribution
    - So long as it can be computed for each of K possible states of z_n
- Since observed variables $\{\mathrm{x}_n\}$ are fixed they can be pre-computed at the start of the EM algorithm

7(a) Sequence Length: Using Multiple Short Sequences

- HMM can be trained effectively if length of sequence is sufficiently long
  - True of all maximum likelihood approaches
- Alternatively we can use multiple short sequences
  - Requires straightforward modification of HMM-EM algorithm
- Particularly important in left-to-right models
  - In given observation sequence, a given state transition for a non-diagonal element of A occurs only once

7(b). Predictive Distribution

- Observed data is $X=\{ x_1,..,x_N\}$
- Wish to predict x_{N+1}
- Application in financial forecasting

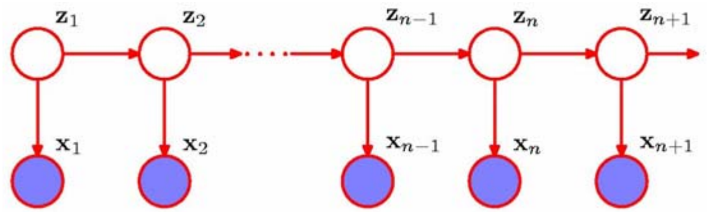
$$
\begin{aligned}
p(\mathrm{x}_{N+1} \mid X) &= \sum_{z_{N+1}} p(\mathrm{x}_{N+1}, \mathrm{z}_{N+1} \mid X) \\
&= \sum_{z_{N+1}} p(\mathrm{x}_{N+1} | \mathrm{z}_{N+1} \mid X) p(\mathrm{z}_{N+1} \mid X) \text{ by Product Rule} \\
&= \sum_{z_{N+1}} p(\mathrm{x}_{N+1} | \mathrm{z}_{N+1}) \sum_{z_N} p(\mathrm{z}_{N+1}, z_N \mid X) \text{ by Sum Rule} \\
&= \sum_{z_{N+1}} p(\mathrm{x}_{N+1} | \mathrm{z}_{N+1}) \sum_{z_N} p(\mathrm{z}_{N+1} \mid z_N) p(z_N \mid X) \text{ by conditional ind H} \\
&= \sum_{z_{N+1}} p(\mathrm{x}_{N+1} | \mathrm{z}_{N+1}) \sum_{z_N} p(\mathrm{z}_{N+1} \mid z_N) \frac{p(z_N, X)}{p(X)} \text{ by Bayes rule} \\
&= \frac{1}{p(X)} \sum_{z_{N+1}} p(\mathrm{x}_{N+1} | \mathrm{z}_{N+1}) \sum_{z_N} p(\mathrm{z}_{N+1} \mid z_N) \alpha(z_N) \text{ by definition of } \alpha
\end{aligned}
$$

- Can be evaluated by first running forward α recursion and summing over z_N and z_{N+1}
- Can be extended to subsequent predictions of x_{N+2}, after x_{N+1} is observed, using a fixed amount of storage

7(c). Sum-Product and HMM

- HMM graph is a tree and hence *sum-product* algorithm can be used to find local marginals for hidden variables
  - Equivalent to forward-backward algorithm
  - Sum-product provides a simple way to derive alpha-beta recursion formulae
- Transform directed graph to factor graph
  - Each variable has a node, small squares represent factors, undirected links connect factor nodes to variables used

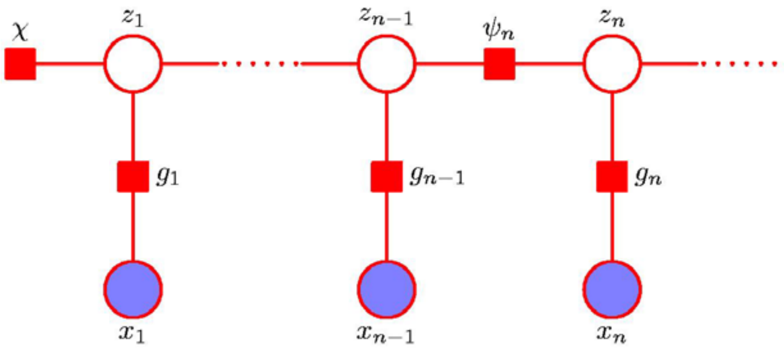
HMM Graph



Joint distribution

$$p(x_1,..x_N,z_1,..z_n)=p(z_1)\left[\prod_{n=2}^{N}p(z_n\mid z_{n-1})\right]\prod_{n=1}^{N}p(x_n\mid z_n)$$

Fragment of Factor Graph



Deriving alpha-beta from Sum-Product

- Begin with simplified form of factor graph
- Factors are given by

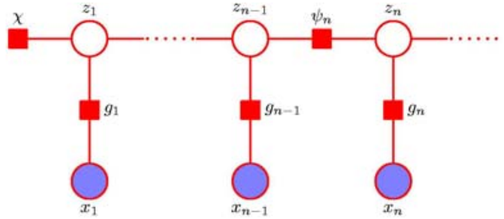
$$h(z_1) = p(z_1)p(x_1 \mid z_1)$$
$$f_n(z_{n-1}, z_n) = p(z_n \mid z_{n-1})p(x_n \mid z_n)$$

- Messages propagated are

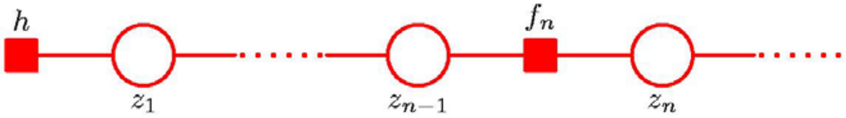
$$\mu_{z_{n-1} \to f_n}(z_{n-1}) = \mu_{f_{n-1} \to z_{n-1}}(z_{n-1})$$
$$\mu_{f_n \to z_n}(z_n) = \sum_{z_{n-1}} f_n(z_{n-1}, z_n)\mu_{z_{n-1} \to f_n}(z_{n-1})$$

- Can show that α recursion is computed
- Similarly starting with the root node β recursion is computed
- So also γ and ξ are derived

Fragment of Factor Graph



Simplified by absorbing emission probabilities into transition probability factors



Final Results

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

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

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

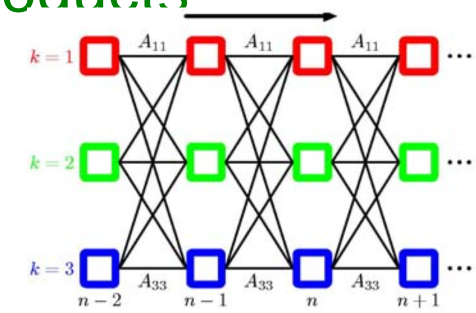
$$\xi(z_{n-1}, z_n) = \frac{\alpha(z_{n-1})p(x_n \mid z_n)p(z_n \mid z_{n-1})\beta(z_n)}{p(X)}$$

7(d). Scaling Factors

- Implementation issue for small probabilities
- At each step of recursion

$$\alpha(\mathrm{z}_n) = p(\mathrm{x}_n \mid \mathrm{z}_n)\sum_{z_{n-1}}\alpha(\mathrm{z}_{n-1})p(\mathrm{z}_n \mid \mathrm{z}_{n-1})$$

  - To obtain new value of $\alpha(z_n)$ from previous value $\alpha(z_{n-1})$ we multiply $p(z_n|z_{n-1})$ and $p(x_n|z_n)$
  - These probabilities are small and products will underflow
  - Logs don't help since we have sums of products



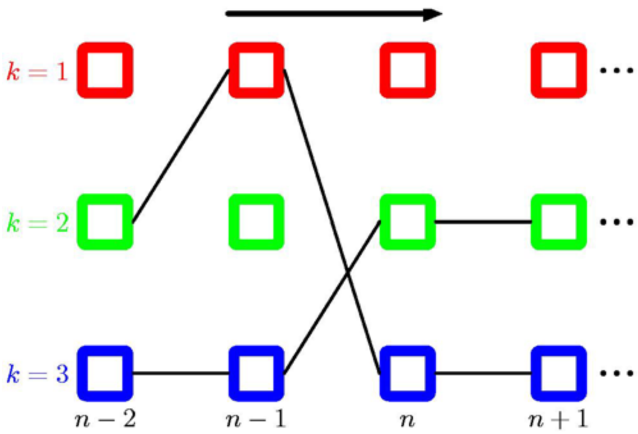
- Solution is rescaling
  - of $\alpha(z_n)$ and $\beta(z_n)$ whose values remain close to unity

7(e). The Viterbi Algorithm

- Finding most probable sequence of hidden states for a given sequence of observables
- In speech recognition: finding most probable phoneme sequence for a given series of acoustic observations
- Since graphical model of HMM is a tree, can be solved exactly using *max-sum* algorithm
  - Known as Viterbi algorithm in the context of HMM
  - Since max-sum works with log probabilities no need to work with re-scaled varaibles as with forward-backward

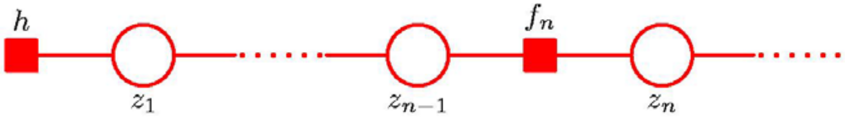
Viterbi Algorithm for HMM

- Fragment of HMM lattice showing two paths
- Number of possible paths grows exponentially with length of chain
- Viterbi searches space of paths efficiently
  - Finds most probable path with computational cost linear with length of chain



Deriving Viterbi from Max-Sum

- Start with simplified factor graph



- Treat variable z_N as root node, passing messages to root from leaf nodes
- Messages passed are

$$\mu_{z_n \to f_{n+1}}(z_n) = \mu_{f_n \to z_n}(z_n)$$

$$\mu_{f_{n+1} \to z_{n+1}}(z_{n+1}) = \max_{z_n} \left\{ \ln f_{n+1}(z_n, z_{n+1}) + \mu_{z_n \to f_{n+1}}(z_n) \right\}$$

Other Topics on Sequential Data

- Sequential Data and Markov Models:

  http://www.cedar.buffalo.edu/~srihari/CSE574/Chap11/Ch11.1-MarkovModels.pdf
- Extensions of HMMs:

  http://www.cedar.buffalo.edu/~srihari/CSE574/Chap11/Ch11.3-HMMExtensions.pdf
- Linear Dynamical Systems:

  http://www.cedar.buffalo.edu/~srihari/CSE574/Chap11/Ch11.4-LinearDynamicalSystems.pdf
- Conditional Random Fields:

  http://www.cedar.buffalo.edu/~srihari/CSE574/Chap11/Ch11.5-ConditionalRandomFields.pdf