Probabilistic Machine Learning:
11. Hidden Markov Model

Tomasz Kajdanowicz, Piotr Bielak, Maciej Falkiewicz, Kacper Kania, Piotr Zieliński

Department of Computational Intelligence
Wroclaw University of Science and Technology

Pre-reading

Prerading:

1. Hidden Markov Models Simplified

Presentation is a compilation of slides from Introduction to Machine Learning Course by Sargur Srihari, Department of Computer Science and Engineering, University at Buffalo. All credits to **Sargur Srihari**.

Table of contents

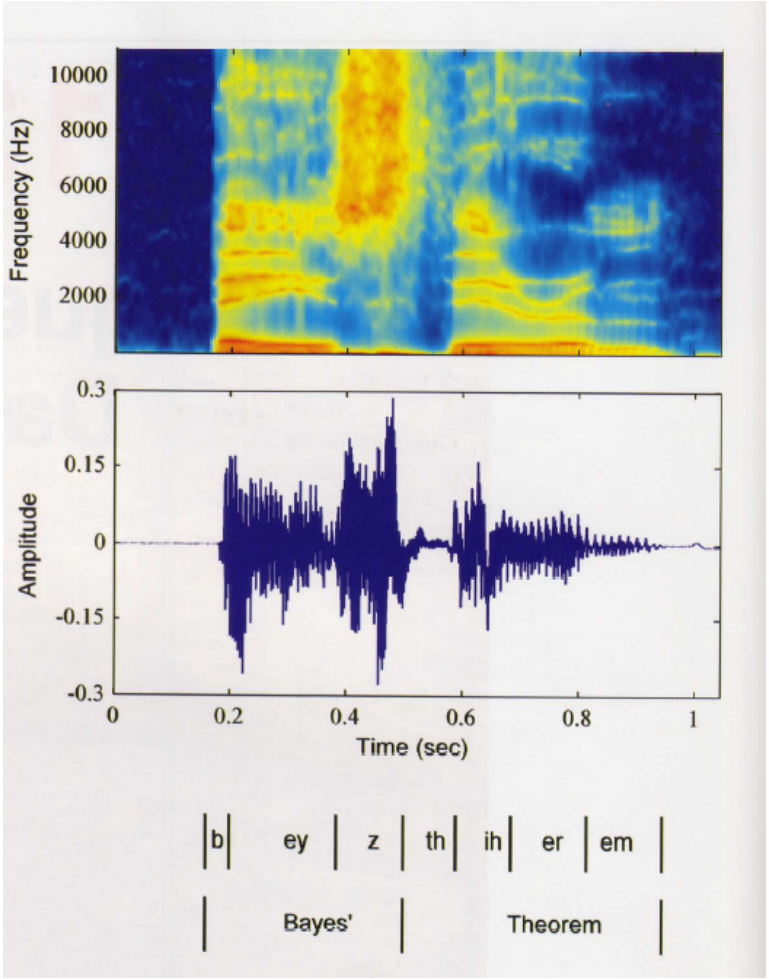
Sequential Data

Sargur Srihari
srihari@buffalo.edu

Sequential Data Examples

- Often arise through measurement of time series
  - Acoustic features at successive time frames in speech recognition
  - Sequence of characters in an English sentence
  - Parts of speech of successive words
  - Snowfall measurements on successive days
  - Rainfall measurements on successive days
  - Daily values of currency exchange rate
  - Nucleotide base pairs in a strand of DNA

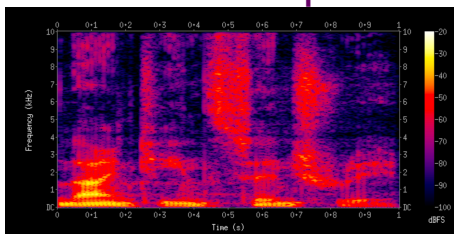
Sound Spectrogram of Speech



- "Bayes Theorem"
- Plot of the intensity of the spectral coefficients versus time index
- Successive observations of speech spectrum highly correlated (Markov dependency)

Two NLP tasks with sequential data

- Sequence-to-sequence
  - Speech recognition using a sound spectrogram
    - decompose sound waves into frequency, amplitude using Fourier transforms



→ "Nineteenth Century"

Frequencies increase up the vertical axis, and time on the horizontal axis.
The lower frequencies are more dense because it is a male voice.
Legend on right shows that the color intensity increases with density

  - NLP: Named Entity Recognition
    - Input: Jim bought 300 shares of Acme Corp. in 2006
    - NER: $[\text{Jim}]_{\text{Person}}$ bought 300 shares of $[\text{Acme Corp.}]_{\text{Organization}}$ in $[2006]_{\text{Time}}$
  - Machine Translation: Echte dicke kiste → Awesome sauce
- Sequence-to-symbol
  - Sentiment:
    - *Best movie ever* → Positive
  - Speaker recognition
    - Sound spectrogram → Harry

Stationary *vs* Non-stationary

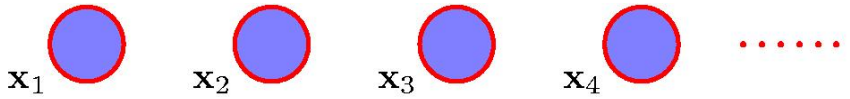
- Stationary:
  - —Data evolves over time but distribution remains same
    - e.g., dependence of current word over previous word remains constant
- Non-stationary:
  - —Generative distribution itself changes over time

Making a Sequence of Decisions

- Processes in time, states at time t are influenced by a state at time t-1
- Wish to predict next value from previous values
  - e.g., financial forecasting
- Impractical to consider general dependence of future dependence on all previous observations
  - Complexity grows without limit as number of observations increases
- Markov models assume dependence on most recent observations

Markov Model Assuming Independence

- Simplest model:



  - Assume observations are independent
  - Graph without links
- To predict whether it rains tomorrow is only based on relative frequency of rainy days
- Ignores influence of whether it rained the previous day

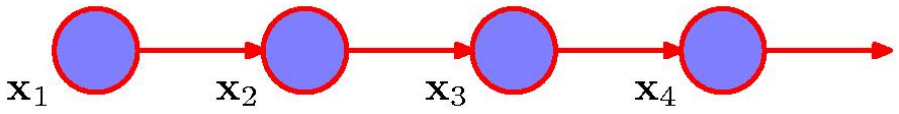
Markov Model

- Most general Markov model for observations $\{x_n\}$
- Product rule to express joint distribution of sequence of observations

$$p(x_1,..x_N) = \prod_{n=1}^{N} p(x_n \mid x_1,..x_{n-1})$$

First Order Markov Model

- Chain of observations $\{x_n\}$



- Joint distribution for a sequence of n variables

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

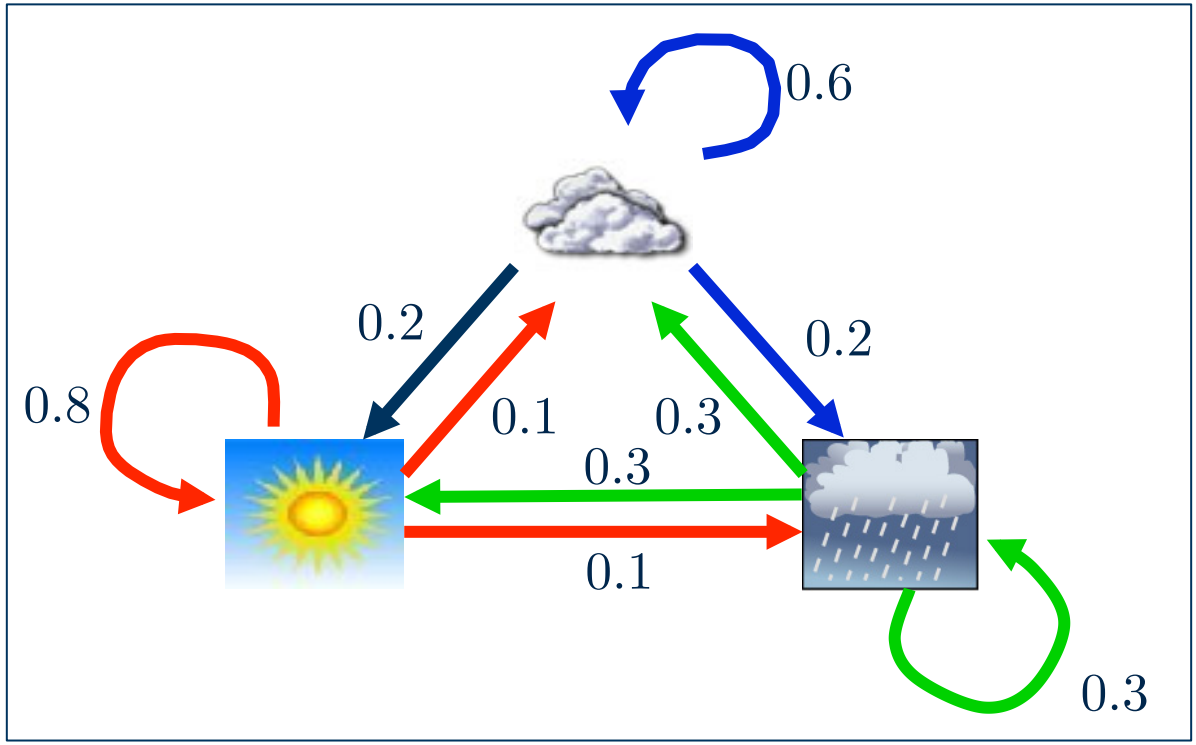
- It can be verified (using product rule from above) that

$$p(x_n \mid x_1..x_{n-1}) = p(x_n \mid x_{n-1})$$

- If model is used to predict next observation, distribution of prediction will only depend on preceding observation and independent of earlier observations
- Stationarity implies conditional distributions $p(x_n|x_{n-1})$ are all equal

Markov Model – Weather

- The weather of a day is one of the following states:
  - State 1: Rainy
  - State 2: Cloudy
  - State 3: Sunny



		Tomorrow		
		Rainy	Cloudy	Sunny
Today	Rainy	0.3	0.3	0.4
	Cloudy	0.2	0.6	0.2
	Sunny	0.1	0.1	0.8

Markov Model – Sequence probability

- What is the probability that the weather for the next 7 days will be "S-S-R-R-S-C-S"?

		Tomorrow		
		Rainy	Cloudy	Sunny
Today	Rainy	0.3	0.3	0.4
	Cloudy	0.2	0.6	0.2
	Sunny	0.1	0.1	0.8

$O = \{S_3, S_3, S_3, S_1, S_1, S_3, S_2, S_3\}$

— Find the probability of O, given the model.

$$P(O \mid Model) = P(S_3, S_3, S_3, S_1, S_1, S_3, S_2, S_3 \mid Model)$$
$$= P(S_3) \cdot P(S_3 \mid S_3) \cdot P(S_3 \mid S_3) \cdot P(S_1 \mid S_3)$$
$$\cdot P(S_1 \mid S_1) \cdot P(S_3 \mid S_1) \cdot P(S_2 \mid S_3) \cdot P(S_3 \mid S_2)$$
$$= \pi_3 \cdot a_{33} \cdot a_{33} \cdot a_{31} \cdot a_{11} \cdot a_{13} \cdot a_{32} \cdot a_{23}$$
$$= 1 \cdot (0.8) \cdot (0.8) \cdot (0.1) \cdot (0.4) \cdot (0.3) \cdot (0.1) \cdot (0.2)$$
$$= 1.536 \times 10^{-4}$$

Markov model for spoken word production

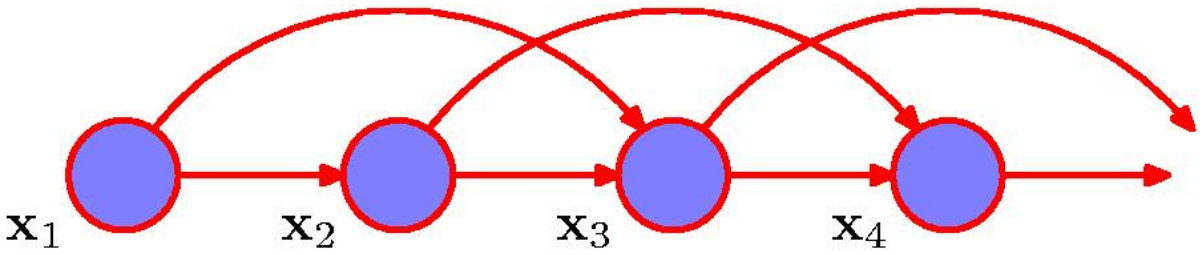
- States represent phonemes
- Production of word: "cat"
- Represented by states

  /k/ /a/ /t/
- Transitions from
  - /k/ to /a/
  - /a/ to /t/
  - /t/ to a silent state
- Although only correct cat sound is represented by model, perhaps other transitions can be introduced,
  - eg, /k/ followed by /t/

Markov Model for word "cat"

Second Order Markov Model

- Each observation is influenced by previous two observations
  - Conditional distribution of observation x_n depends on the values of two previous observations x_{n-1} and x_{n-2}



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

M^{th} Order Markov Source

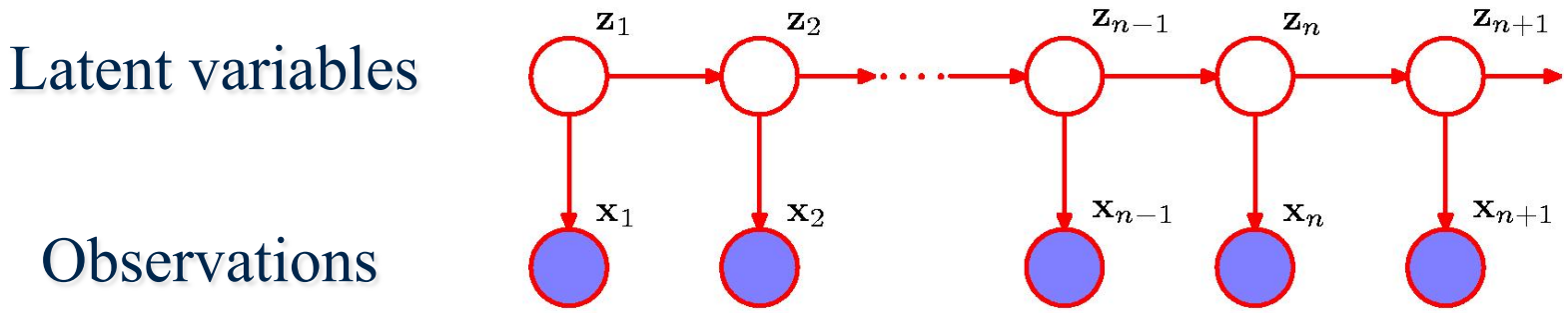
- Conditional distribution for a particular variable depends on previous M variables
- Pay a price for number of parameters
- Discrete variable with K states
  - First order: $p(x_n|x_{n\text{-}1})$ needs $K\text{-}1$ parameters for each value of $x_{n\text{-}1}$ for each of K states of x_n giving $K(K\text{-}1)$ parameters
  - M^{th} order will need $K^{M\text{-}1}(K\text{-}1)$ parameters

Latent Variables

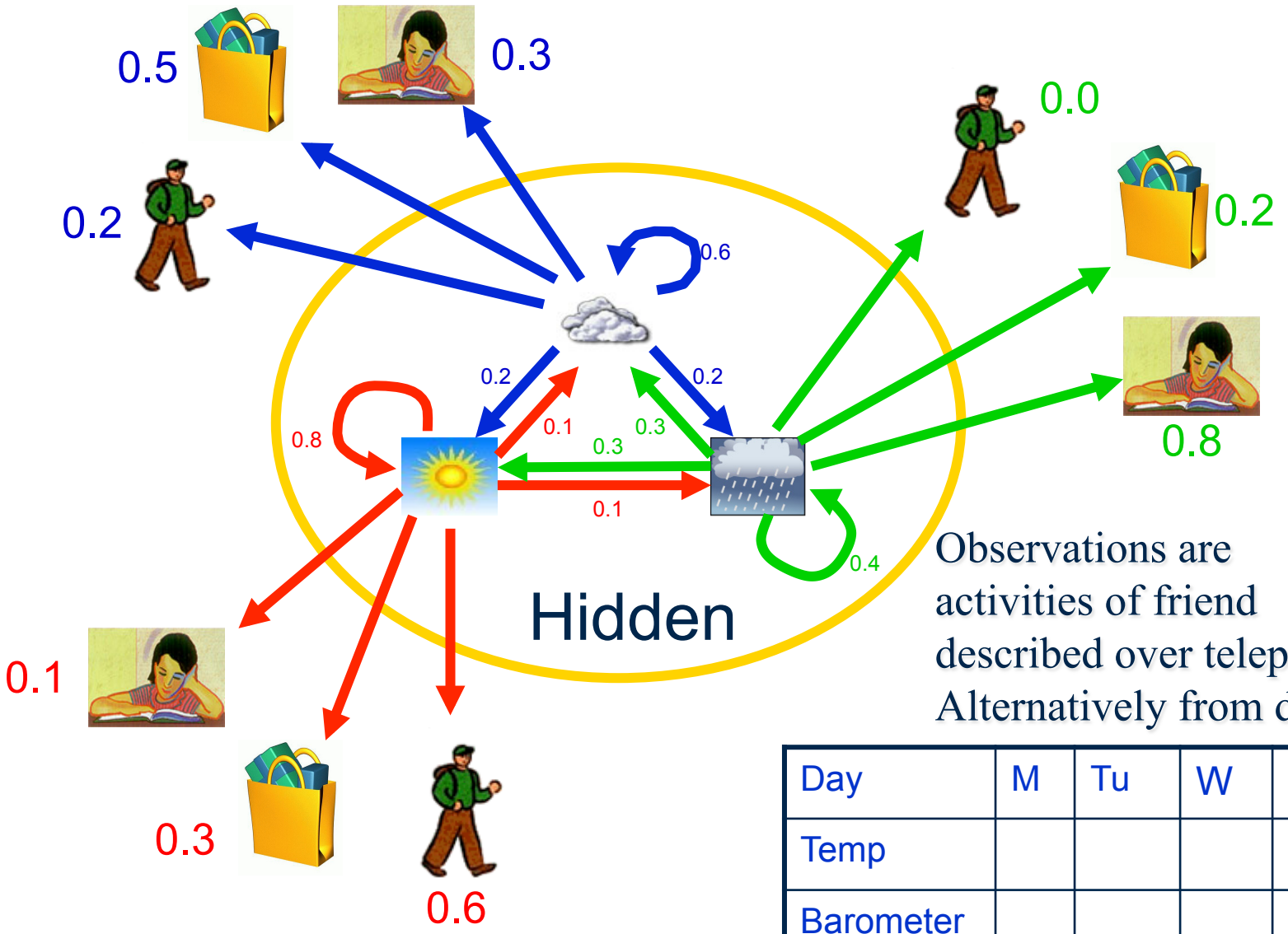
- While Markov models are tractable they are severely limited
- Introduction of latent variables provides a more general framework
- Lead to state-space models
- When latent variables are:
  - Discrete
    - they are called *Hidden Markov models*
  - Continuous
    - they are *linear dynamical systems*

Introducing Latent Variables

- Model for sequences not limited by Markov assumption of any order but with limited number of parameters
- For each observation x_n, introduce a latent variable z_n
- z_n may be of different type or dimensionality to the observed variable
- Latent variables form the Markov chain
- Gives the “state-space model”



Hidden Markov Model



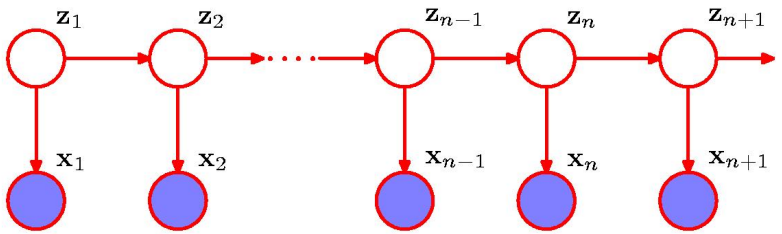
Observations are activities of friend described over telephone
Alternatively from data:

Day	M	Tu	W	Th
Temp				
Barometer				

Conditional Independence with Latent Variables

- Satisfies key assumption that

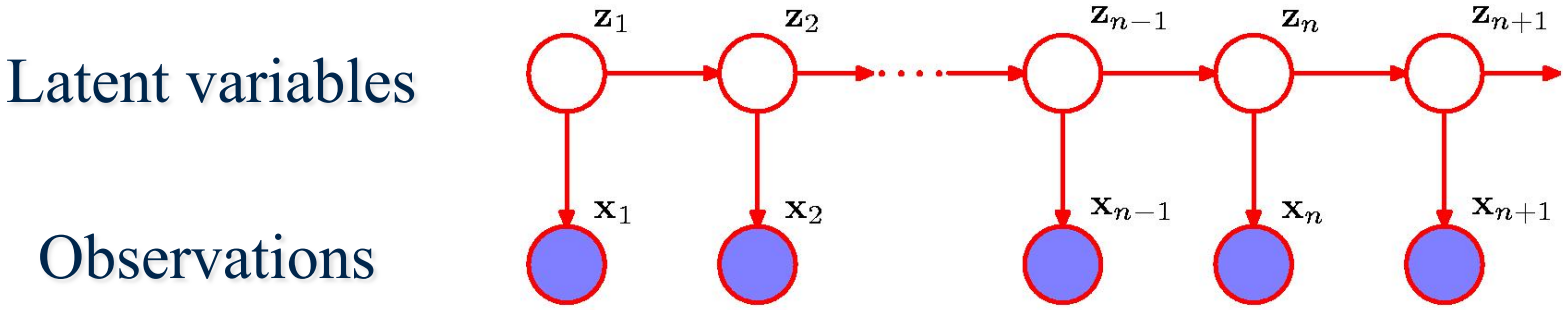
$$z_{n+1} \perp z_{n-1} \mid z_n$$



- From d-separation

  When latent node z_n is filled, the only path between z_{n-1} and z_{n+1} has a head-to-tail node that is blocked

Jt Distribution with Latent Variables



- Joint distribution for this model

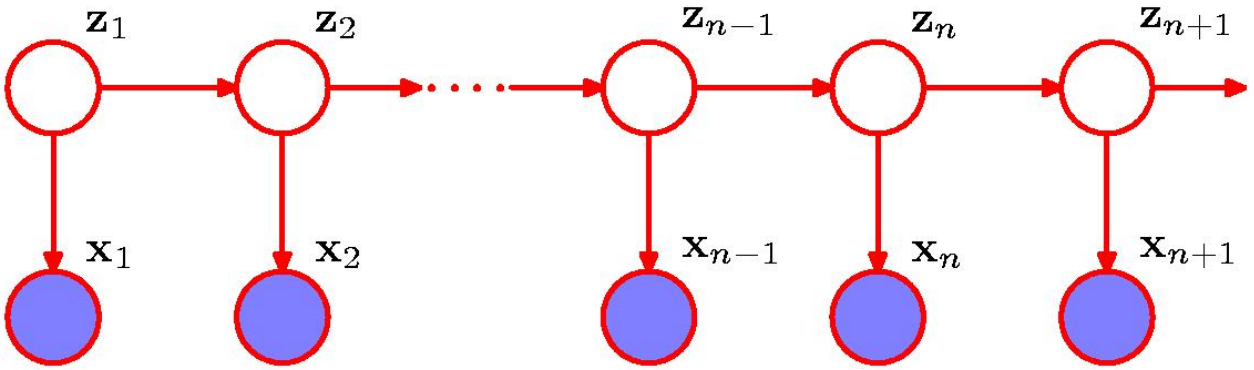
$$p(x_1,..x_N,z_1,..z_n) = p(z_1)\left[\prod_{n=2}^{N} p(z_n \mid z_{n-1})\right]\prod_{n=1}^{N} p(x_n \mid z_n)$$

- There is always a path between any x_n and x_m via latent variables which is never blocked
- Thus predictive distribution $p(x_{n+1}|x_1,..,x_n)$ for observation x_{n+1} does not exhibit conditional independence properties and is hence dependent on all previous observations

Two Models Described by Graph

Latent variables

Observations



1. Hidden Markov Model: If latent variables are discrete:
   Observed variables in a HMM may be discrete or continuous
2. Linear Dynamical Systems: If both latent and observed variables are Gaussian

Hidden Markov Models

Sargur Srihari

srihari@cedar.buffalo.edu

HMM Overview

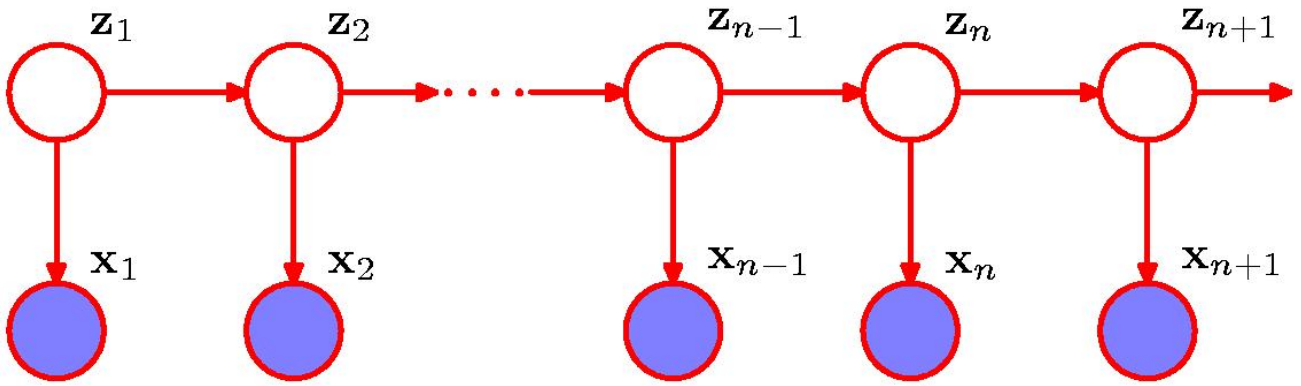
1. What is an HMM?
2. State-space Representation
3. HMM Parameters
4. Generative View of HMM

Role of HMMs in ML

- Ubiquitous tool for modeling time series data
  - Used in most all speech recognition systems
  - Computational molecular biology
  - Group amino acid sequences into proteins
- It is a BN for representing probability distributions over sequences of observations
- HMM has two defining properties:
  - Observation $\mathbf{x}_t$ at time t was generated by some process whose state $\mathbf{z}_t$ is hidden from the observer
  - Assumes that state at $\mathbf{z}_t$ is dependent only on state $\boldsymbol{z}_{t\text{-}1}$ and independent of all prior states (First order)
- Example:
  - $\mathbf{z}$ are phoneme sequences, $\mathbf{x}$ are acoustic observations

Graphical Model of an HMM

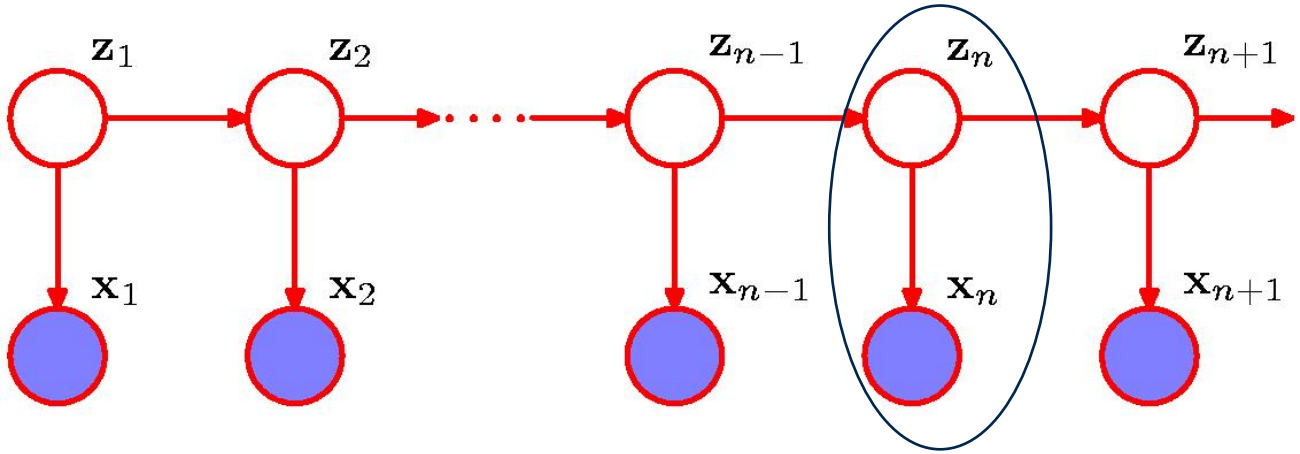
- Has the state space model shown below and latent variables are discrete



- Joint distribution has the form

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

Mixture Viewed as HMM



- A single time slice corresponds to a mixture distribution with component densities $p(\mathrm{x}|\mathrm{z})$
- 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 $one\text{-}of\text{-}K$ coding scheme

Transitional Probability Matrix

- Joint distribution:

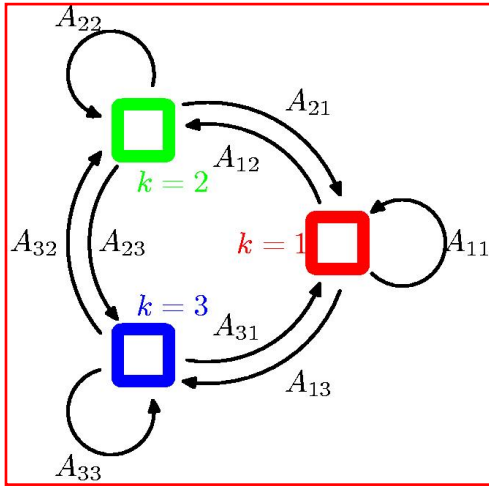
$$p(x_1,..x_N,z_1,..z_n) = p(z_1)\left[\prod_{n=2}^{N} p(z_n \mid z_{n-1})\right]\prod_{n=1}^{N} p(x_n \mid z_n)$$

  - We allow $\mathbf{z}_n$ to depend on $\mathbf{z}_{n-1}$ via $p(\mathbf{z}_n|\mathbf{z}_{n-1})$
    - Latent variables are 1-of-K, thus this conditional distribution is specified by a transition probability matrix $\boldsymbol{A}$ with $A_{jk} = p(z_{nk}=1|z_{n-1,j}=1)$
  - Conditional distribution is stated as

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

    - Exponent $z_{n-1,j}z_{n,k}$, which is a product, is 0 or 1
    - Hence product evaluates to a single A_{jk} for each setting of values of $\mathbf{z}_n$, $\mathbf{z}_{n-1}$
    - Thus $p(\mathbf{z}_n=3|\mathbf{z}_{n-1}=2)=\mathrm{A}_{23}$

Transition diagram where latent variables have three possible states



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

State of z_n State of z_{n-1} A matrix

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

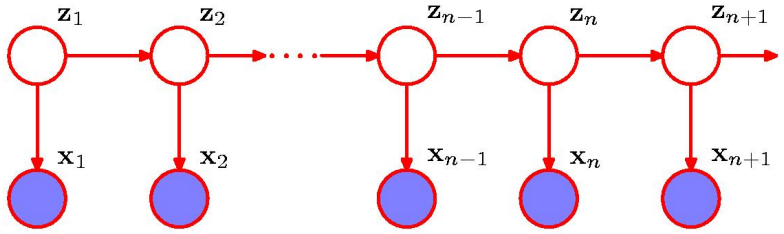
j	1	2	.	K
$z_{n-1,j}$	1	0	.	0

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

	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}

Matrix $\boldsymbol{A}$ has $K(K\text{-}1)$ independent parameters

Initial Variable Probabilities



- Initial latent node $\mathbf{z}_1$ does not have a parent node
  - Its marginal distribution $p(\mathbf{z}_1)$ is represented by a vector of probabilities $\boldsymbol{\pi}$ with elements $\pi_k = p(z_{1k}=1)$ so that

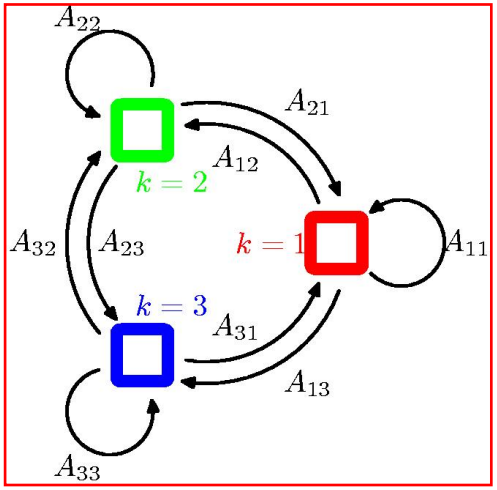
$$p(\mathbf{z}_1 \mid \boldsymbol{\pi}) = \prod_{k=1}^{K} \pi_k^{z_{1,k}} \quad \text{where } \Sigma_k \pi_k = 1$$

- Note that $\boldsymbol{\pi}$ is an HMM parameter
  - representing probabilities of each state for the first variable

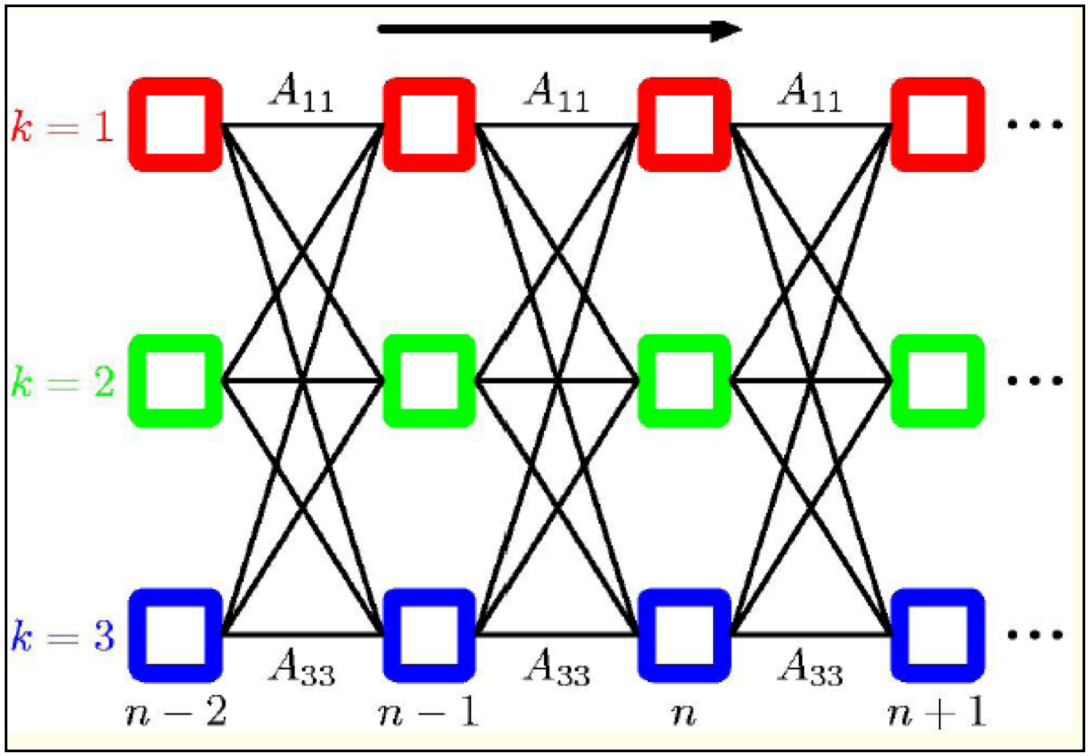
Unfolding State Transition over time

Lattice or trellis representation of the latent states

State Transition diagram



Each column corresponds to one of the $\mathbf{z}_n$



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

- Specification of probabilistic model is completed by defining conditional distributions of observed variables $p(\mathrm{x}_n|\mathrm{z}_n,\phi)$ where ϕ are parameters
  - These are known as *emission probabilities* which can be either continuous (by Gaussians) or discrete (by tables)
- Because $\mathbf{x}_n$ is observed, distribution $p(\mathbf{x}_n|\mathbf{z}_n,\phi)$ consists of a table of K numbers corresponding to K states of binary vector $\mathbf{z}_n$
  - Emission probabilities can be represented as

$$p(\mathbf{x}_n|\mathbf{z}_n,\phi) = \prod_{k=1}^{K}(p(\mathbf{x}_n|\phi_k))^{z_{nk}}$$

Homogeneous Models

- All conditional distributions governing the latent variables share the same parameters $\mathbf{A}$
- All the emission distributions share the same parameters ϕ

Joint Distribution over Latent and Observed variables

- Joint can be expressed in terms of parameters:

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

where $X=\{x_1,..x_N\},\ Z=\{z_1,..z_N\},\ \theta=\{\pi,A,\phi\}$

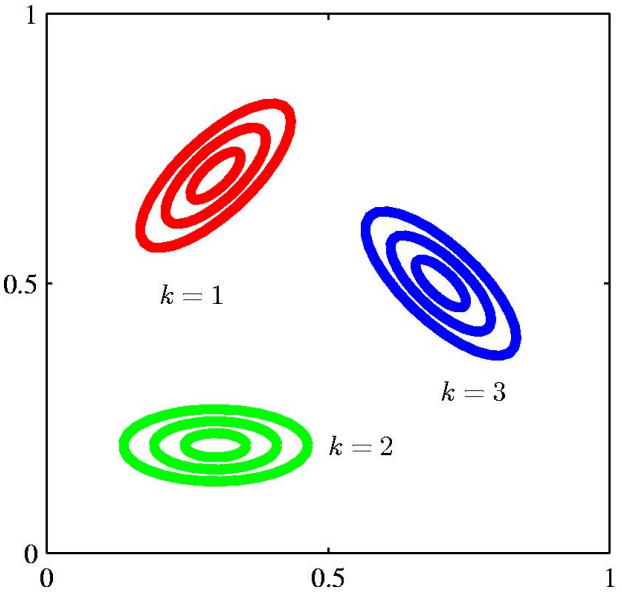
- Most discussion of HMM is independent of emission probabilities
  - Tractable for discrete tables, Gaussian, GMMs

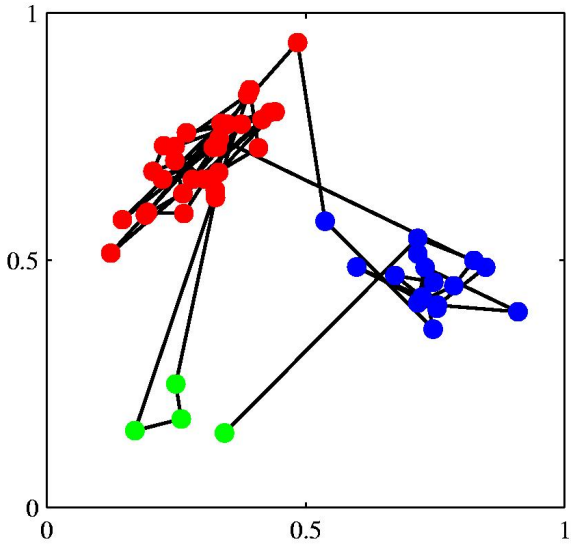
HMM from a generative viewpoint

- Can get a better understanding of HMMs by considering it from a generative viewpoint
- First choose latent variable $\mathbf{z}_1$ with probabilities determined by probabilities π_k and then sample the corresponding observation $\mathbf{x}_1$
- Now we choose the state of the variable $\mathbf{z}_2$ according to the transition probabilities $p(\mathbf{z}_2|\mathbf{z}_1)$ according to the already instantiated value of $\mathbf{z}_1$
- This is an example of ancestral sampling for a directed PGM

Example of sampling from a HMM

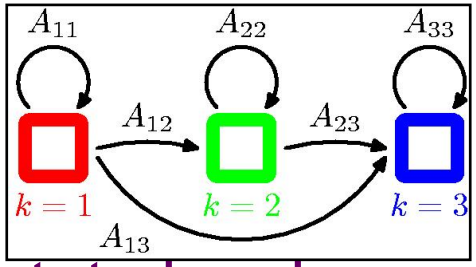
- Generative viewpoint
- Three states of a latent variable
- Gaussian emission model $p(\mathrm{x}|\mathrm{z})$
- Two-dimensional x
- 50 data points generated
- Transition probabilities
  - 5% probability of making transition
  - 90% of remaining in same



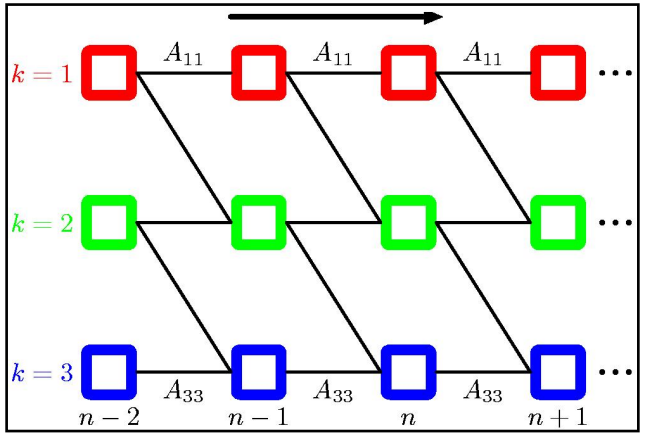


A variant of HMM

- Imposing restrictions on transition matrix $\mathbf{A}$
  - Left-to right HMM
    - Setting the elements of $A_{jk}=0$ if $k < j$

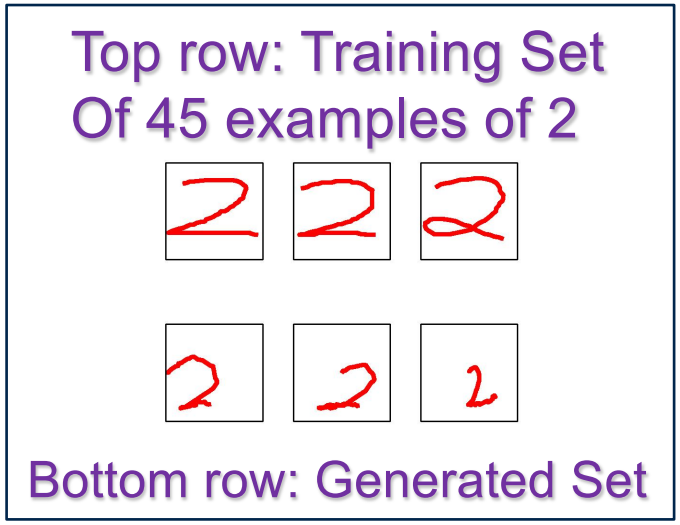


    - Once a state has been vacated, it cannot be reentered
    - Corresponding lattice



Left-to-Right HMM Applied to Digits

- Examples of on-line handwritten digits



  - $K{=}16$ states: corresponding to a line segment of fixed length in one of 16 angles
  - Emission probabilities: 16 x 16 table, associated with allowed angle values for each state
  - Transition probabilities set to zero except for those that keep state index k the same or increment by one
  - Parameters optimized by 25 iterations of EM

HMM invariance to time warping

- Time warping (compression and stretching)
- On-line handwriting recognition
  - A typical digit consists of two strokes
    - Arc starts at top left down to cusp/loop at bottom left
    - A straight sweep ending at bottom right
    - Relative sizes of the two sections vary and hence the location of the cusp/loop
  - HMM accommodates this by no of transitions to same state vs transitions to successive state
- Speech recognition
  - Warping of time axis is speed of speech
  - HMM accommodates such distortion

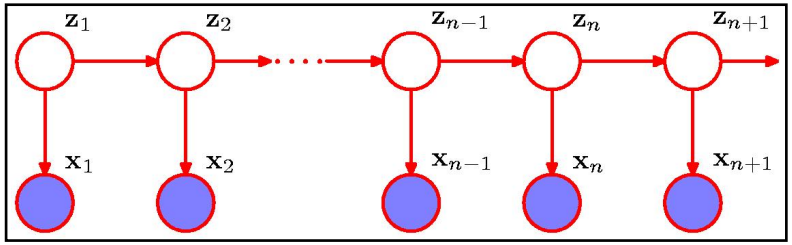
Maximum Likelihood for the HMM

Sargur Srihari
srihari@cedar.buffalo.edu

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

HMM Parameters



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

where $X = \{x_1,..x_N\},\ Z = \{z_1,..z_N\},\ \theta = \{\pi, A, \phi\}$

We have three sets 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

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 $p(\mathrm{x}_k|\mathrm{z}_k)$

- A and π parameters are often initialized uniformly
- Initialization of ϕ depends on form of distribution

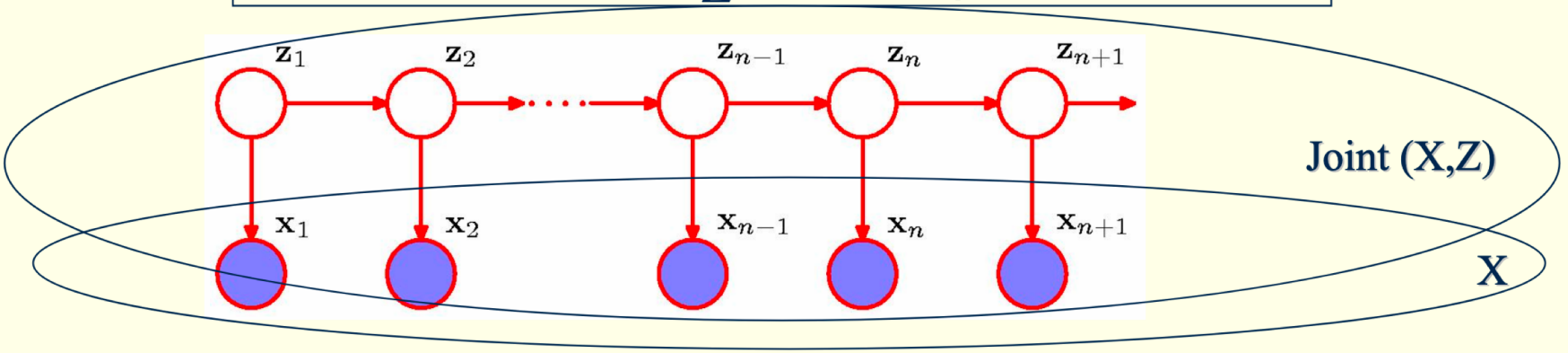
Determining HMM Parameters

- 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 $Z=\{z_1,..z_n\}$

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

where $X=\{x_1,..x_N\}$, $Z=\{z_1,..z_N\}$, $\theta=\{\pi,A,\phi\}$

$$p(X\mid\theta)=\Sigma_Z\, p(X,Z\mid\theta)$$



Computational Issues for Parameters

- Joint distribution is $p(X|\theta) = \sum_Z p(X,Z|\theta)$
  - where

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

where $X = \{x_1,..x_N\},\ Z = \{z_1,..z_N\},\ \theta = \{\pi, A, \phi\}$

- Joint distribution $p(X,Z|\boldsymbol{\theta})$ does not factorize over n, we cannot treat each of the summations over $\mathbf{z}_n$ independently
- There are N variables summed over each of which has K states, so there are K^N terms
  - No. of terms grows exponentially with length of chain, summing over all paths in trellis

Solution to computational task

- Use conditional independence properties to reorder summations to obtain algorithm that scales linearly with length of chain
- Use Expectation Maximization to maximizing the log-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

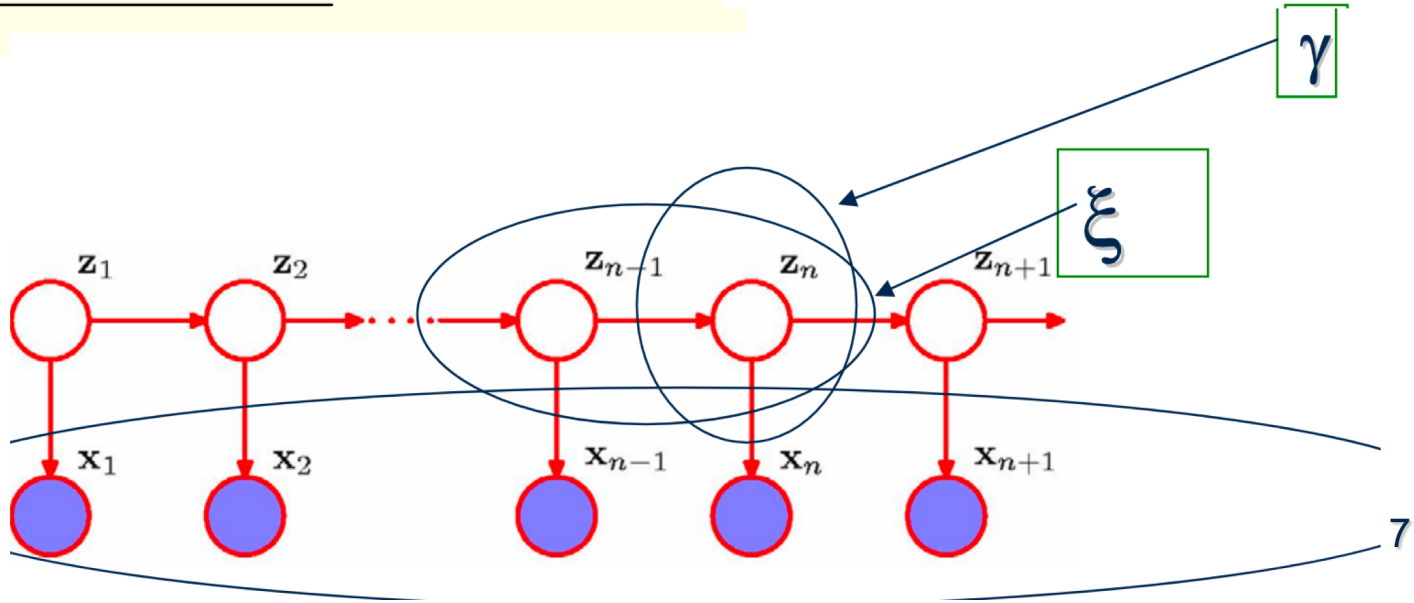
- Introduce notation Gamma and Xi

  $\gamma(z_n) = p(z_n|X,\theta^{old})$: 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

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



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

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

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

Expansion of Q

- We begin with

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

- Substitute

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

- And use definitions of γ and ξ to get:

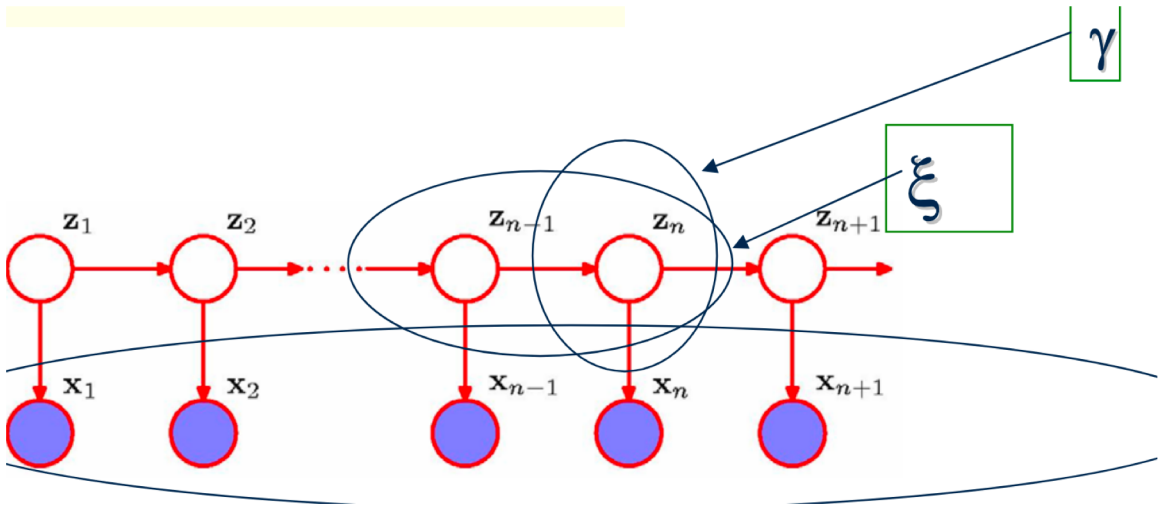
$$Q(\theta,\theta^{old}) = \sum_{k=1}^{K} \gamma(z_{1k}) \ln \pi_k + \sum_{n=2}^{N}\sum_{j=1}^{K}\sum_{k=1}^{K} \xi(z_{n-1,j},z_{nk}) \ln A_{jk}$$
$$+ \sum_{n=1}^{N}\sum_{k=1}^{K} \gamma(z_{nk}) \ln p(x_n \mid \phi_k)$$

E-Step

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

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

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)

$$\pi_k = \frac{\gamma(z_{1k})}{\sum_{j=1}^{K} \gamma(z_{1j})}$$

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

- Maximization w.r.t. ϕ_k
  - Only last term of Q depends on ϕ_k → $\sum_{n=1}^{N} \sum_{k=1}^{K} \gamma(z_{nk}) \ln p(x_n \mid \phi_k)$
  - 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:

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

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

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

The forward-backward algorithm

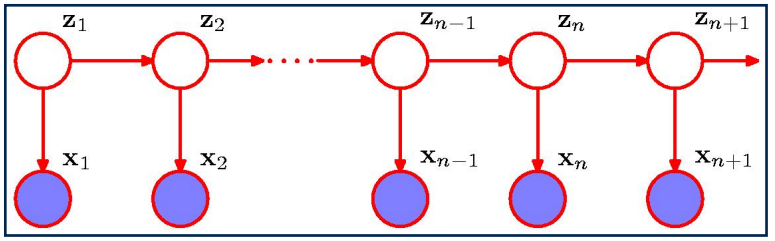
Sargur Srihari
srihari@buffalo.edu

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

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

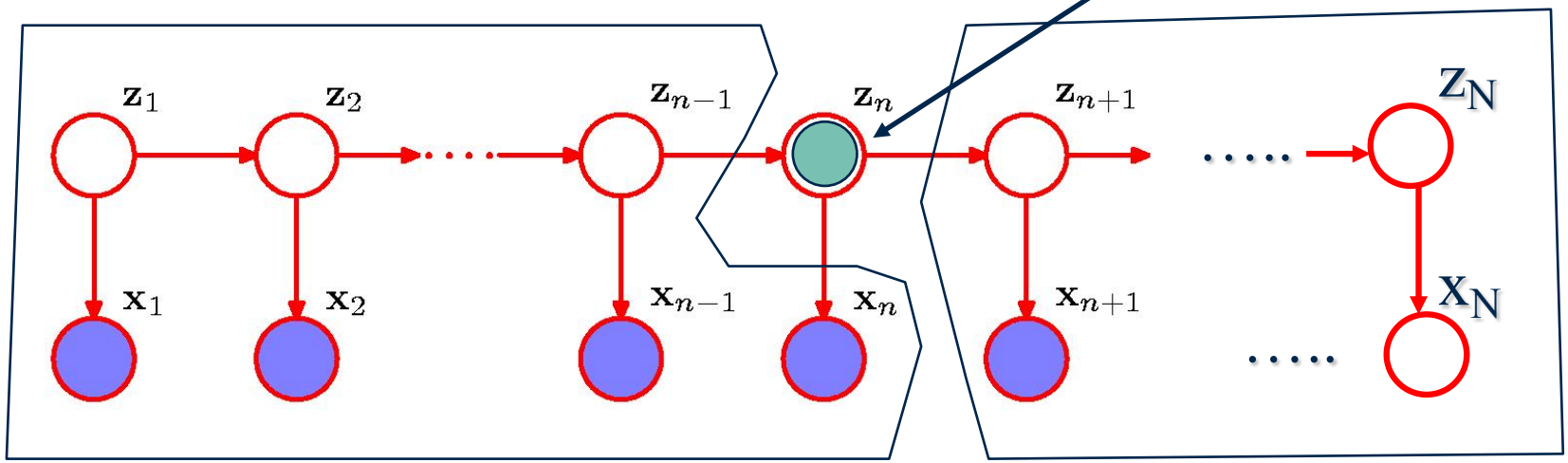
Derivation of Forward-Backward

- Several conditional-independences (A-H) hold

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

- Proved using d-separation:

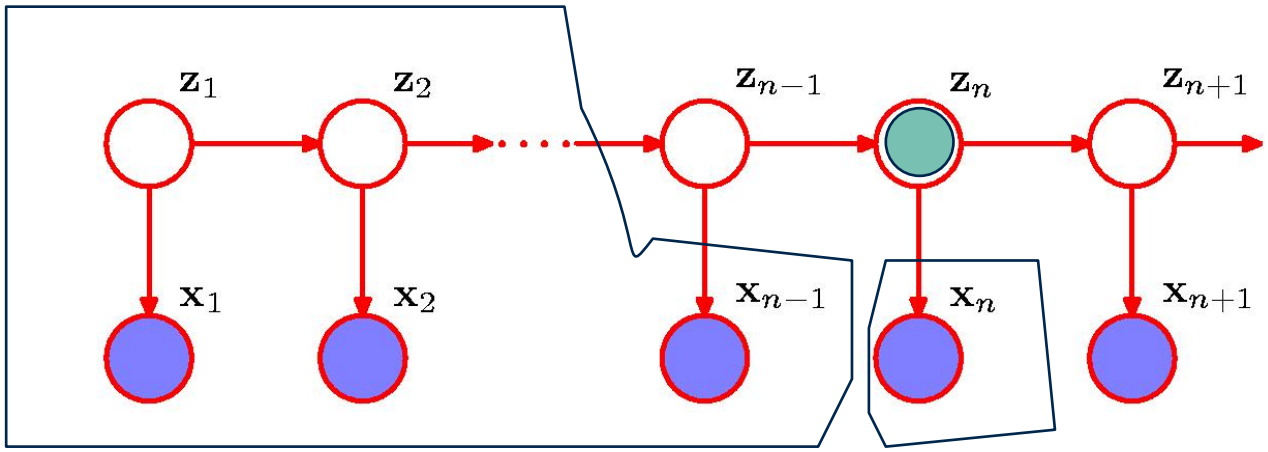
Path from x_1 to x_{n-1} passes through z_n which is observed.
Path is head-to-tail. Thus $(\mathrm{x}_1,..\mathrm{x}_{n-1}) \perp\!\!\!\perp \mathrm{x}_n \mid \mathrm{z}_n$
Similarly $(\mathrm{x}_1,..\mathrm{x}_{n-1},\mathrm{x}_n) \perp\!\!\!\perp \mathrm{x}_{n+1},..x_N \mid \mathrm{z}_n$



Conditional independence B

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

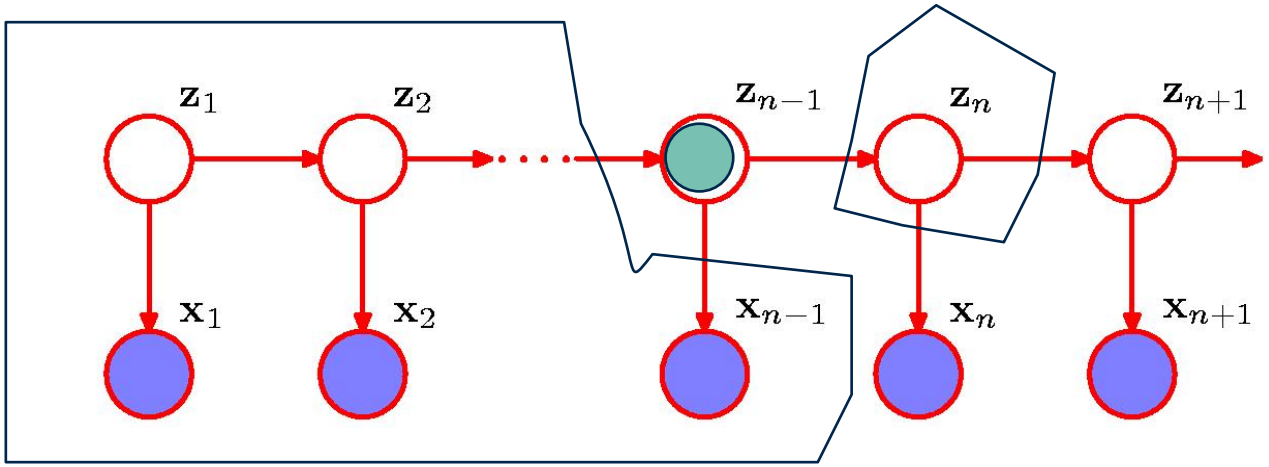
B. $p(x_1,..x_{n-1}|x_n,z_n) = p(x_1,..x_{n-1}|z_n)$



Conditional independence C

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

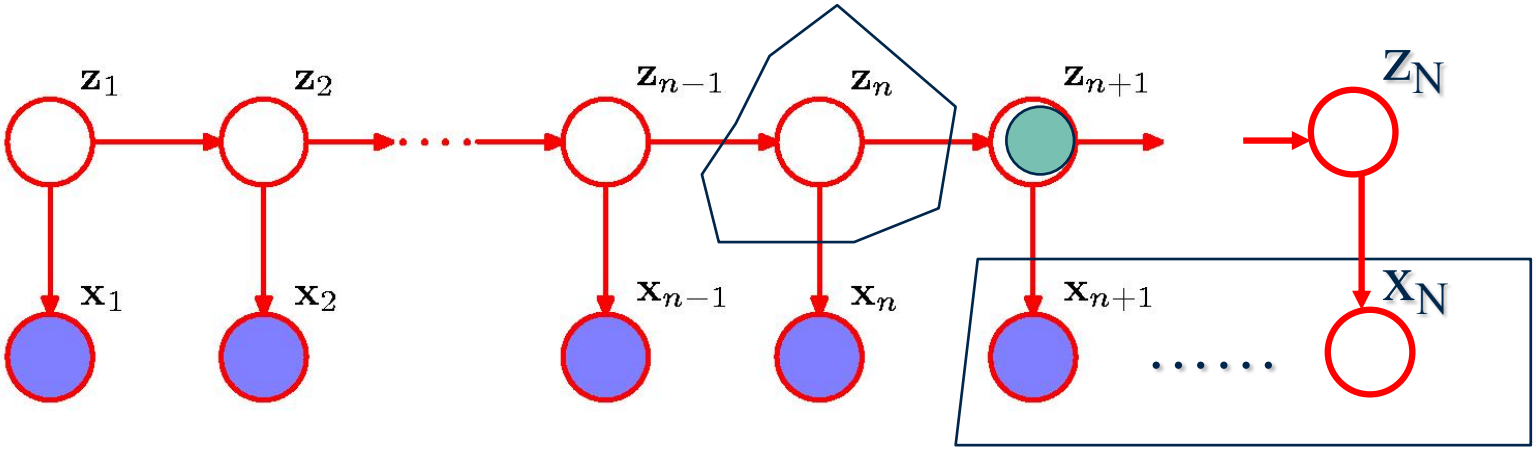
C. $p(x_1,..x_{n-1}|z_{n-1},z_n) = p(x_1,..x_{n-1}|z_{n-1})$



Conditional independence D

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

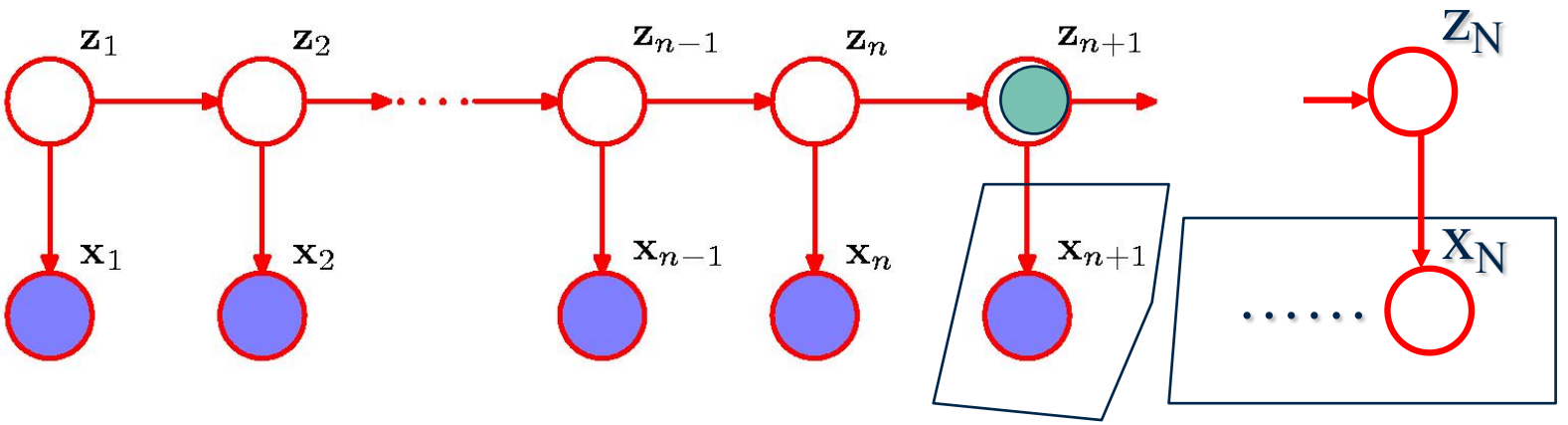
D. $p(x_{n+1},..x_N|z_n,z_{n+1}) = p(x_1,..x_N|z_{n+1})$



Conditional independence E

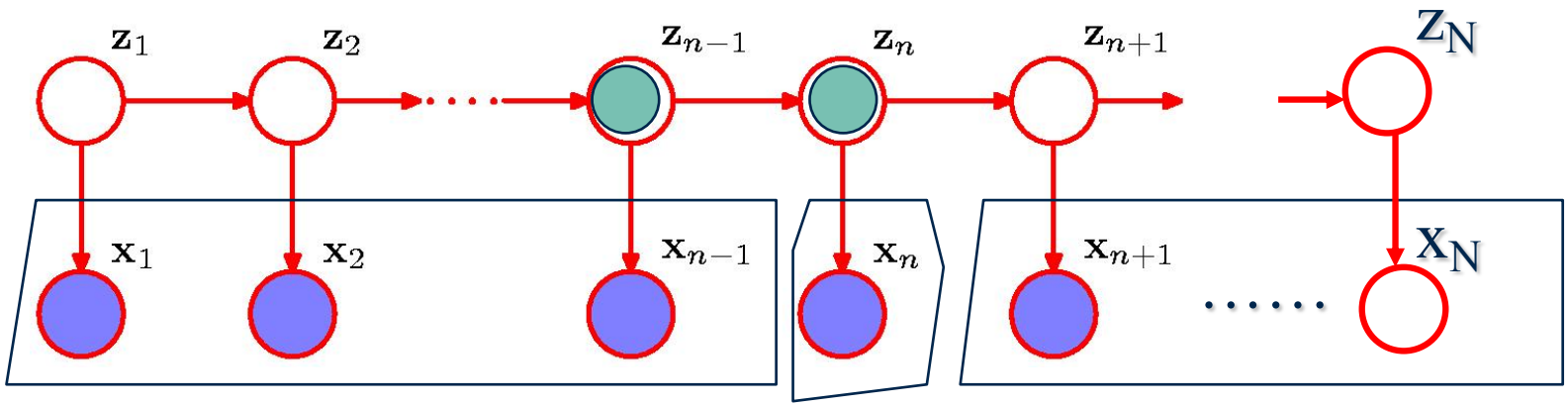
- Since $(\mathbf{x}_{n+2},..\mathbf{x}_N) \perp\!\!\!\perp \mathbf{z}_n \mid \mathbf{z}_{n+1}$

E. $p(\mathbf{x}_{n+2},..\mathbf{x}_N|\mathbf{z}_{n+1},\mathbf{x}_{n+1}) = p(\mathbf{x}_{n+2},..\mathbf{x}_N|\mathbf{z}_{n+1})$



Conditional independence F

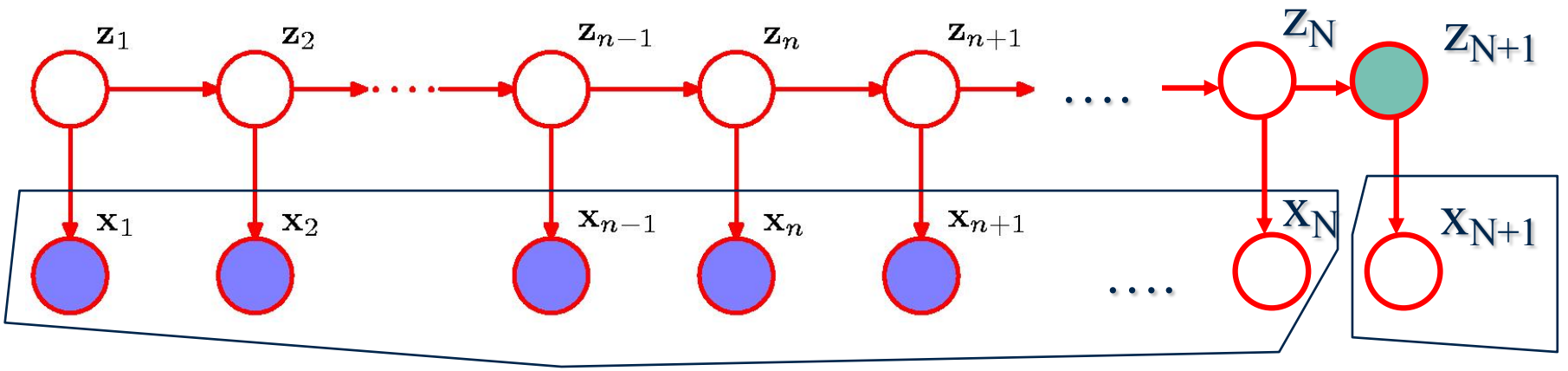
F. $p(X|z_{n-1},z_n) = p(x_1,..x_{n-1}|z_{n-1})p(x_n|z_n)$
$p(x_{n+1},..x_N|z_n)$



Conditional independence G

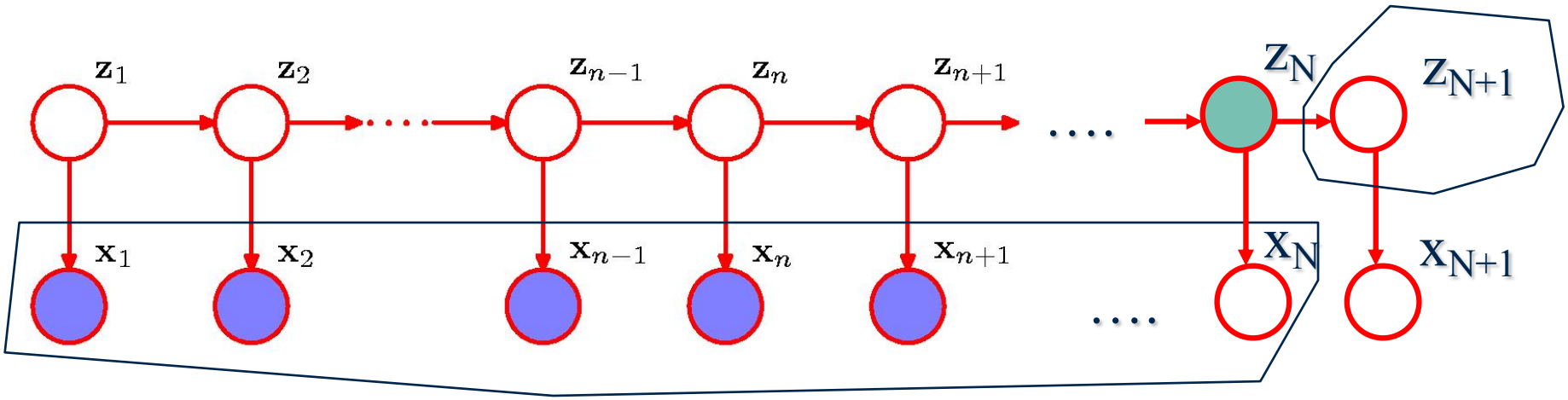
Since $(x_1,..x_N) \perp\!\!\!\perp x_{N+1} \mid z_{N+1}$

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 alpha and beta

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

$$\gamma(z_n) = \frac{p(x_1,..x_n \mid z_n)p(x_{n+1},..x_N \mid z_n)p(z_n)}{p(X)}$$

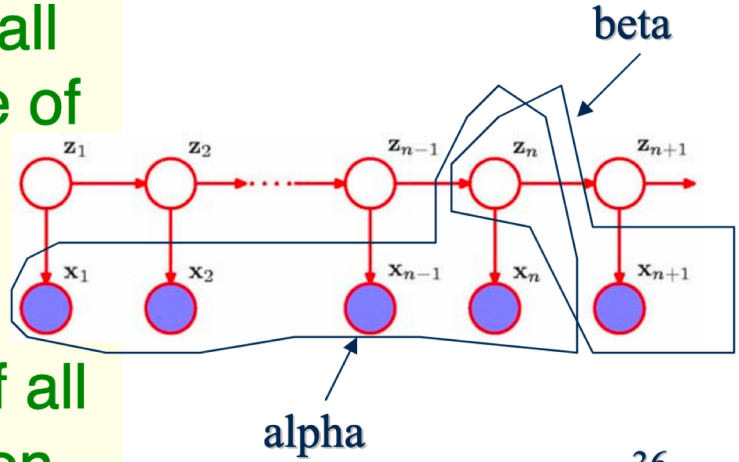
$$= \frac{p(x_1,..x_n, z_n)p(x_{n+1},..x_N \mid z_n)}{p(X)} = \boxed{\frac{\alpha(z_n)\beta(z_n)}{p(X)}}$$

- where $\boxed{\alpha(z_n) = p(x_1,..,x_n,z_n)}$

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

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

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



36

Recursion relation for alpha

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

Forward recursion for alpha evaluation

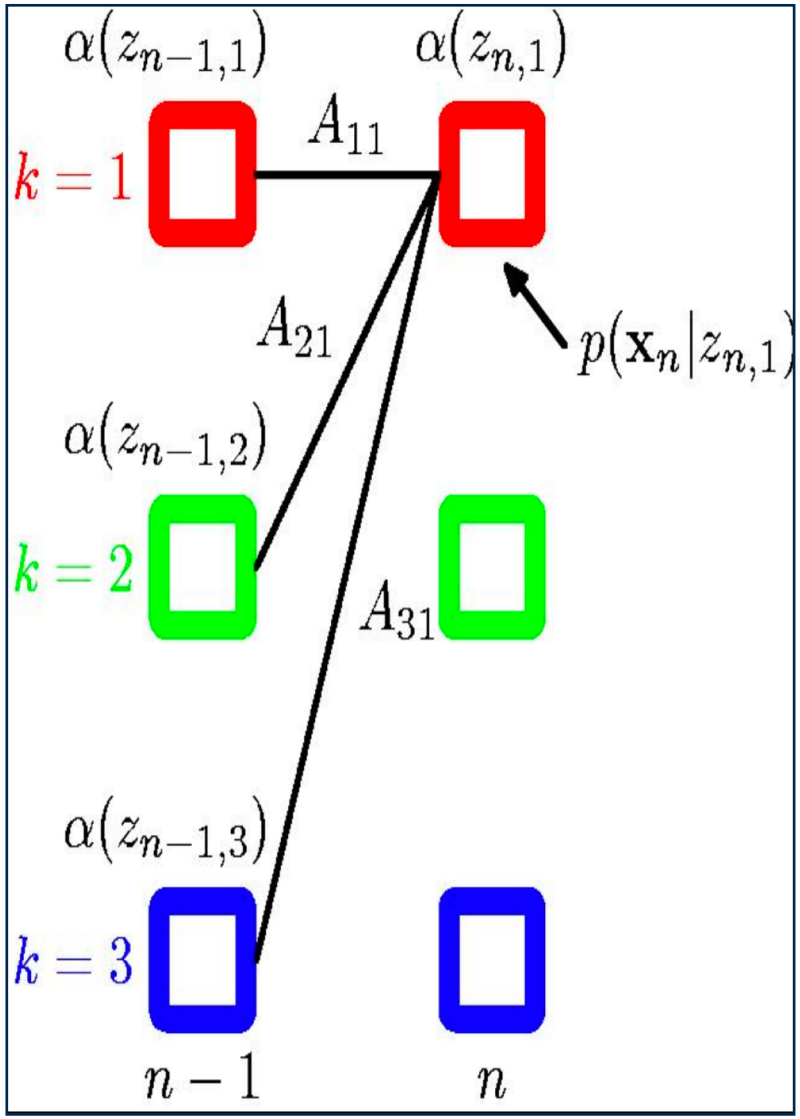
- Recursion Relation is

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

- 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

$$\alpha(\mathbf{z}_1) = p(\mathbf{x}_1, \mathbf{z}_1) = p(\mathbf{z}_1) p(\mathbf{x}_1 \mid \mathbf{z}_1) = \prod_{k=1}^{K} \{\pi_k p(\mathbf{x}_1 \mid \phi_k)\}^{z_{1k}}$$

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



Recursion relation for beta

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

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

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

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

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

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

Backward recursion for beta

- Backward message passing

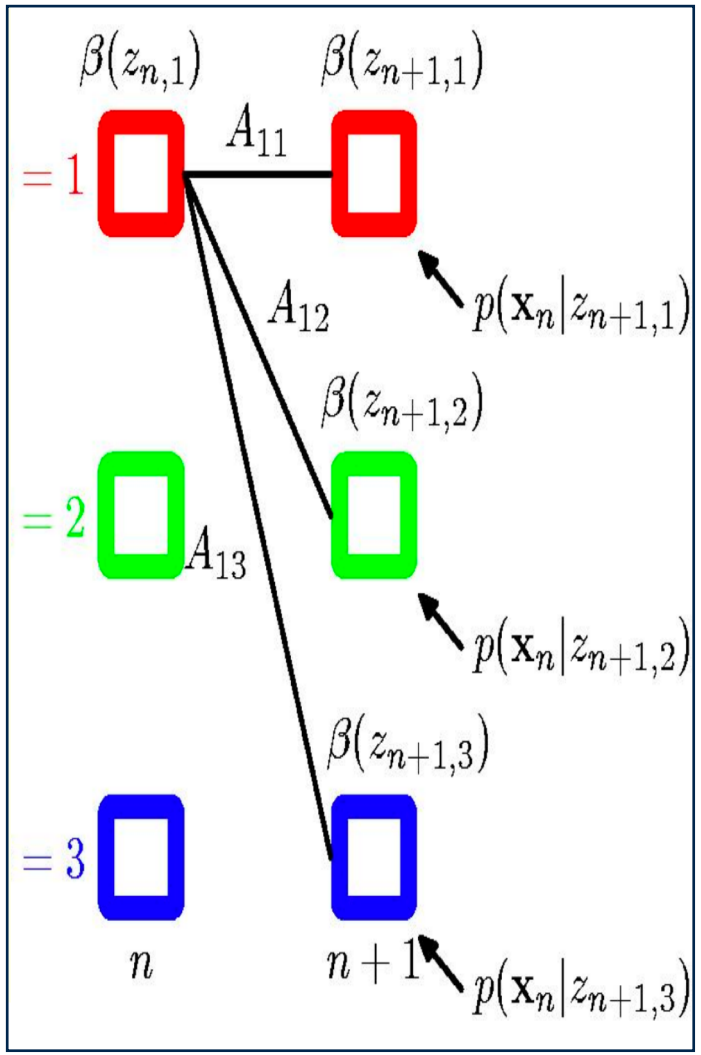
$$\beta(\mathbf{z}_n) = \sum_{\mathbf{z}_{n+1}} \beta(\mathbf{z}_{n+1}) p(\mathbf{x}_{n+1}|\mathbf{z}_n) p(\mathbf{z}_{n+1}|\mathbf{z}_n)$$

- Evaluates $\beta(\mathbf{z}_n)$ in terms of $\beta(\mathbf{z}_{n+1})$

- Starting condition for recursion is

$$p(\mathbf{z}_N|\mathbf{X}) = \frac{p(\mathbf{X},\mathbf{z}_N)\beta(\mathbf{z}_N)}{p(\mathbf{X})}$$

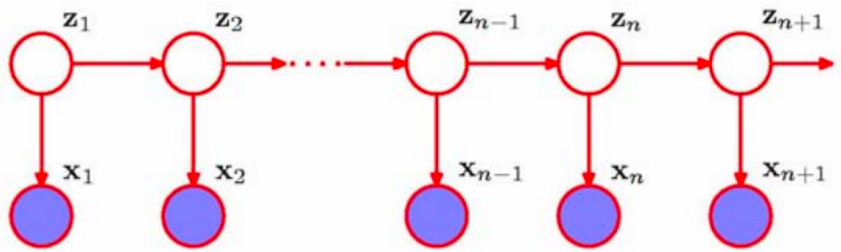
- Is correct provided we set $\beta(\mathbf{z}_N) = 1$ for all settings of $\mathbf{z}_N$
  - This is the initial condition for backward computation



M-step Equations

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

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



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

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)

$$\xi(z_{n-1},z_n) = p(z_{n-1},z_n \mid X) \text{ by definition}$$

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

$$= \frac{p(x_1,..x_{n-1}|z_{n-1})p(x_n \mid z_n)p(x_{n+1},..,x_N \mid z_n)p(z_n \mid z_{n-1})p(z_{n-1})}{p(X)} \text{ by cond ind F}$$

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

- 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 $p(\mathrm{x_n}|\mathrm{z_n})$
- 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

Length of Sequence

- 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

Predictive Distribution

- Observed data is $\mathrm{X}=\{\mathrm{x}_1,..,\mathrm{x}_N\}$
- Wish to predict x_{N+1}
- Application in financial forecasting

$$p(\mathrm{x}_{N+1}\mid X)=\sum_{z_{N+1}} p(\mathrm{x}_{N+1},\mathrm{z}_{N+1}\mid X)$$

$$=\sum_{z_{N+1}} p(\mathrm{x}_{N+1}|\mathrm{z}_{N+1}\mid X)p(\mathrm{z}_{N+1}\mid X) \text{ by Product Rule}$$

$$=\sum_{z_{N+1}} p(\mathrm{x}_{N+1}|\mathrm{z}_{N+1})\sum_{z_N} p(\mathrm{z}_{N+1},z_N\mid X) \text{ by Sum Rule}$$

$$=\sum_{z_{N+1}} p(\mathrm{x}_{N+1}|\mathrm{z}_{N+1})\sum_{z_N} p(\mathrm{z}_{N+1}\mid z_N)p(z_N\mid X) \text{ by conditional ind H}$$

$$=\sum_{z_{N+1}} p(\mathrm{x}_{N+1}|\mathrm{z}_{N+1})\sum_{z_N} p(\mathrm{z}_{N+1}\mid z_N)\frac{p(z_N,X)}{p(X)} \text{ by Bayes rule}$$

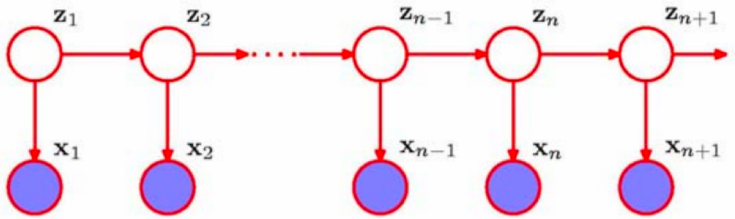
$$=\frac{1}{p(X)}\sum_{z_{N+1}} p(\mathrm{x}_{N+1}|\mathrm{z}_{N+1})\sum_{z_N} p(\mathrm{z}_{N+1}\mid z_N)\alpha(z_N) \text{ by definition of } \alpha$$

- 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

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

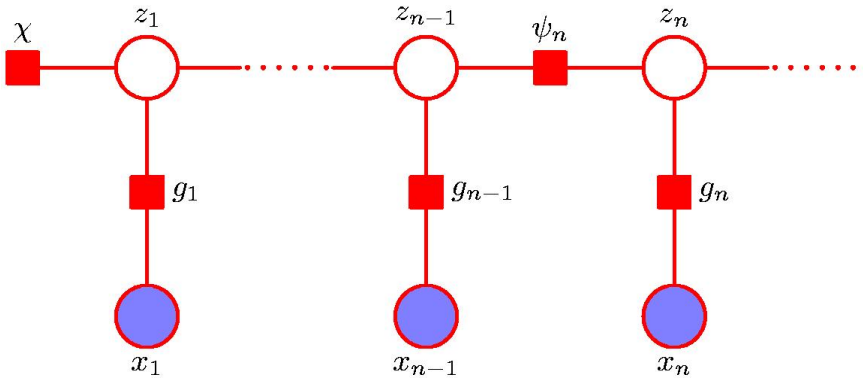
HMM Graph



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

$$h(z_1) = p(z_1)p(x_1 \mid z_1)$$

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

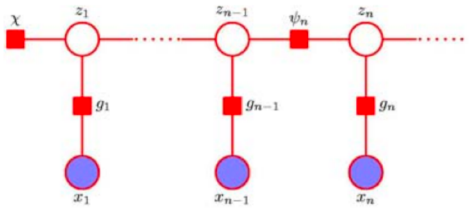
- Messages propagated are

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

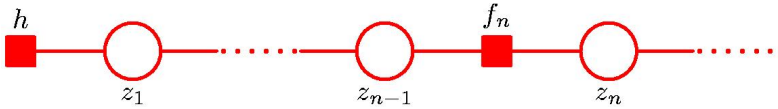
$$\mu_{f_n\to z_n}(z_n) = \sum_{z_{n-1}} f_n(z_{n-1}, z_n)\mu_{z_{n-1}\to f_n}(z_{n-1})$$

- 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

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

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

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

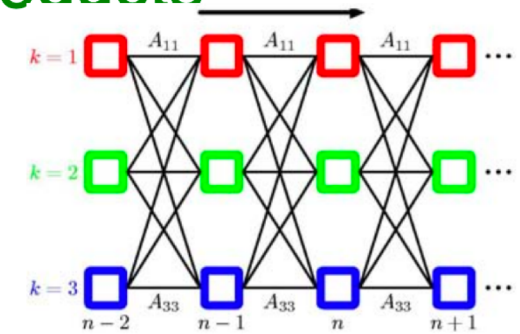
$$\xi(z_{n-1}, z_n) = \frac{\alpha(z_{n-1})p(x_n \mid z_n)p(z_n \mid z_{n-1})\beta(z_n)}{p(X)}$$

Scaling Factors

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

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

  - 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



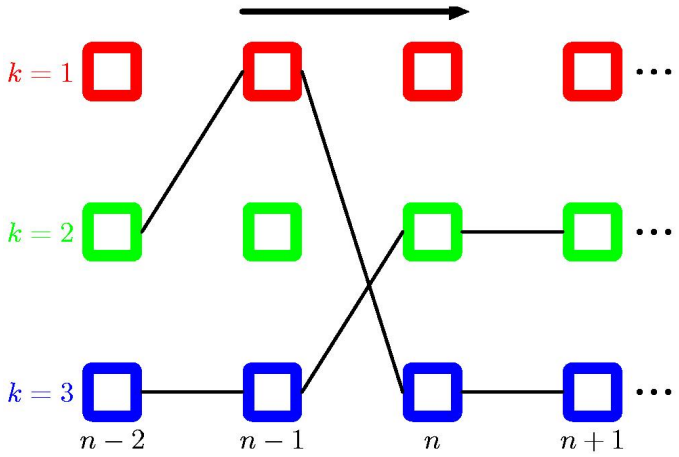
- Solution is rescaling
  - of $\alpha(z_n)$ and $\beta(z_n)$ whose values remain close to unity

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

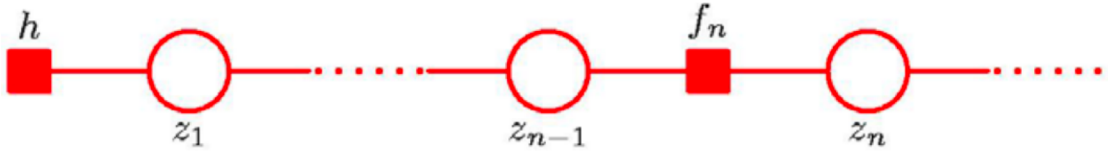
Viterbi Algorithm for HMM

- 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

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

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