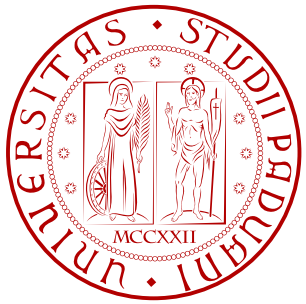
HIDDEN MARKOV MODELS (HMM) FOR MODELING DATA SEQUENCES

Michele Rossi
rossi@dei.unipd.it

Dept. of Information Engineering
University of Padova, IT



Outline

- What are they good for?
- Review of the i.i.d. case / Markov models
- Hidden Markov Models (HMMs)
  - The model / state transition diagrams / *trellis* diagrams
  - Generating sequences via HMMs
- Computational tools
  - Maximum likelihood for the HMM
  - Fordward/backward algorithm
  - Predictive distribution
  - Viterbi algorithm

HMM – what are they useful for?

- To model correlated temporal data, e.g., time series
  - Temporal correlation
  - i.i.d. models are no longer appropriate
- Used within countless application domains
  - Voice recognition
  - Online handwriting recognition
  - Keystroke dynamics based authentication
  - Sequences of proteins or DNA
  - Human motion analysis
  - Etc.

REVIEW OF THE I.I.D. CASE: GAUSSIAN MIXTURES

i.i.d. case – Gaussian mixtures (1/2)

- We use a latent variable (vector) **z**
- **z** is a K-binary random variable
  - **z** = $(z_1, z_2, \ldots, z_K)^T$
  - having a 1-of-K representation scheme
  - a single element z_k is equal to 1
  - all other elements z_j are equal to zero
- Hence, we have:

$$z_k \in \{0, 1\} \qquad \sum_{k=1}^{K} z_k = 1$$

- K possible states for **z** according to which element is 1

i.i.d. case – Gaussian mixtures (2/2)

- A Gaussian Mixture (GM) can be written in the form

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k G(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- Mixing coefficients π_k represent which element of **z** is 1

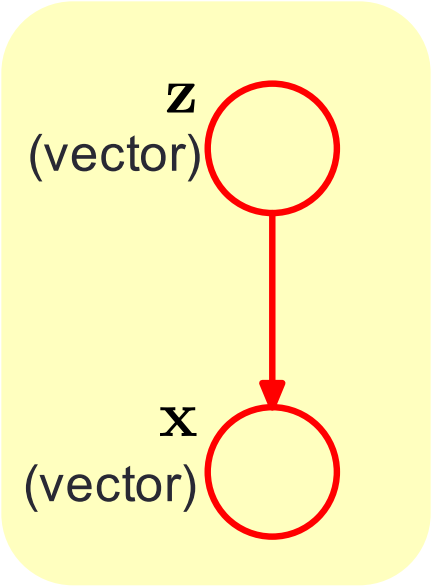
$$p(z_k = 1) = \pi_k$$

where:

$$0 \leq \pi_k \leq 1$$

$$\sum_{k=1}^{K} \pi_k = 1$$

Graphical model (1/2)



- Graphical representation of a mixture model
  - **z** is the latent variable ("internal model state")
  - **x** is the observed state (measured by an "observer")
- PMF of latent (inner) variable

$$p(\boldsymbol{z}) = \prod_{k=1}^{K} \pi_k^{z_k}$$

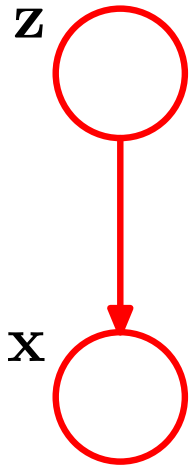
- Conditional PDF of **x** given a particular state for **z**

$$p(\boldsymbol{x}|z_k = 1) = G(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- That can be written in the more general form

$$p(\boldsymbol{x}|\boldsymbol{z}) = \prod_{k=1}^{K} G(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{z_k}$$

Graphical model (2/2)



- The join distribution of **x** and **z** is expressed as:

$$p(\boldsymbol{x}, \boldsymbol{z}) = p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})$$

- The distribution of **x** is obtained marginalizing
  - Summing the joint pdf over all possible values of **z**

$$p(\boldsymbol{x}) = \sum_{\boldsymbol{z}} p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z}) = \sum_{k=1}^{K} \pi_k G(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- Which is our Gaussian mixture
  - The inner state **z** (latent variable) *does not appear directly*
  - If we have several (i.i.d.) observations $\mathbf{x}_1$, $\mathbf{x}_2$, …, $\mathbf{x}_N$. Then, for each of them, there will be a corresponding latent variable $\mathbf{z}_1$, $\mathbf{z}_2$, …, $\mathbf{z}_N$

TRACKING TEMPORAL CORRELATION: HMM

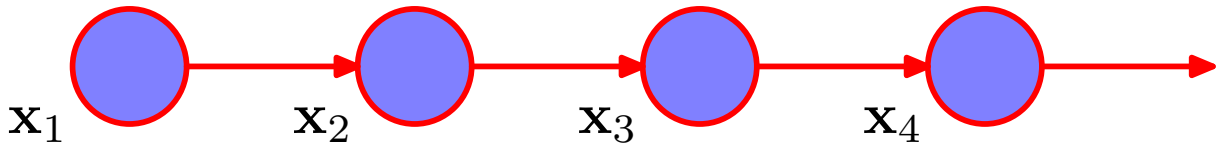
Markov models

- Consider now N observations of a phenomenon

$$\boldsymbol{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$$

  - We would like to track their temporal “structure” (correlation)
  - Moving beyond the i.i.d. model

- We use the powerful construct of Markov chains
  - In the diagram below, observation i depends on the previous one i-1
  - But it does not depend on previous observations i-2, i-3, i-4, ...
  - Memory of this model is limited to *one step in the past*



Product rule of probability

- 3 events A, B, C, use Bayes rule:

$$P(A|B,C) = \frac{P(A,B,C)}{P(B,C)}$$

- For B | C we have:

$$P(B|C) = \frac{P(B,C)}{P(C)}$$

- Putting things together, we get:

$$\begin{aligned} P(A,B,C) &= P(A|B,C)P(B,C) = \\ &= P(A|B,C)P(B|C)P(C) \end{aligned}$$

First order Markov models

- A complete model should track the joint distribution
  - This joint PDF can be written as (product rule of probability):

$$p(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N) = \prod_{n=1}^{N} p(\boldsymbol{x}_n | \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{n-1})$$

- This amounts to tracking a high amount of "memory"
  - Usually computationally too demanding
- First order Markov chain, fundamental assumption is:

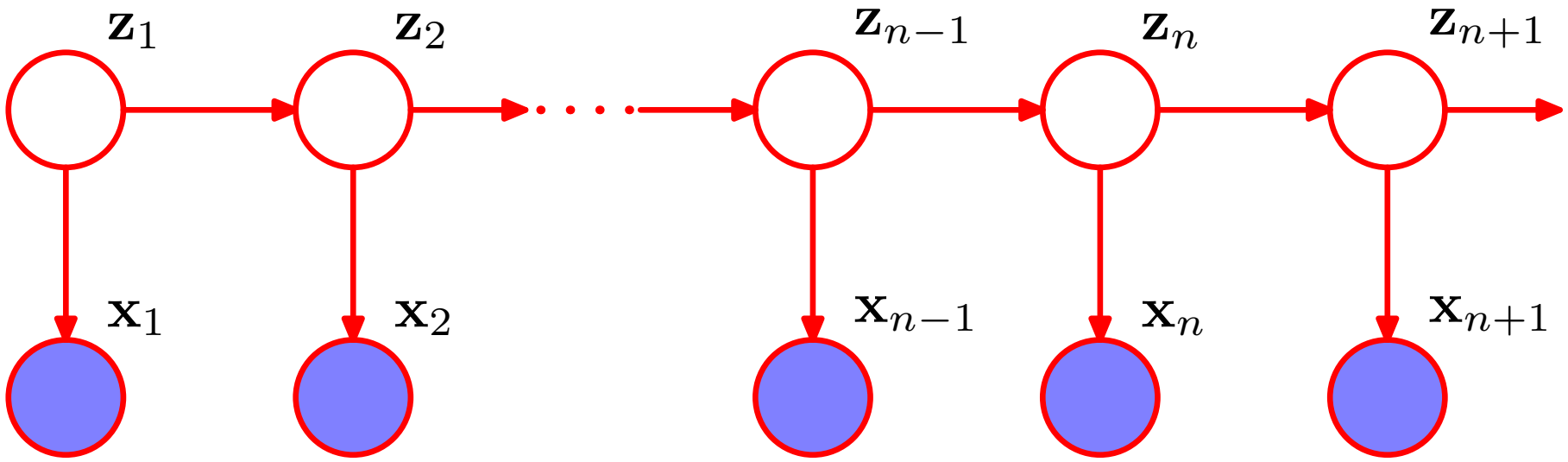
$$p(\boldsymbol{x}_n | \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{n-1}) = p(\boldsymbol{x}_n | \boldsymbol{x}_{n-1}) \,, \; \forall \, n \geq 2$$

- The joint PDF becomes:

$$p(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N) = p(\boldsymbol{x}_1) \prod_{n=2}^{N} p(\boldsymbol{x}_n | \boldsymbol{x}_{n-1})$$

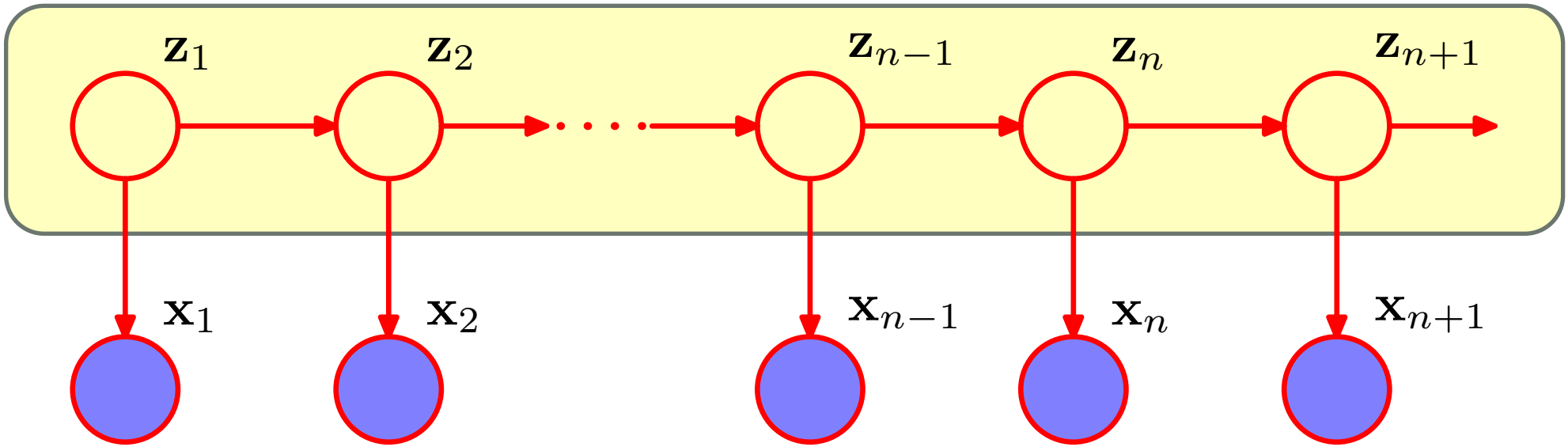
Hidden Markov Model (HMM) (1/5)

- Is a graphical model
- States: white filled circles
- Emissions or observations: blue filled circles
- Arrows: dependencies



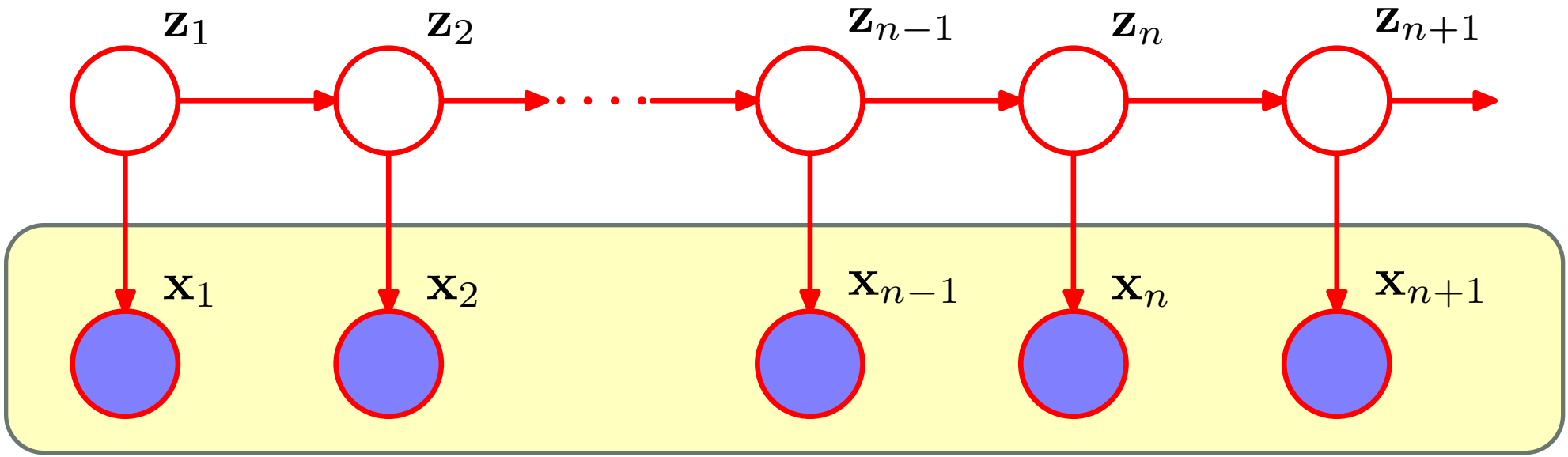
HMM (2/5)

- Hidden states (internal variables)
  - Each vector has a 1-of-K coding (K values)
  - Discrete and finite
  - Are also called “latent variables” $\mathbf{z}_i$



HMM (3/5)

- Observations (or emissions)
  - Represent the measured, *noisy* sequence
  - Can be either discrete or continuous r.v. (e.g., GM PDF)
  - Each observation $\mathbf{x}_i$ is conditioned on the corresponding variable $\mathbf{z}_i$
    - and does not depend on other observations



HMM (4/5)

- We allow $\mathbf{z}_n$ to depend on $\mathbf{z}_{n-1}$ through a conditional PDF

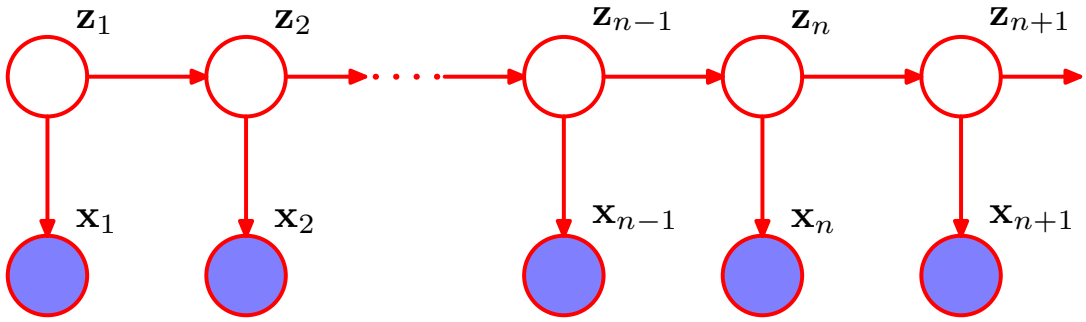
$$p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1})$$

- Considering the 1-of-K representation of $\mathbf{z}$
  - Conditional PDF is compactly represented through a matrix $\boldsymbol{A} = [A_{jk}]$
  - We have:

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

$$0 \leq A_{jk} \leq 1$$

$$\sum_{k=1}^{K} A_{jk} = 1\,,\ \forall\, j$$



  - Matrix **A** has K(K-1) *independent parameters*
  - And does not depend on n (stationarity)

HMM (5/5)

- We can write the conditional distribution explicitly as:

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

acting as selectors

- The initial latent node (state) $\mathbf{z}_1$ is "special" as it does not have a parent node. It has a marginal distribution represented by a vector of steady-state probabilities

$$\boldsymbol{\pi} = [\pi_1\ \pi_2\ \ldots\ \pi_K]^T \qquad \sum_{k=1}^{K}\pi_k = 1$$

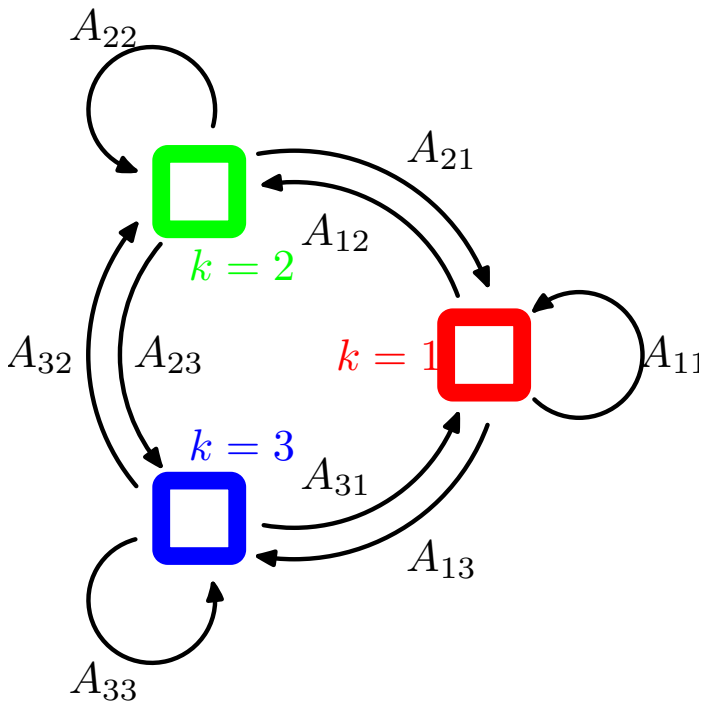
$$p(\boldsymbol{z}_1|\boldsymbol{\pi}) = \prod_{k=1}^{K}\pi_k^{z_{1k}}$$

State transition diagram - example

- Three states, K=3
  - Each latent variable $\mathbf{z}_i$ is: $\boldsymbol{z}_i = (z_{i1}, z_{i2}, z_{i3})^T$
  - 1-of-3 representation (only one element equal to 1)
  - State set is:

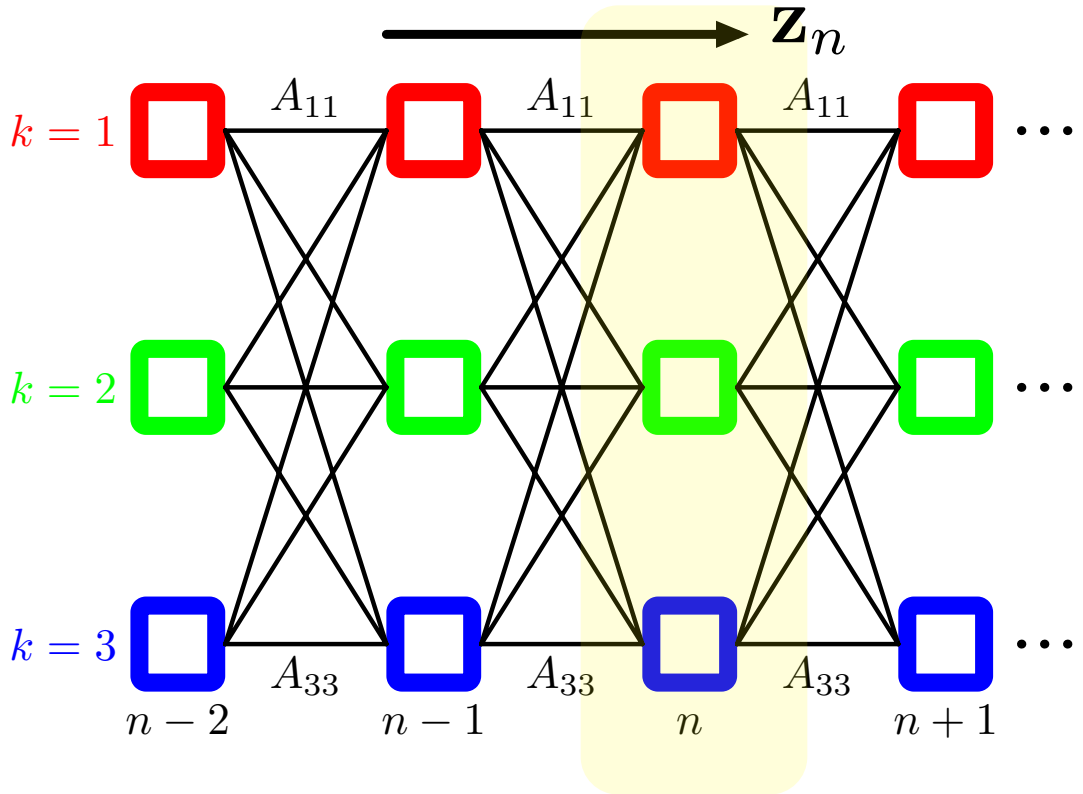
$$\mathcal{S} = \{(0,0,1)^T, (0,1,0)^T, (1,0,0)^T\}$$

- State transition diagram
  - Black lines represent *transition probabilities*
  - *States* are denoted by 1, 2 and 3 (boxes)
  - At every time step (time is discrete)
    - System moves (starting from current state)
    - Outcome can be:
      - Same state or any other state



Unfolding the transition diagram in time

- We obtain a *lattice* or *trellis*
- Each column corresponds to one state (latent) variable $\mathbf{z}_n$
- This representation is *hugely* important
  - E.g., to fit the model to the data

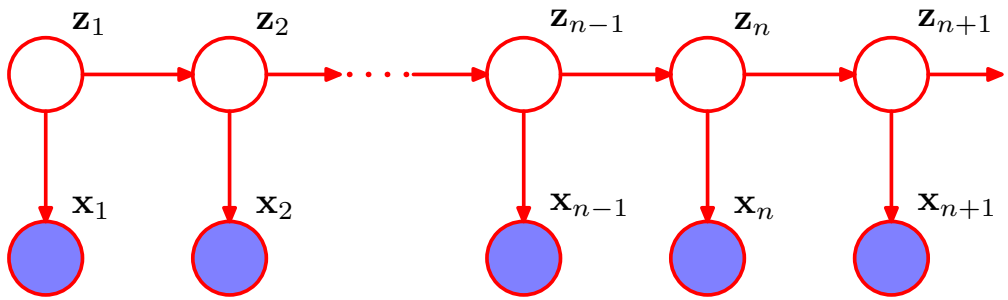


HMM – Emission Probabilities

- Are the conditional probabilities of $\mathbf{x}_n$ given $\mathbf{z}_n$: $p(\boldsymbol{x}_n|\boldsymbol{z}_n,\phi)$
- ϕ is the set containing the parameters governing the model
- This PDF can be:
  - Gaussian (mixture), e.g., if vector **x** has continuous elements
  - Probability tables if **x** is discrete, mapped by a neural network, ...
- Emission probabilities are compactly written as:

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

acts as a selector for the right PDF (for which $z_{nk}=1$)

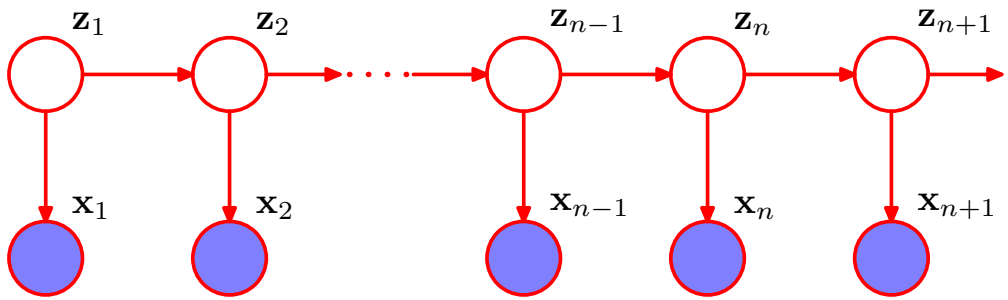


HMM – Emission Probabilities

- Are the conditional probabilities of $\mathbf{x}_n$ given $\mathbf{z}_n$: $p(\boldsymbol{x}_n|\boldsymbol{z}_n,\phi)$
- ϕ is the set containing the parameters governing the model
- This PDF can be:
  - Gaussian (mixture), e.g., if vector **x** has continuous elements
  - Probability tables if **x** is discrete, mapped by a neural network, ...
- Emission probabilities are compactly written as:

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

parameters associated with state k, i.e., z_{nk}=1



Remarks

- At time n, once $\mathbf{z}_n$ is known
  - $\mathbf{x}_{n+1}, \ldots, \mathbf{x}_N$ (future data) are *independent of* $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ (past data)
- Hence, conditional independence property C1:

$$p(\boldsymbol{X}|\boldsymbol{z}_n) = p(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N|\boldsymbol{z}_n) =$$
$$= p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n|\boldsymbol{z}_n) \times p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N|\boldsymbol{z}_n)$$

- If we do not know the value of latent variables
  - The *predictive distribution* of $\mathbf{x}_{n+1}$, given all the previous observations *does not exhibit* any conditional independence property:

$$p(\boldsymbol{x}_{n+1}|\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n)$$

- In this case, $\mathbf{x}_{n+1}$ depends on all previous observations

Joint PDF over latent and observed variables

- Let us consider N observations
- We define the sets: $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$ observations

$\boldsymbol{Z} = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_N\}$ latent variables

$\boldsymbol{\theta} = \{\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{\phi}\}$ HMM parameters

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

system evolution (state transitions) $P(\boldsymbol{Z}|\boldsymbol{\theta})$

$P(\boldsymbol{X}|\boldsymbol{Z}, \boldsymbol{\theta})$ emission probs

Termed the complete (**X** and **Z**) data likelihood (*latent* and *observation* variables)

HMM formalism

- Let us consider N observations

$$\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\} \quad \text{observations}$$

$$\boldsymbol{Z} = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_N\} \quad \text{latent variables}$$

$$\boldsymbol{\theta} = \{\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{\phi}\} \quad \text{HMM parameters}$$

- An HMM model is represented by the tuple:

$$\mathcal{M} = \{\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{\phi}\}$$

Training an HMM

- **Given** an observation sequence $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$
- **Find** the HMM parameters that best represent the observed data
- Question to answer: given an observation sequence and set of possible models, which model most closely fits the data?
- The parameters to learn are:

$$\boldsymbol{\theta} = \{\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{\phi}\}$$

Inference in an HMM

- Given an observation sequence $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$
- Given a pre-trained HMM
- Compute the most likely hidden state sequence

Generating a data sequence with HMMs

- Pick the initial latent variable $\mathbf{z}_1$
  - Sample $\mathbf{z}_1$ from distribution $\boldsymbol{\pi}$
  - Sample $\mathbf{x}_1$ from $p(\mathbf{x}_1 \mid \mathbf{z}_1, \boldsymbol{\phi})$
- Choose the next state variable $\mathbf{z}_2$ given $\mathbf{z}_1$
  - Suppose that the value of $\mathbf{z}_1$ is state j
  - Then, we move to the next state k with probability A_{jk}
  - Once we know $\mathbf{z}_2$ we draw a sample for $\mathbf{x}_2$
- Repeated this procedure until we get to $(\mathbf{z}_N, \mathbf{x}_N)$

Example

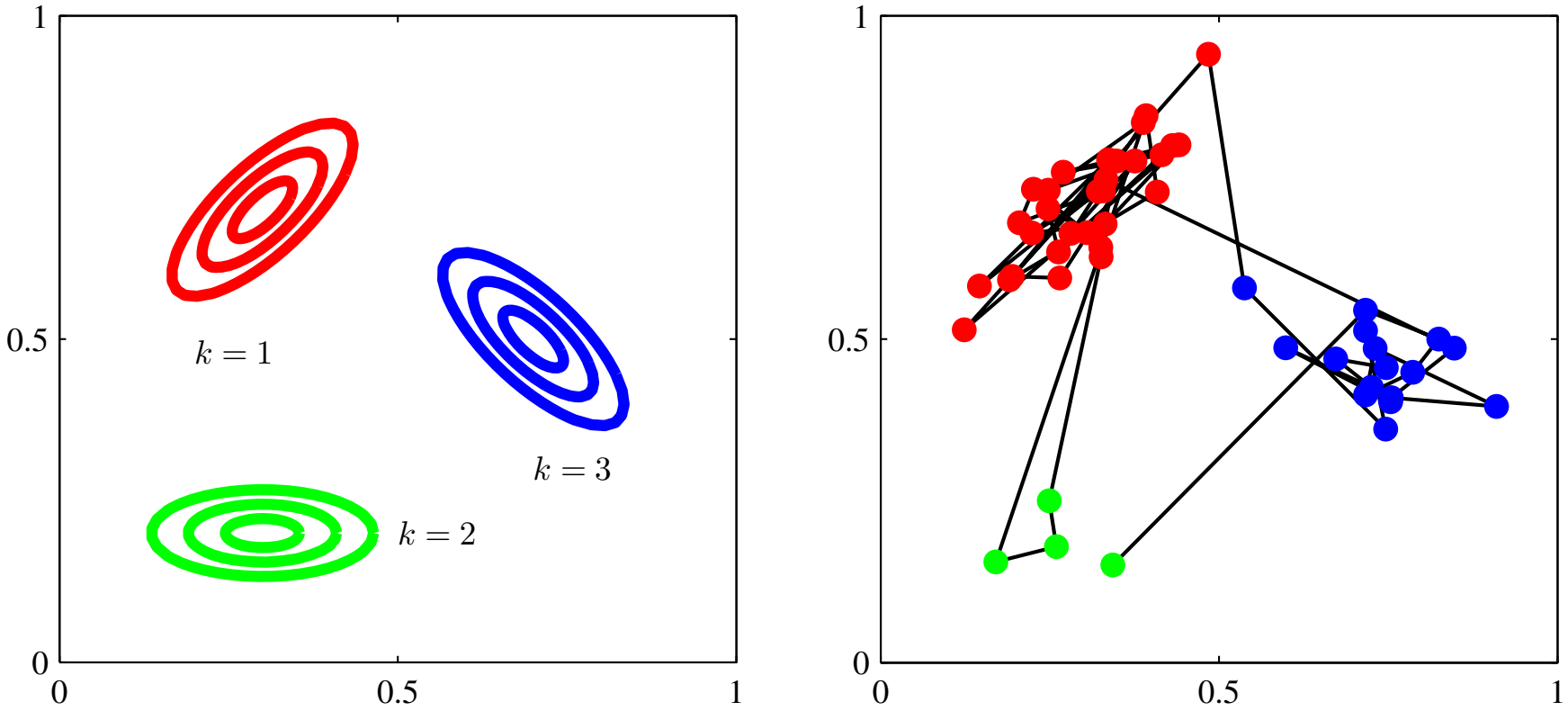
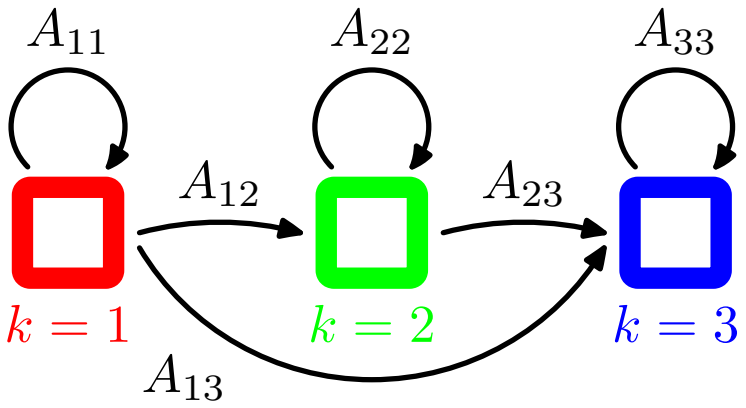


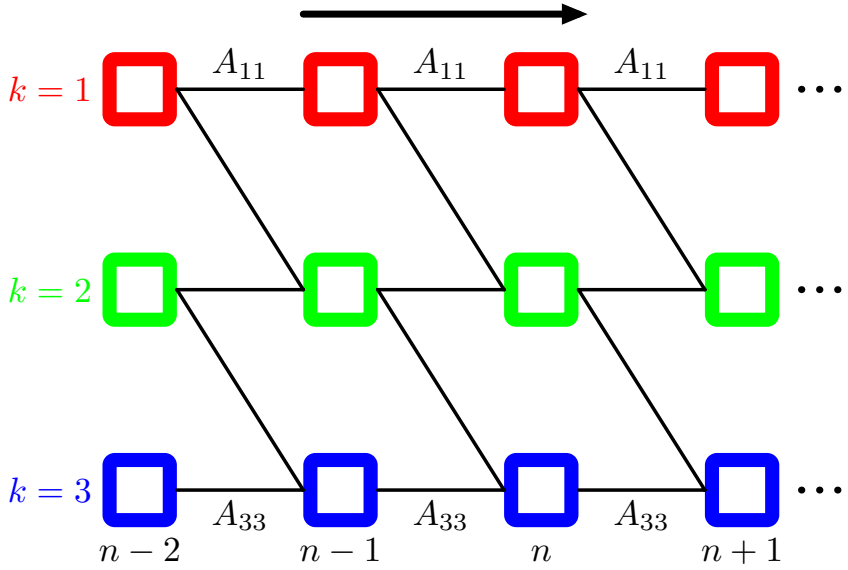
Illustration of sampling from a hidden Markov model having a 3-state latent variable $\mathbf{z}$ and a Gaussian emission model $p(\mathbf{x}|\mathbf{z})$ where $\mathbf{x}$ is 2-dimensional. (a) Contours of constant probability density for the emission distributions corresponding to each of the three states of the latent variable. (b) A sample of 50 points drawn from the hidden Markov model, colour coded according to the component that generated them and with lines connecting the successive observations. Here the transition matrix was fixed so that in any state there is a 5% probability of making a transition to each of the other states, and consequently a 90% probability of remaining in the same state.

Constraints on transition matrix

- Imposing constraints on the transition matrix **A** leads to many HMM variants…

Example 1: let-to-right HMM
state id cannot decrease





Example 2: let-to-right HMM

state is constrained to grow by at most 1 unit at each iteration

MAXIMUM LIKELIHOOD FOR THE HMM

Maximum likelihood for the HMM

- The data $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$
- We *could* marginalize and obtain, the data likelihood:

$$p(\boldsymbol{X}|\boldsymbol{\theta}) = \sum_{\boldsymbol{Z}} p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})$$

- Big problem ahead…
  - For each time step n = 1, 2, …,N
  - The latent variable $\mathbf{z}_n$ can take K values
  - There are a total of K^N possible paths to analyze
  - The complexity associated with this computation is exponential
  - Moreover, the above probability distribution will in general be a combination of Gaussian mixtures, whose maximization is *complex* and impossible in close-form

Expectation Maximization (EM)

- We use the EM algorithm
  - To devise an *efficient framework* to maximize the likelihood of an HMM
  - This allows *finding the parameters* $\boldsymbol{\theta}$ *that best represent the data*
- EM algorithm start with an initial selection of the parameters $\boldsymbol{\theta}^{\text{old}}$
- In the E step: we take these old parameter values
  - Compute the *posterior* distribution of the latent variables $p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{\text{old}})$
  - We then use this posterior to evaluate the expectation of the complete data log-likelihood function, as a function of the parameters $\boldsymbol{\theta}$:

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

- Next step (M step): given the *posterior distribution*, maximize Q to find the new parameters

Notation (1/2)

- Marginal (first-order) posterior distribution of a latent variable $\mathbf{z}_n$:

$$\gamma(\boldsymbol{z}_n) \stackrel{\Delta}{=} p(\boldsymbol{z}_n|\boldsymbol{X}, \boldsymbol{\theta}^{\text{old}})$$

- Joint (second-order) posterior distribution of two subsequent latent variables (times n-1 and n):

$$\xi(\boldsymbol{z}_{n-1}, \boldsymbol{z}_n) \stackrel{\Delta}{=} p(\boldsymbol{z}_{n-1}, \boldsymbol{z}_n|\boldsymbol{X}, \boldsymbol{\theta}^{\text{old}})$$

Notation (2/2)

- Conditional probability that $z_{nk}=1$ (given observations $\mathbf{X}$ and parameters $\boldsymbol{\theta}$)

$$p(z_{nk}=1|\boldsymbol{X},\boldsymbol{\theta}) \overset{(1)}{=} E[z_{nk}] = \sum_{\boldsymbol{z}_n} z_{nk} p(\boldsymbol{z}_n|\boldsymbol{X},\boldsymbol{\theta}^{\text{old}}) \triangleq \gamma(z_{nk})$$

(1) holds as the expectation of a binary random variable corresponds to the probability that it takes the value 1 (same as expectation of an indicator function)

- Analogously, the conditional prob. that elements $z_{n-1,j}$ and z_{nk} are both 1 is:

$$p(z_{n-1,j}=1, z_{nk}=1|\boldsymbol{X},\boldsymbol{\theta}) = E[z_{n-1,j} z_{nk}] =$$

$$= \sum_{\boldsymbol{z}_{n-1},\boldsymbol{z}_n} p(\boldsymbol{z}_{n-1},\boldsymbol{z}_n|\boldsymbol{X},\boldsymbol{\theta}^{\text{old}}) z_{n-1,j} z_{nk} \triangleq \xi(z_{n-1,j}, z_{nk})$$

MAXIMUM LIKELIHOOD: M STEP

M step: rewriting Q

Using

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

into the following expression

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

leads to (using the quantities defined in the previous two slides):

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

$$+ \sum_{n=1}^{N} \sum_{k=1}^{N} \gamma(z_{nk}) \ln p(\boldsymbol{x}_n|\phi_k)$$

M step: overview

Maximize Q, given the old parameters

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

$$+ \sum_{n=1}^{N} \sum_{k=1}^{K} \gamma(z_{nk}) \ln p(\boldsymbol{x}_n | \phi_k)$$

as a function of:

$$\boldsymbol{\pi} = (\pi_1, \pi_2, \ldots, \pi_K)^T \quad \text{(mixture probabilities)}$$

$$\boldsymbol{A} = [A_{jk}] \quad \text{(KxK transition prob. maxtrix)}$$

$$\phi_k = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}, \; k = 1, \ldots, K \quad \text{(emission PDF parameters)}$$

M step (maximize over π and $\mathbf{A}$)

- The goal is to maximize $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}})$ with respect to the parameters $\boldsymbol{\theta} = \{\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{\phi}\}$
- The terms $\gamma(z_{nk})$ and $\xi(z_{n-1,j}, z_{nk})$ are treated as constants

- Using Lagrange multipliers, we promptly get the result:

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

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

M step (maximize over ϕ_k)

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

$$+ \sum_{n=1}^{N} \sum_{k=1}^{N} \gamma(z_{nk}) \ln p(\boldsymbol{x}_n | \phi_k)$$

- The first two terms are constant wrt ϕ_k
- Only the third term matters
- It is the weighted log-likelihood of the emission densities $p(\boldsymbol{x}_n | \phi_k)$
- **NOTE:** if these densities are Gaussian (as in the Gaussian mixture case), the maximization of the third term with respect to means and covariance matrices is *exactly the same* that we have seen when maximizing the log-likelihood for *soft K-means*

M step (maximize over ϕ_k)

- For Gaussian emission probabilities
- We have:

$$p(\boldsymbol{x}_n|\phi_k) = G(\boldsymbol{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- and maximizing the Q function returns:

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk})\boldsymbol{x}_n}{\sum_{n=1}^{N} \gamma(z_{nk})}$$

$$\boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk})(\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^{N} \gamma(z_{nk})}$$

MAXIMUM LIKELIHOOD: E STEP

Remark

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

$$+ \sum_{n=1}^{N} \sum_{k=1}^{N} \gamma(z_{nk}) \ln p(\boldsymbol{x}_n | \phi_k)$$

the goal of the E step is to evaluate the quantities:

$$\gamma(z_{nk}) \text{ and } \xi(z_{n-1,j}, z_{nk})$$

EFFICIENTLY

see next, this is achieved through the *forward/backward algorithm*

Function $\gamma(\boldsymbol{z}_n)$ (1/2)

- To simplify notation: we omit the condition on $\boldsymbol{\theta}^{\text{old}}$
- We begin evaluating $\gamma(\boldsymbol{z}_n)$
- From its own definition, we have that:

$$\gamma(\boldsymbol{z}_n) \overset{(1)}{=} p(\boldsymbol{z}_n|\boldsymbol{X}) \overset{(2)}{=} \frac{p(\boldsymbol{X}|\boldsymbol{z}_n)p(\boldsymbol{z}_n)}{p(\boldsymbol{X})}$$

- (1) is as we omit the condition on $\boldsymbol{\theta}^{\text{old}}$, for (2) we use Bayes' rule
- Note that:

$$p(\boldsymbol{X}) = p(\boldsymbol{X}|\boldsymbol{\theta}^{\text{old}}) = p(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N|\boldsymbol{\theta}^{\text{old}})$$

- It is the likelihood of the data (observations) given the parameters

Function $\gamma(\boldsymbol{z}_n)$ (2/2)

- We have got

$$\gamma(\boldsymbol{z}_n) = \frac{p(\boldsymbol{X}|\boldsymbol{z}_n)p(\boldsymbol{z}_n)}{p(\boldsymbol{X})} = \frac{p(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N|\boldsymbol{z}_n)p(\boldsymbol{z}_n)}{p(\boldsymbol{X})}$$

- Using the conditional independence property C1

$$\gamma(\boldsymbol{z}_n) = \frac{p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n|\boldsymbol{z}_n)p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N|\boldsymbol{z}_n)p(\boldsymbol{z}_n)}{p(\boldsymbol{X})} =$$

$$= \frac{p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{z}_n)p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N|\boldsymbol{z}_n)}{p(\boldsymbol{X})}$$

- We define:

$$\alpha(\boldsymbol{z}_n) \triangleq p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{z}_n)$$

$$\beta(\boldsymbol{z}_n) \triangleq p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N|\boldsymbol{z}_n)$$

Compactly, we obtain:

$$\begin{aligned}\gamma(\boldsymbol{z}_n) &= p(\boldsymbol{z}_n|\boldsymbol{X}) \\ &= \frac{p(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n,\boldsymbol{z}_n)p(\boldsymbol{x}_{n+1},\ldots,\boldsymbol{x}_N|\boldsymbol{z}_n)}{p(\boldsymbol{X})} \\ &= \frac{\alpha(\boldsymbol{z}_n)\beta(\boldsymbol{z}_n)}{p(\boldsymbol{X})}\end{aligned}$$

- With:

$$\alpha(\boldsymbol{z}_n) \triangleq p(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n,\boldsymbol{z}_n)$$

$$\beta(\boldsymbol{z}_n) \triangleq p(\boldsymbol{x}_{n+1},\ldots,\boldsymbol{x}_N|\boldsymbol{z}_n)$$

E STEP
FORWARD-BACKWARD ALGORITHM

What is it?

- It is an efficient computational technique
- Uses message passing (a recursive approach)

- Computation moves
  - Forward from time 1 to time N → $\alpha(\boldsymbol{z}_n) \triangleq p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{z}_n)$
  - Backward from time N to time 1 → $\beta(\boldsymbol{z}_n) \triangleq p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_n)$

More independence properties

- Given $\mathbf{z}_n$, observation $\mathbf{x}_n$ is independent of $\mathbf{x}_1$, $\mathbf{x}_2$, ..., $\mathbf{x}_{n-1}$
- Conditional independence property C2

$$p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{x}_n, \boldsymbol{z}_n) = p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_n)$$

- For $\mathbf{x}_1$, $\mathbf{x}_2$, ..., $\mathbf{x}_{n-1}$, knowing $\mathbf{z}_n$ is useless if we already know $\mathbf{z}_{n-1}$
- Conditional independence property C3

$$p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_{n-1}, \boldsymbol{z}_n) = p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_{n-1})$$

FORWARD recursion for $\alpha(\boldsymbol{z}_n)$

$$\alpha(\boldsymbol{z}_n) = p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{z}_n) \overset{(1)}{=} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{z}_n) =$$

$$\overset{(2)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_n) p(\boldsymbol{z}_n) = p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(3)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(4)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(5)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(6)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) =$$

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

We start from the definition of α and in (1) use Bayes' rule

$$\alpha(\boldsymbol{z}_n) = p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{z}_n) \overset{(1)}{=} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{z}_n) =$$

$$\overset{(2)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_n) p(\boldsymbol{z}_n) = p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(3)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(4)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(5)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(6)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) =$$

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

(2) use property C2, i.e., the fact that $\mathbf{x}_n$ is independent of previous observations $\mathbf{x}_1$, $\mathbf{x}_2$, …, $\mathbf{x}_{n-1}$, once $\mathbf{z}_n$ is known

$$\alpha(\boldsymbol{z}_n) = p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{z}_n) \overset{(1)}{=} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{z}_n) =$$

$$\overset{(2)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_n) p(\boldsymbol{z}_n) = p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(3)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(4)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(5)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(6)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) =$$

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

(3) add variable $\mathbf{z}_{n-1}$ and marginalize with respect to it

$$\alpha(\boldsymbol{z}_n) = p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{z}_n) \overset{(1)}{=} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{z}_n) =$$

$$\overset{(2)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_n) p(\boldsymbol{z}_n) = p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(3)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(4)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(5)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(6)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) =$$

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

(4) use Bayes' rule P(A,B,C) = P(A,C|B) P(B)

$$\alpha(\boldsymbol{z}_n) = p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{z}_n) \overset{(1)}{=} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{z}_n) =$$

$$\overset{(2)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_n) p(\boldsymbol{z}_n) = p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(3)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(4)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(5)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(6)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) =$$

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

(5) use property C3: $\mathbf{z}_n$ is independent of $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{n-1}$, if $\mathbf{z}_{n-1}$ is known, so we can split the joint pdf into two terms: one depends on $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{n-1}$, the other on $\mathbf{z}_n$

$$\alpha(\boldsymbol{z}_n) = p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{z}_n) \overset{(1)}{=} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{z}_n) =$$

$$\overset{(2)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_n) p(\boldsymbol{z}_n) = p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(3)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(4)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(5)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(6)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) =$$

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

(6) use Bayes' rule for conditional probabilities

$$\alpha(\boldsymbol{z}_n) = p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{z}_n) \overset{(1)}{=} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{z}_n) =$$

$$\overset{(2)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_n) p(\boldsymbol{z}_n) = p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(3)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(4)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(5)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(6)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) =$$

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

- Recognize that the term inside the summation is α itself
- This leads to a recursion!!!

$$\alpha(\boldsymbol{z}_n) = p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{z}_n) \overset{(1)}{=} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{z}_n) =$$

$$\overset{(2)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_n) p(\boldsymbol{z}_n) = p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(3)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}, \boldsymbol{z}_n) =$$

$$\overset{(4)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(5)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1} | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_{n-1}) =$$

$$\overset{(6)}{=} p(\boldsymbol{x}_n | \boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{z}_{n-1}) p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1}) =$$

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

Recursion for $\alpha(\boldsymbol{z}_n)$: forward recursion illustration

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

- Notation: with $z_{n,k}$ we mean state k at time n (k=1, 2, …,K):

$$z_{n,1} \Rightarrow \boldsymbol{z}_n|_{z_{n,1}=1} = (1, 0, \ldots, 0)^T$$

$$\ldots$$

$$z_{n,k} \Rightarrow \boldsymbol{z}_n|_{z_{n,k}=1} = (0, \ldots, 0, 1, 0, \ldots, 0)^T$$

k-th entry is 1

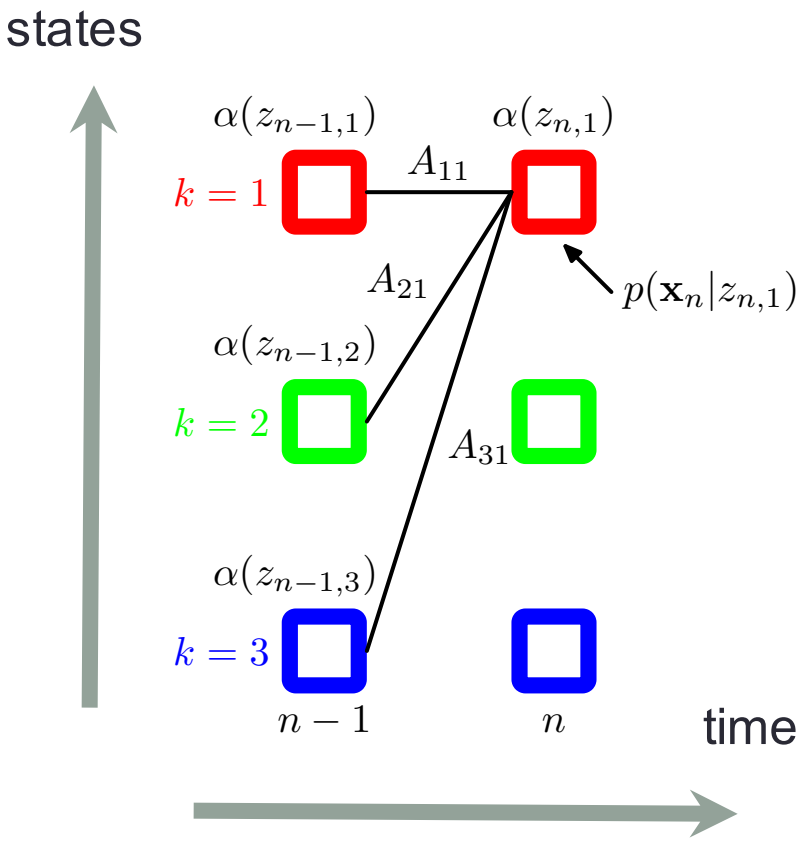
$$\ldots$$

- For K=3, we have:

$$\alpha(z_{n,k}) = p(\boldsymbol{x}_n|z_{n,k}) \sum_{j=1}^{3} \alpha(z_{n-1,j}) p(z_{n,k}|z_{n-1,j})$$

$$\alpha(z_{n,k}) = p(\boldsymbol{x}_n|z_{n,k}) \sum_{j=1}^{3} \alpha(z_{n-1,j}) p(z_{n,k}|z_{n-1,j}) =$$

$$= p(\boldsymbol{x}_n|z_{n,k}) \sum_{j=1}^{3} \alpha(z_{n-1,j}) A_{jk}$$



Forward recursion (k=1): illustration of the computation of $\alpha(z_{n,1})$. It is obtained by:

(i) multiplying the values of α at the previous step $\alpha(z_{n-1,j})$ by the HMM transition probabilities A_{j1} (from state j in previous step n-1 to state 1 in current step n) and

(ii) summing up these contributions *and* multiplying the result by the emission probability $p(\mathbf{x}_n|z_{n,1})$. $\mathbf{x}_n$ is the data vector observed at the current step n.

The same procedure is repeated for each state k=1,2,3 in the current step n.

It is a forward message passing algorithm.

FORWARD recursion - remarks

- Initialization: to start the forward recursion, we need an initial condition, that is given by:

$$\alpha(\boldsymbol{z}_1) = p(\boldsymbol{x}_1, \boldsymbol{z}_1) = p(\boldsymbol{z}_1)p(\boldsymbol{x}_1|\boldsymbol{z}_1) = \prod_{k=1}^{K} \left[\pi_k p(\boldsymbol{x}_1|\phi_k)\right]^{z_{1k}}$$

- Vector form: $\alpha(\boldsymbol{z}_1) = (\alpha(z_{11}), \alpha(z_{12}), \ldots, \alpha(z_{1K}))^T$
  - Remember: 1-of-K representation
  - This Eq. tells us that: $\alpha(z_{1,k}) = \pi_k p(\boldsymbol{x}_1|\phi_k)$

- Complexity: each step of the forward recursion implies, for each state in step n (K states): (i) we need to sum K terms and this has to be repeated for (ii) all N time steps (number of observations). The overall complexity is $O(K^2 N)$

More independence properties

- Given $\mathbf{z}_{n+1}$, observations $\mathbf{x}_{n+1}$, ..., $\mathbf{x}_N$ are independent of $\mathbf{z}_n$
- Conditional independence property C4

$$p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_n, \boldsymbol{z}_{n+1}) = p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_{n+1})$$

- For $\mathbf{x}_{n+2}$, ..., $\mathbf{x}_N$, depend on $\mathbf{z}_{n+1}$ but are independent of $\mathbf{x}_{n+1}$
- Conditional independence property C5

$$p(\boldsymbol{x}_{n+2}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_{n+1}, \boldsymbol{x}_{n+1}) = p(\boldsymbol{x}_{n+2}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_{n+1})$$

BACKWARD recursion for $\beta(\boldsymbol{z}_n)$

$$\beta(\boldsymbol{z}_n) = p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_n) \overset{(1)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N, \boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

$$\overset{(2)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_n, \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

$$\overset{(3)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

$$\overset{(4)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+2}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_{n+1}) p(\boldsymbol{x}_{n+1} | \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

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

(1) use definition of β and marginalize over $\mathbf{z}_{n+1}$

$$\beta(\boldsymbol{z}_n) = p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_n) \overset{(1)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N, \boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

$$\overset{(2)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_n, \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

$$\overset{(3)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

$$\overset{(4)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+2}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_{n+1}) p(\boldsymbol{x}_{n+1} | \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

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

(2) use Bayes' rule

$$\beta(\boldsymbol{z}_n) = p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_n) \overset{(1)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N, \boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

$$\overset{(2)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_n, \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

$$\overset{(3)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

$$\overset{(4)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+2}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_{n+1}) p(\boldsymbol{x}_{n+1} | \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

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

(3) use independence property C4

$$\beta(\boldsymbol{z}_n) = p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_n) \overset{(1)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N, \boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

$$\overset{(2)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_n, \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

$$\overset{(3)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

$$\overset{(4)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+2}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_{n+1}) p(\boldsymbol{x}_{n+1} | \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

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

(4) use independence property C5: so we can split the PDF into two terms one depends on $\mathbf{x}_{n+2}, \ldots, \mathbf{x}_N$ and one on $\mathbf{x}_{n+1}$, given $\mathbf{z}_{n+1}$

$$\beta(\boldsymbol{z}_n) = p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_n) \overset{(1)}{=} \sum_{\boldsymbol{z}_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N, \boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

$$\overset{(2)}{=} \sum_{\boldsymbol{z}_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_n, \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

$$\overset{(3)}{=} \sum_{\boldsymbol{z}_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

$$\overset{(4)}{=} \sum_{\boldsymbol{z}_{n+1}} p(\boldsymbol{x}_{n+2}, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_{n+1}) p(\boldsymbol{x}_{n+1} | \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1} | \boldsymbol{z}_n)$$

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

Final result: use definition of β and obtain recursion relation

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

$$\beta(\boldsymbol{z}_n) = p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N|\boldsymbol{z}_n) \overset{(1)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N, \boldsymbol{z}_{n+1}|\boldsymbol{z}_n)$$

$$\overset{(2)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N|\boldsymbol{z}_n, \boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1}|\boldsymbol{z}_n)$$

$$\overset{(3)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N|\boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1}|\boldsymbol{z}_n)$$

$$\overset{(4)}{=} \sum_{z_{n+1}} p(\boldsymbol{x}_{n+2}, \ldots, \boldsymbol{x}_N|\boldsymbol{z}_{n+1}) p(\boldsymbol{x}_{n+1}|\boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1}|\boldsymbol{z}_n)$$

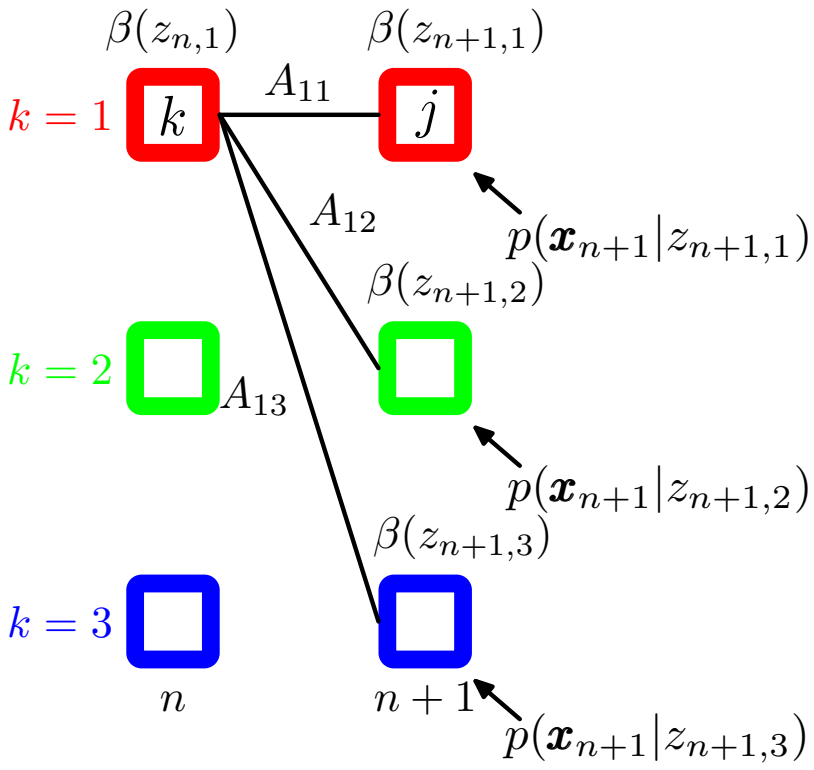
$$= \sum_{z_{n+1}} \beta(\boldsymbol{z}_{n+1}) p(\boldsymbol{x}_{n+1}|\boldsymbol{z}_{n+1}) p(\boldsymbol{z}_{n+1}|\boldsymbol{z}_n)$$

BACKWARD recursion - illustration

$$\beta(z_{n,k}) = \sum_{j=1}^{3} \beta(z_{n+1,j}) p(\boldsymbol{x}_{n+1}|z_{n+1,j}) p(z_{n+1,j}|z_{n,k}) =$$

$$= \sum_{j=1}^{3} \beta(z_{n+1,j}) p(\boldsymbol{x}_{n+1}|z_{n+1,j}) A_{kj}$$

- Same notation $z_{n,k}$ as before
- Iterate
  - for each state k=1,…,K
  - for each n=N,N-1,...,1
- Complexity $O(K^2 N)$
  - K sums for K states (K^2)
  - Repeat for N time steps



BACKWARD recursion - initialization

- We need an initial value for the recursion $\beta(\boldsymbol{z}_N) = ?$
- 1) Take (see def)

$$\gamma(\boldsymbol{z}_n) = \frac{p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{z}_n)\beta(\boldsymbol{z}_n)}{p(\boldsymbol{X})}$$

- 2) Set n=N
- 3) From 1) and 2) we obtain:

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

- 4) In order for this equality to hold it must be:

$$\beta(\boldsymbol{z}_N) = 1\,,\, \forall\, \boldsymbol{z}_N$$

Computing the data likelihood (1/2)

- A quantity of interest is the data likelihood

$$p(\boldsymbol{X}) = p(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N)$$

- From the previous definitions we recall that:

$$\gamma(\boldsymbol{z}_n) = p(\boldsymbol{z}_n|\boldsymbol{X}) = \frac{\alpha(\boldsymbol{z}_n)\beta(\boldsymbol{z}_n)}{p(\boldsymbol{X})}$$

- Summing both sides (the LHS is a normalized PDF)

$$1 = \sum_{\boldsymbol{z}_n} p(\boldsymbol{z}_n|\boldsymbol{X}) = \frac{\sum_{\boldsymbol{z}_n} \alpha(\boldsymbol{z}_n)\beta(\boldsymbol{z}_n)}{p(\boldsymbol{X})}$$

Computing the data likelihood (2/2)

- We have obtained

$$p(\boldsymbol{X}) = \sum_{\boldsymbol{z}_n} \alpha(\boldsymbol{z}_n)\beta(\boldsymbol{z}_n)$$

- Hence, p(**X**) can be computed for any convenient choice of $\mathbf{z}_n$
- For instance, taking n=N
  - We can simply run the forward recursion and compute $\alpha(\mathbf{z}_N)$
  - $\beta(\mathbf{z}_N)$ is not needed as $\beta(\mathbf{z}_N)=1$ for all $\mathbf{z}_N$

- This leads to:

$$p(\boldsymbol{X}) = \sum_{\boldsymbol{z}_N} \alpha(\boldsymbol{z}_N)$$

Important remark

- We have $$p(\boldsymbol{X}) = \sum_{\boldsymbol{z}_N} \alpha(\boldsymbol{z}_N) \quad (***)$$
- The *naïve* procedure entails computing the joint PDF $p(\boldsymbol{X}, \boldsymbol{Z})$
- And summing over all possible choices for **Z**

$$p(\boldsymbol{X}) = \sum_{\boldsymbol{Z}} p(\boldsymbol{X}, \boldsymbol{Z})$$

- Note that, each of such choices corresponds to a specific path of hidden states $\mathbf{z}_n$, for time steps n=1,2,…,N
- Every term in the sum is a path in the trellis, since at each time step there are K hidden states and there are N time steps, we have a total of K^N possible paths (elements in the sum) → exponential complexity $O(K^N)$ *vs* complexity of (***) above is $O(K^3 N)$

M step wrap up (1/2)

- For the means we have:

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^{N} \gamma(z_{n,k})\boldsymbol{x}_n}{\sum_{n=1}^{N} \gamma(z_{n,k})} = \frac{\sum_{n=1}^{N} \alpha(z_{n,k})\beta(z_{n,k})\boldsymbol{x}_n}{\sum_{n=1}^{N} \alpha(z_{n,k})\beta(z_{n,k})}$$

- The data likelihood p(**X**) cancels out
- Recalling that:

$$\gamma(z_{n,k}) = \frac{\alpha(z_{n,k})\beta(z_{n,k})}{p(\boldsymbol{X})}$$

- This applies to all the other quantities of interest

M step wrap up (2/2)

- Moreover, we can write:

$$\begin{aligned}\xi(\boldsymbol{z}_{n-1}, \boldsymbol{z}_n) &= p(\boldsymbol{z}_{n-1}, \boldsymbol{z}_n|\boldsymbol{X}) \\ &= \frac{p(\boldsymbol{X}|\boldsymbol{z}_{n-1}, \boldsymbol{z}_n)p(\boldsymbol{z}_{n-1}, \boldsymbol{z}_n)}{P(\boldsymbol{X})} \\ &= \frac{p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}|\boldsymbol{z}_{n-1})p(\boldsymbol{x}_n|\boldsymbol{z}_n)p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N|\boldsymbol{z}_n)p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1})p(\boldsymbol{z}_{n-1})}{P(\boldsymbol{X})} \\ &= \frac{\alpha(\boldsymbol{z}_{n-1})p(\boldsymbol{x}_n|\boldsymbol{z}_n)p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1})\beta(\boldsymbol{z}_n)}{P(\boldsymbol{X})}\end{aligned}$$

- We have used - conditional independence property C6:

$$p(\boldsymbol{X}|\boldsymbol{z}_{n-1}, \boldsymbol{z}_n) = p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}|\boldsymbol{z}_{n-1})p(\boldsymbol{x}_n|\boldsymbol{z}_n)p(\boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_N|\boldsymbol{z}_n)$$

- Along with the definition of α and β
- Hence, all the other quantities can be obtained using α and β

HMM: outline of the EM-based training algorithm

1) Make an initial selection of parameters $\boldsymbol{\theta}^{\text{old}}$: $\boldsymbol{\theta} = \{\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{\phi}\}$

Initial selection: usually π and **A** are selected uniformly at random (meeting their non-negativity and summation constraints). Initialization of ϕ depends on the form of the emission PDF. For Gaussians, means and covariance matrices can be initialized using, e.g., K-means clustering on the training data (disregarding temporal structure)

2) E step: run the forward and the backward recursions to calculate γ and ξ

3) M step: use γ and ξ into the update equations to find a new set of parameters

Keep iterating 2) & 3) until convergence (e.g., change in the likelihood function p(**X**) is below some threshold)

EM-based training: remarks

- In all recursions, the observations enter in the form $p(\boldsymbol{x}_n|\boldsymbol{z}_n)$
- The recursions are therefore independent of
  - The form of the emission PDF
  - The dimensionality of the data
- Optimization: since the observations $\mathbf{x}_n$ are fixed, $p(\boldsymbol{x}_n|\boldsymbol{z}_n)$
  - Can be pre-computed and tabulated before the algorithm starts
  - And kept fixed during the whole EM training procedure
- EM training is most effective when
  - Number of data samples is much larger than the number of parameters
  - Hence, for an effective training long data sequences would be required
  - Multiple short sequences can also be used → this requires a (minor) modification to the update equation and to the computation of the likelihood (see shortly…)

THE PREDICTIVE DISTRIBUTION

Prediction with HMM (1/3)

- Observed data $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$
- The model is already trained (using previous data)
- We would like to predict $\boldsymbol{x}_{N+1}$

HMM predictive distribution (2/3)

$$
\begin{aligned}
p(\boldsymbol{x}_{N+1}|\boldsymbol{X}) &= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}, \boldsymbol{z}_{N+1}|\boldsymbol{X}) = \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) p(\boldsymbol{z}_{N+1}|\boldsymbol{X}) \\
&= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}, \boldsymbol{z}_N|\boldsymbol{X}) \\
&= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N) p(\boldsymbol{z}_N|\boldsymbol{X}) \\
&= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N) \frac{p(\boldsymbol{z}_N, \boldsymbol{X})}{p(\boldsymbol{X})} \\
&= \frac{1}{p(\boldsymbol{X})} \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N) \alpha(\boldsymbol{z}_N)
\end{aligned}
$$

HMM predictive distribution

Marginalization over $\mathbf{z}_{N+1}$ + Bayes' rule, using $p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}, \boldsymbol{X}) = p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1})$

$$p(\boldsymbol{x}_{N+1}|\boldsymbol{X}) = \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}, \boldsymbol{z}_{N+1}|\boldsymbol{X}) = \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) p(\boldsymbol{z}_{N+1}|\boldsymbol{X})$$

$$= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}, \boldsymbol{z}_N|\boldsymbol{X})$$

$$= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N) p(\boldsymbol{z}_N|\boldsymbol{X})$$

$$= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N) \frac{p(\boldsymbol{z}_N, \boldsymbol{X})}{p(\boldsymbol{X})}$$

$$= \frac{1}{p(\boldsymbol{X})} \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N) \alpha(\boldsymbol{z}_N)$$

HMM predictive distribution

Marginalization over $\mathbf{z}_N$

$$
\begin{aligned}
p(\boldsymbol{x}_{N+1}|\boldsymbol{X}) &= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}, \boldsymbol{z}_{N+1}|\boldsymbol{X}) = \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) p(\boldsymbol{z}_{N+1}|\boldsymbol{X}) \\
&= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}, \boldsymbol{z}_N|\boldsymbol{X}) \\
&= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N) p(\boldsymbol{z}_N|\boldsymbol{X}) \\
&= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N) \frac{p(\boldsymbol{z}_N, \boldsymbol{X})}{p(\boldsymbol{X})} \\
&= \frac{1}{p(\boldsymbol{X})} \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N) \alpha(\boldsymbol{z}_N)
\end{aligned}
$$

HMM predictive distribution

Bayes' rule + the fact that $p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N, \boldsymbol{X}) = p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N)$

$$
\begin{aligned}
p(\boldsymbol{x}_{N+1}|\boldsymbol{X}) &= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}, \boldsymbol{z}_{N+1}|\boldsymbol{X}) = \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) p(\boldsymbol{z}_{N+1}|\boldsymbol{X}) \\
&= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}, \boldsymbol{z}_N|\boldsymbol{X}) \\
&= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N) p(\boldsymbol{z}_N|\boldsymbol{X}) \\
&= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N) \frac{p(\boldsymbol{z}_N, \boldsymbol{X})}{p(\boldsymbol{X})} \\
&= \frac{1}{p(\boldsymbol{X})} \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N) \alpha(\boldsymbol{z}_N)
\end{aligned}
$$

HMM predictive distribution

The Bayes' rule, once again…
... plus definition of α

$$
\begin{aligned}
p(\boldsymbol{x}_{N+1}|\boldsymbol{X}) &= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}, \boldsymbol{z}_{N+1}|\boldsymbol{X}) = \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) p(\boldsymbol{z}_{N+1}|\boldsymbol{X}) \\
&= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}, \boldsymbol{z}_N|\boldsymbol{X}) \\
&= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N) p(\boldsymbol{z}_N|\boldsymbol{X}) \\
&= \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N) \frac{p(\boldsymbol{z}_N, \boldsymbol{X})}{p(\boldsymbol{X})} \\
&= \frac{1}{p(\boldsymbol{X})} \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N) \alpha(\boldsymbol{z}_N)
\end{aligned}
$$

HMM predictive distribution (3/3)

$$p(\boldsymbol{x}_{N+1}|\boldsymbol{X}) = \frac{1}{p(\boldsymbol{X})} \sum_{\boldsymbol{z}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{z}_{N+1}) \sum_{\boldsymbol{z}_N} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N)\alpha(\boldsymbol{z}_N)$$

- Compute a forward recursion ($\alpha(\mathbf{z}_N)$) up to $\mathbf{z}_N$
- Compute two additional sums
  - On $\mathbf{z}_N$: transition probabilities
  - On $\mathbf{z}_{N+1}$: emission probabilities

Lesson learned

- Thomas Bayes

  is almighty and …

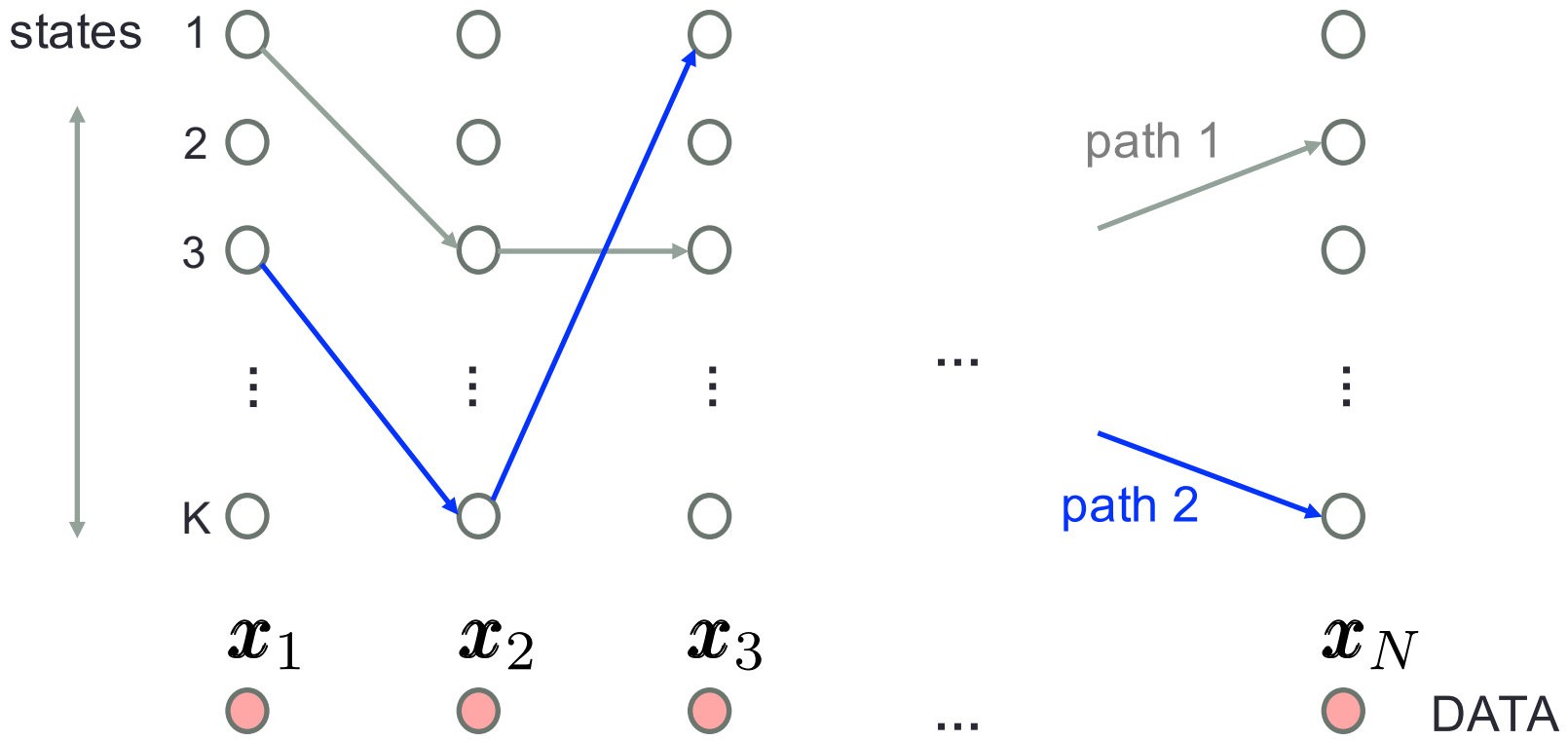
  obiquitous

Reverend Thomas Bayes (1701? – 1761)

THE VITERBI ALGORITHM

It is a “Dynamic Programming” algorithm

Goal



- Observed data is given $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$
- Hidden states $\boldsymbol{Z} = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_N\}$ are unknown
- Find state sequence (path) that maximizes $p(\boldsymbol{Z}|\boldsymbol{X})$
- Problem: number of paths is K^N → exponential complexity

Why?

- In certain applications internal states have a physical meaning
  - It is generally associated with an identification problem of some kind

- Examples:
  - Speech recognition: spoken sequence of phonemes
  - Identity recognition: assess identity from keyboard stroke dynamics
  - Telecom networks: assess application type from generated data bursts

Dynamic Programming

- An efficient procedure is given by the
  - Dynamic Programming theory (sequential decision making)
  - For the problem at stake, it is also known as the “Viterbi algorithm”, see next

- Complete PDF, time step 1:

$$p(\boldsymbol{z}_1)p(\boldsymbol{x}_1|\boldsymbol{z}_1) = p(\boldsymbol{x}_1, \boldsymbol{z}_1)$$

- Complete PDF, time step 2:

$$p(\boldsymbol{z}_1)p(\boldsymbol{x}_1|\boldsymbol{z}_1) \times p(\boldsymbol{z}_2|\boldsymbol{z}_1)p(\boldsymbol{x}_2|\boldsymbol{z}_2) = p(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{z}_1, \boldsymbol{z}_2)$$

- Complete PDF, time step 3:

$$p(\boldsymbol{z}_1)p(\boldsymbol{x}_1|\boldsymbol{z}_1)p(\boldsymbol{z}_2|\boldsymbol{z}_1)p(\boldsymbol{x}_2|\boldsymbol{z}_2) \times p(\boldsymbol{z}_3|\boldsymbol{z}_2)p(\boldsymbol{x}_3|\boldsymbol{z}_3) =$$
$$= p(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{z}_3)$$

Trick – take the logarithm

- Time step 1:

$$\begin{aligned}\omega(\boldsymbol{z}_1) &\stackrel{\Delta}{=} \log\left(p(\boldsymbol{z}_1)p(\boldsymbol{x}_1|\boldsymbol{z}_1)\right) \\ &= \log p(\boldsymbol{z}_1) + \log(p(\boldsymbol{x}_1|\boldsymbol{z}_1)) = \log\left(p(\boldsymbol{x}_1, \boldsymbol{z}_1)\right)\end{aligned}$$

- Time step 2:

$$\begin{aligned}\omega(\boldsymbol{z}_2) &\stackrel{\Delta}{=} \log\left[p(\boldsymbol{z}_1)p(\boldsymbol{x}_1|\boldsymbol{z}_1)p(\boldsymbol{z}_2|\boldsymbol{z}_1)p(\boldsymbol{x}_2|\boldsymbol{z}_2)\right] \\ &= \omega(\boldsymbol{z}_1) + \log p(\boldsymbol{z}_2|\boldsymbol{z}_1) + \log p(\boldsymbol{x}_2|\boldsymbol{z}_2) \\ &= \log p(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{z}_1, \boldsymbol{z}_2)\end{aligned}$$

continuing along the same lines…

- Time step n:

$$\begin{aligned}\omega(\boldsymbol{z}_n) &\stackrel{\Delta}{=} \omega(\boldsymbol{z}_{n-1}) + \log p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) + \log p(\boldsymbol{x}_n|\boldsymbol{z}_n) \\ &= \log p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_n)\end{aligned}$$

Observe the following

- Using this procedure, at step n we obtain the logarithm of the complete PDF (observations **X** and hidden states **Z**)

- Repeating it for N time steps: $\omega(\boldsymbol{z}_N) = \log p(\boldsymbol{X}, \boldsymbol{Z}) = \log p(\boldsymbol{Z}, \boldsymbol{X})$

- Also, using Bayes's rule we have: $p(\boldsymbol{Z}|\boldsymbol{X}) = \dfrac{p(\boldsymbol{X}, \boldsymbol{Z})}{p(\boldsymbol{X})}$

- Moreover:

$$\arg\max_{\boldsymbol{Z}} p(\boldsymbol{Z}|\boldsymbol{X}) = \arg\max_{\boldsymbol{Z}} \frac{p(\boldsymbol{Z}, \boldsymbol{X})}{p(\boldsymbol{X})} =$$

$$\arg\max_{\boldsymbol{Z}} p(\boldsymbol{Z}, \boldsymbol{X}) = \arg\max_{\boldsymbol{Z}} \left(\log p(\boldsymbol{Z}, \boldsymbol{X})\right) =$$

$$= \arg\max_{\boldsymbol{Z}} \omega(\boldsymbol{z}_N)$$

Iterative solution (1/2)

- Start setting:

$$\omega^*(\boldsymbol{z}_1) = \log p(\boldsymbol{z}_1) + \log p(\boldsymbol{x}_1|\boldsymbol{z}_1)$$

- For step n=2,…,N, we have (see previous slides):

$$\omega(\boldsymbol{z}_n) = \log p(\boldsymbol{x}_n|\boldsymbol{z}_n) + \log p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) + \omega(\boldsymbol{z}_{n-1})$$

- To find the maximum, we need to:
  - 1) apply a Dynamic Programming update rule, with **n≥2**:

$$\omega^*(\boldsymbol{z}_n) = \log p(\boldsymbol{x}_n|\boldsymbol{z}_n) + \max_{\boldsymbol{z}_{n-1}} \left[\log p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) + \omega^*(\boldsymbol{z}_{n-1})\right]$$

only quantities that depend on $\mathbf{z}_{n-1}$

  - 2) at each step n: keep track of the optimizing $\mathbf{z}_{n-1}$ (see "argmax")

Iterative solution (2/2)

- Final optimal update rule is a Dynamic Programming (DP) equation:

$$\omega^*(\boldsymbol{z}_n) = \log p(\boldsymbol{x}_n|\boldsymbol{z}_n) + \max_{\boldsymbol{z}_{n-1}} [\log p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) + \omega^*(\boldsymbol{z}_{n-1})]$$

- For each time step
  - Search for the best $\mathbf{z}_{n-1}$ → across K states
  - Compute DP rule for each state $\mathbf{z}_n$ → update for K states

- N time steps →
  - Complexity is $O(K^2 N)$
  - As opposed to $O(K^N)$ for an exhaustive search over all paths
  - Dramatic saving in computation time!

Practical issues to look at

- Re-estimation to train HMM with multiple short sequences
- Normalization of probabilities to avoid overflow
  - Technicality, very important for implementations
- Initialization of model parameters

- Lawrence R. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, Proceedings of the IEEE, Vol. 77, No. 2, 1989.

- Implementation: Hidden Markov Model Portable Toolkit (HTK) for building and manipulating Hidden Markov Models, see: http://htk.eng.cam.ac.uk/

References

- Chapter 13 - “Sequential Data” of [1]

- [1] Cristopher M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

- [2] Lawrence R. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, Proceedings of the IEEE, Vol. 77, No. 2, 1989.

HIDDEN MARKOV MODELS (HMM) FOR MODELING DATA SEQUENCES

Michele Rossi
rossi@dei.unipd.it

Dept. of Information Engineering
University of Padova, IT