香港科技大學
THE HONG KONG
UNIVERSITY OF SCIENCE
AND TECHNOLOGY

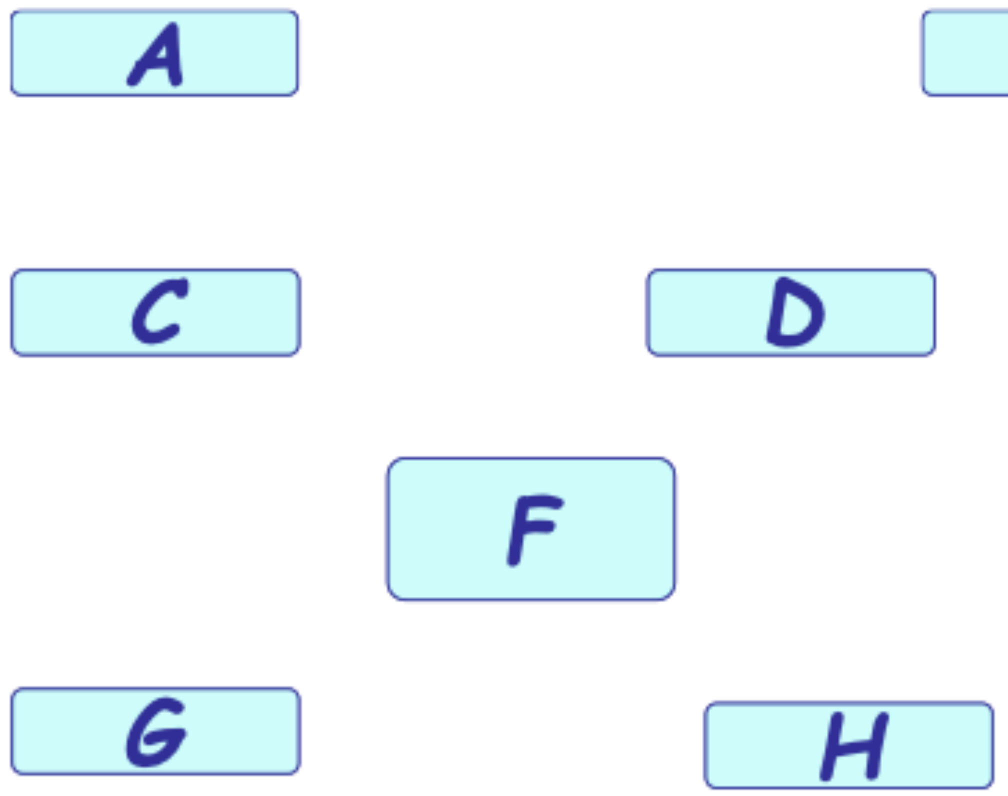
COMP 5212
Machine Learning
Lecture 16

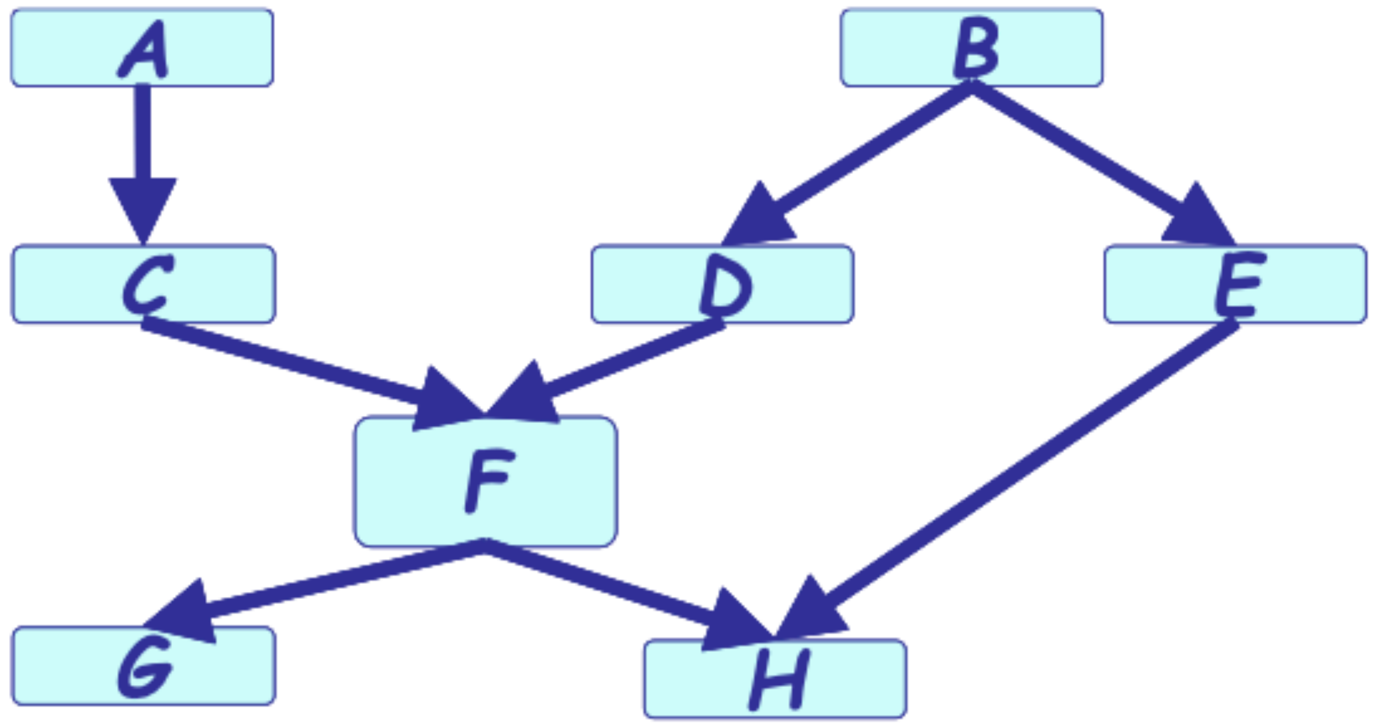
PGM, Hidden Markov Models

Junxian He
Mar 26, 2024

Recap: Probabilistic Graphical Models

It is a smart way to write/specify/compose/design exponentially-large probability distributions without paying an exponential cost, and at the same time endow the distributions with ***structured semantics***





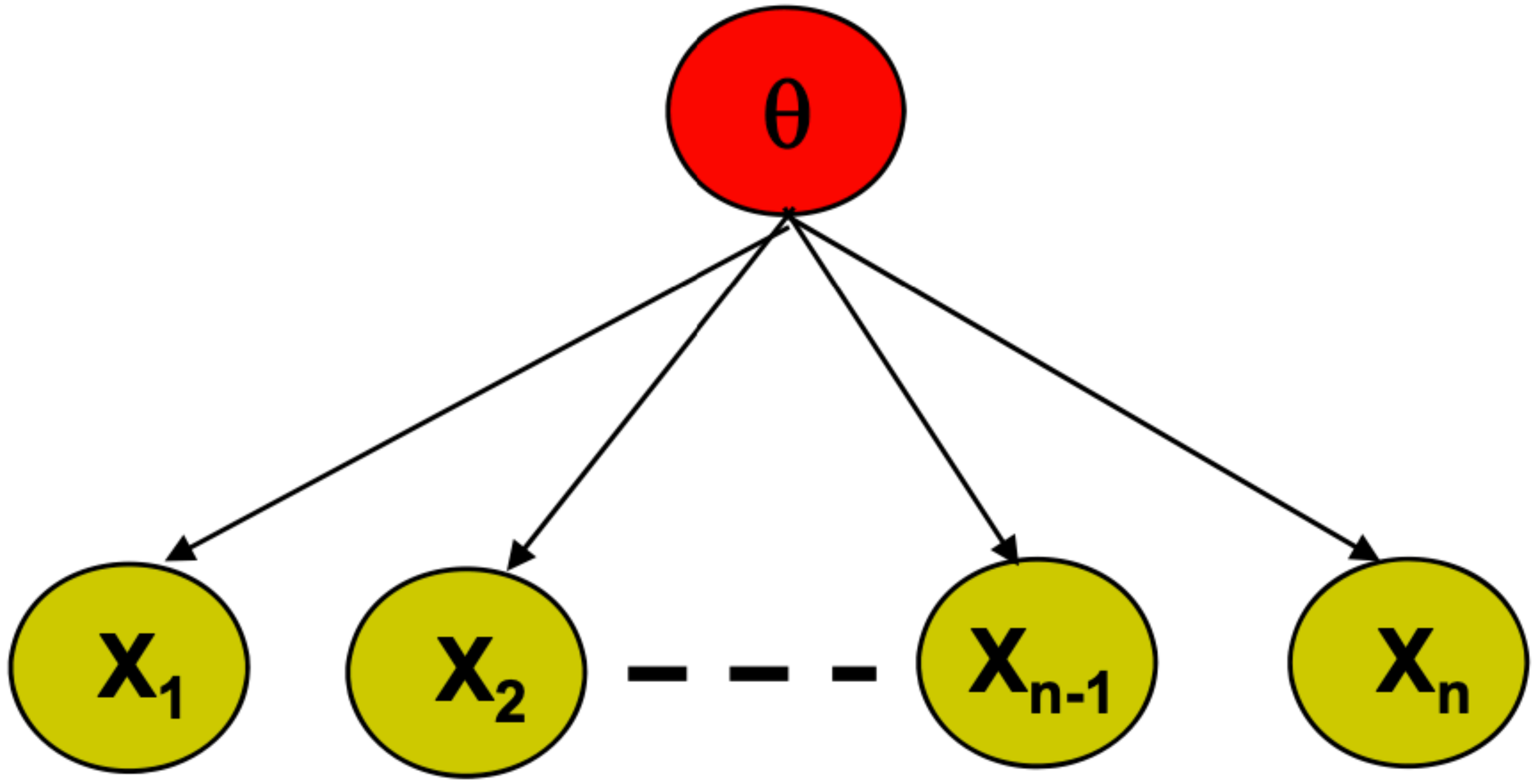
$$P(X_1,X_2,X_3,X_4,X_5,X_6,X_7,X_8)$$

$$P(X_{1:8}) = P(X_1)P(X_2)P(X_3 \mid X_1X_2)P(X_4 \mid X_2)P(X_5 \mid X_2)$$
$$P(X_6 \mid X_3, X_4)P(X_7 \mid X_6)P(X_8 \mid X_5, X_6)$$

More formal definition: It refers to a family of distributions on a set of random variables that are compatible with all the probabilistic independence propositions encoded by a graph that connects these variables

Probabilistic Graphical Model is a graphical language to express conditional independence

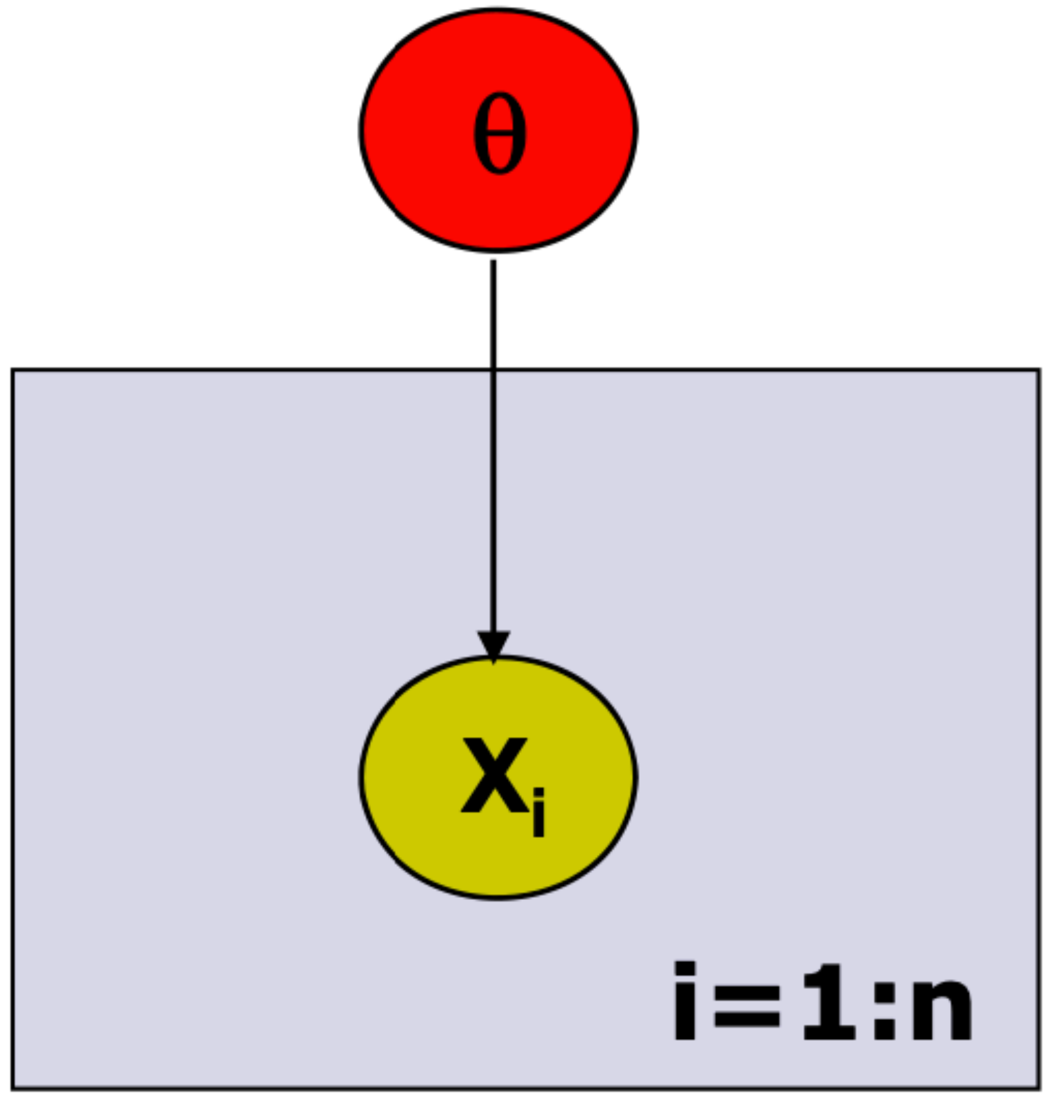
Conditionally Independent Observations



Model parameters

Data {X1, X2 …. Xn}

"Plate" Notation



Model parameters

Data = $\{x_1, \ldots x_n\}$

variables within a plate are replicated
in a conditionally independent manner

Example: Gaussian Model

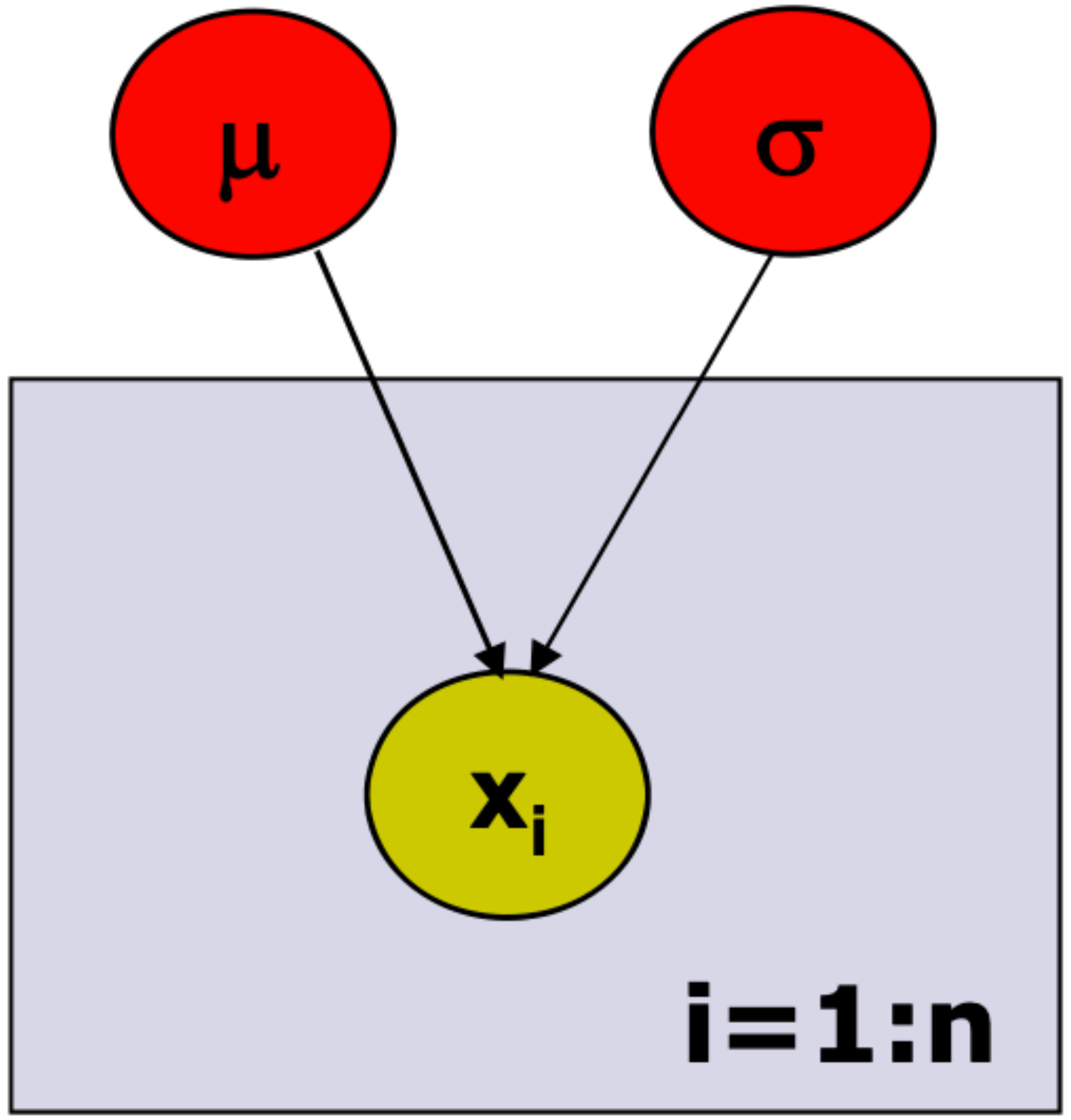
Generative model:

$p(x_1, \ldots x_n \mid \mu, \sigma) \quad = P \ p(x_i \mid \mu, \sigma)$

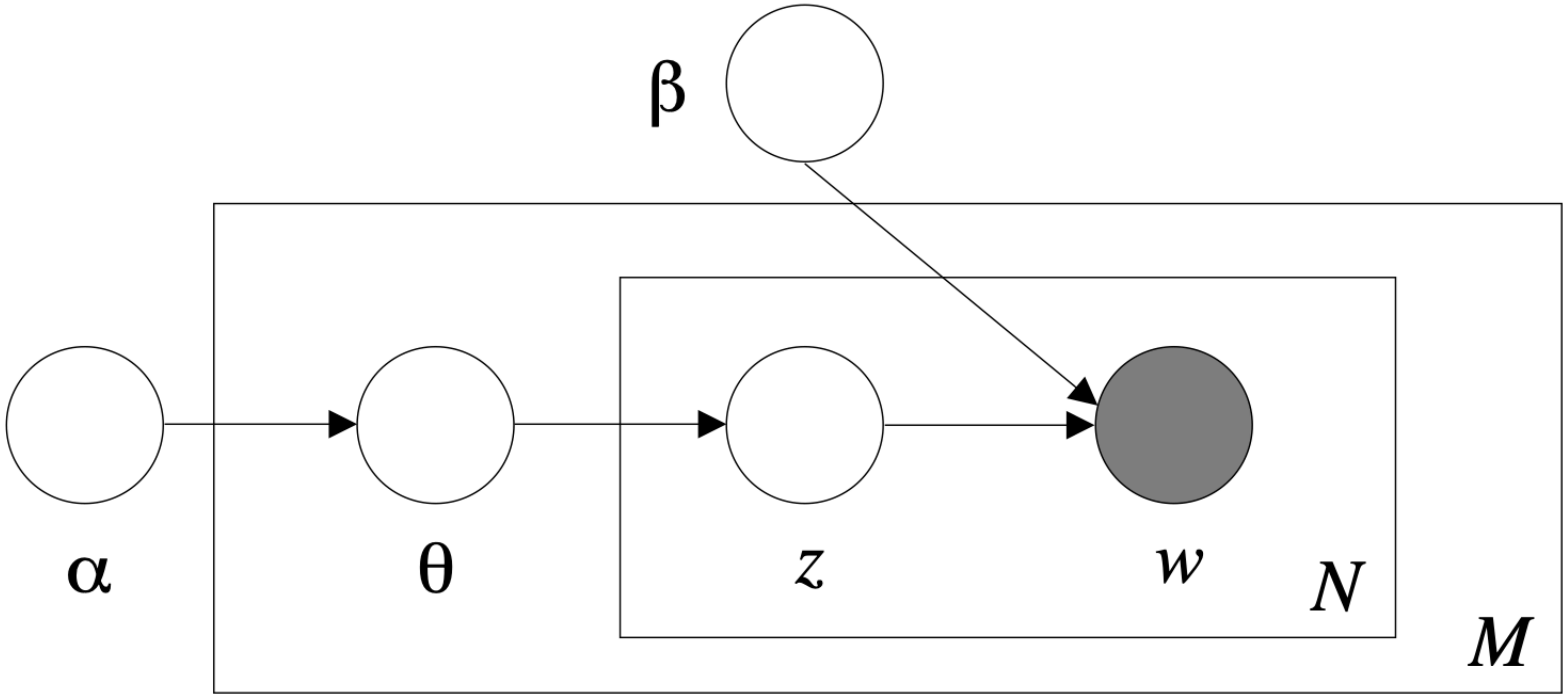
$= \ p(\text{data} \mid \text{parameters})$

$= \ p(D \mid \theta)$

where $\theta = \{\mu, \sigma\}$



Observed Variable and Latent Variable Notations



We typically use gray variables to denote observed variables

Gaussian Mixture Model / Gaussian Discriminative Analysis in PGMs

Inference and Learning

Query a node (random variable) in the graph

- Task 1: How do we answer **queries** about P?
  - We use **inference** as a name for the process of computing answers to such queries

- Task 2: How do we estimate a **plausible model** M from data D?

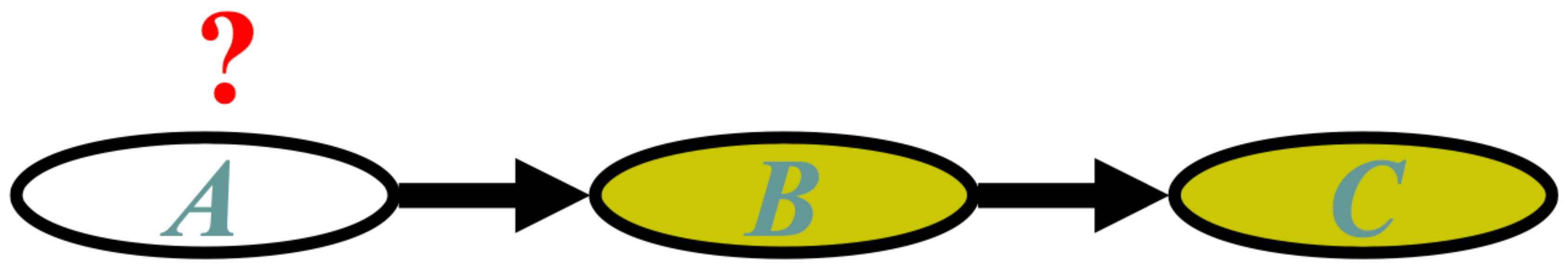
  i. We use **learning** as a name for the process of obtaining point estimate of M.

Examples

- **Prediction**: what is the probability of an outcome given the starting condition

  - the query node is a descendent of the evidence

- **Diagnosis**: what is the probability of disease/fault given symptoms



  - the query node an ancestor of the evidence

In practice, the observed variable is often the data that is on the leaf nodes

How to Learn the Parameters

1. When θ is the parameter and does not have prior —> MLE

$$p(x, z; \theta)$$

2. When we add the prior over θ —> MAP (Bayesian)

$$p(x, z, \theta)$$

How to do MLE on Latent Variable Models?

Expectation Maximization!

The E-step computes the posterior distribution $p(z|x)$

This process is referred to as inference

Approaches to Inference

- Exact inference algorithms
  - The elimination algorithm
  - Belief propagation
  - The junction tree algorithms (but will not cover in detail here)

- Approximate inference techniques
  - Variational algorithms
  - Stochastic simulation / sampling methods
  - Markov chain Monte Carlo methods

Variational Autoencoders

Elimination Algorithm/ Marginalization

$$P(h) = \sum_{g}\sum_{f}\sum_{e}\sum_{d}\sum_{c}\sum_{b}\sum_{a} P(a,b,c,d,e,f,g,h)$$

a naïve summation needs to enumerate over an exponential number of terms

$$X1 \rightarrow X2 \rightarrow X3 \rightarrow X4$$

What if the random variables follow this chain structure?

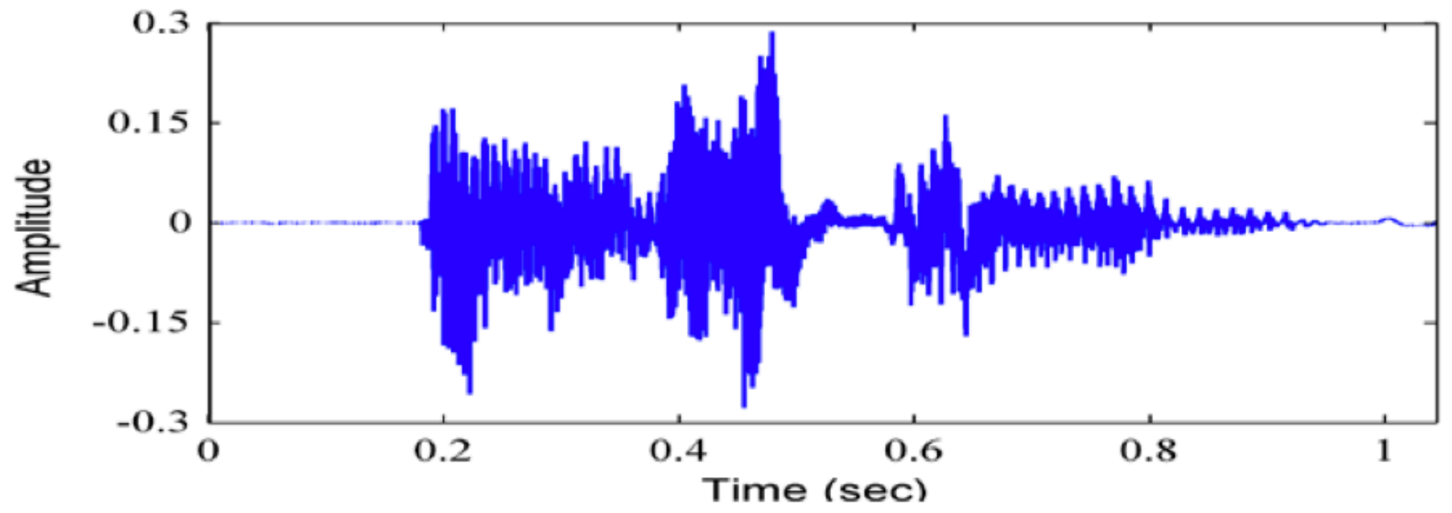
Hidden Markov Models

i.i.d to sequential data

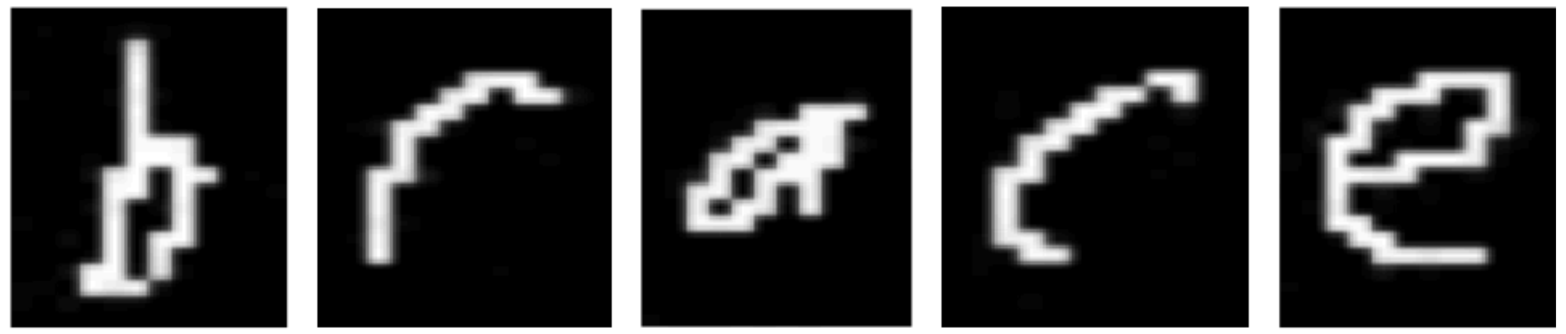
- So far we assumed independent, identically distributed data

$$\{X_i\}_{i=1}^n \overset{iid}{\sim} p(\mathbf{X})$$

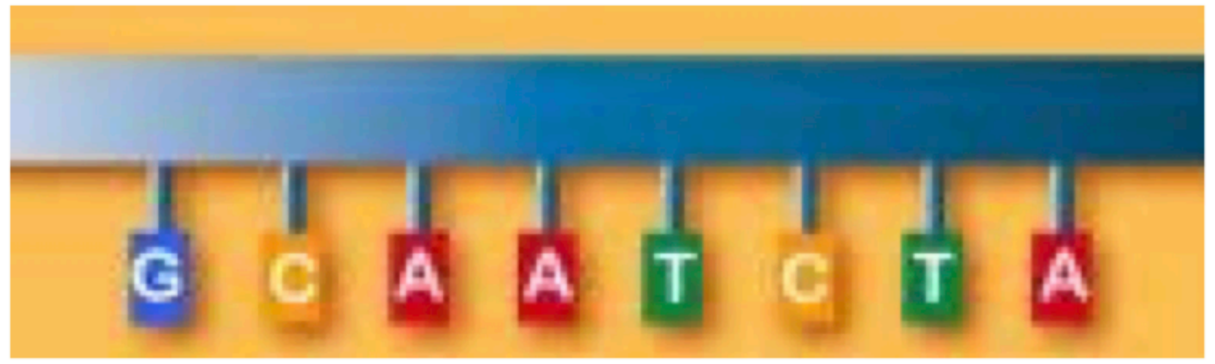
- Sequential (non i.i.d.) data
  - Time-series data
    E.g. Speech



  - Characters in a sentence



  - Base pairs along a DNA strand



(Sequential data is still i.i.d on the sequence level)

Markov Models

- Joint distribution of *n* arbitrary random variables

$$\begin{aligned} p(\mathbf{X}) &= p(X_1, X_2, \ldots, X_n) \\ &= p(X_1)p(X_2|X_1)p(X_3|X_2, X_1)\ldots p(X_n|X_{n-1}, ..., X_1) \\ &= \prod_{i=1}^{n} p(X_n|X_{n-1}, \ldots, X_1) \end{aligned}$$

Chain rule

- Markov Assumption (m^{th} order)

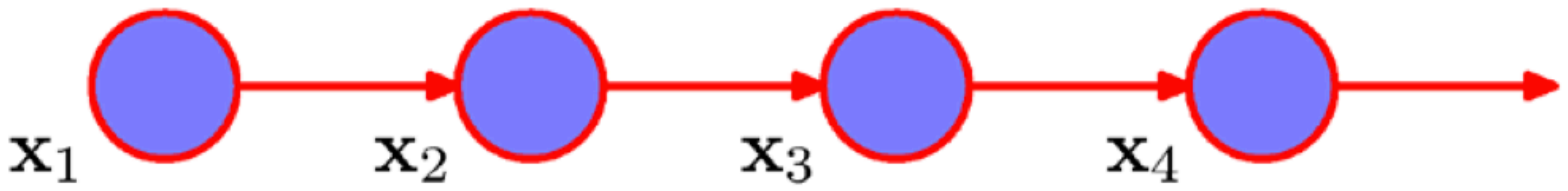
$$p(\mathbf{X}) = \prod_{i=1}^{n} p(X_n|X_{n-1}, \ldots, X_{n-m})$$

Current observation only depends on past m observations

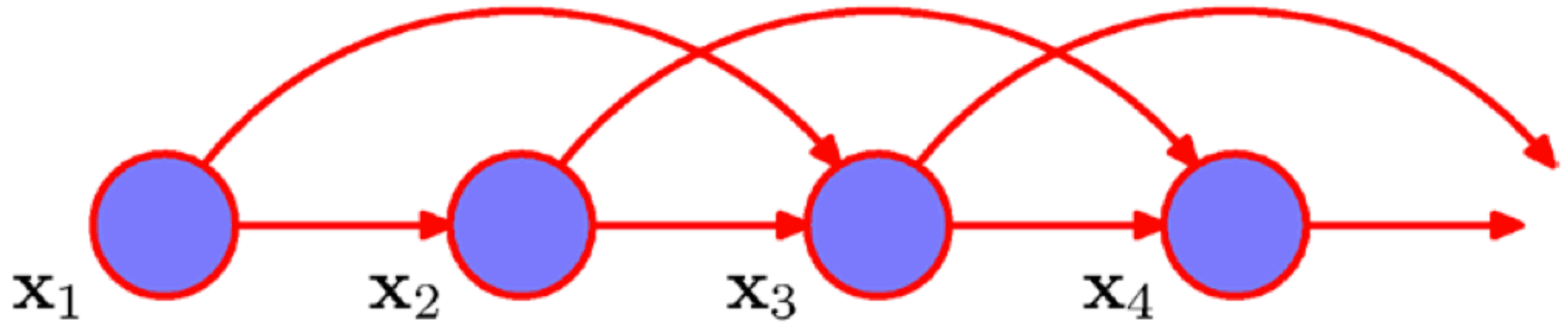
Markov Models

- Markov Assumption

1st order
$$p(\mathbf{X}) = \prod_{i=1}^{n} p(X_n|X_{n-1})$$



2nd order
$$p(\mathbf{X}) = \prod_{i=1}^{n} p(X_n|X_{n-1}, X_{n-2})$$



Markov Models

Homogeneous/stationary Markov model (probabilities don't depend on n)

parameters in stationary model K-ary variables

❑ Markov Assumption

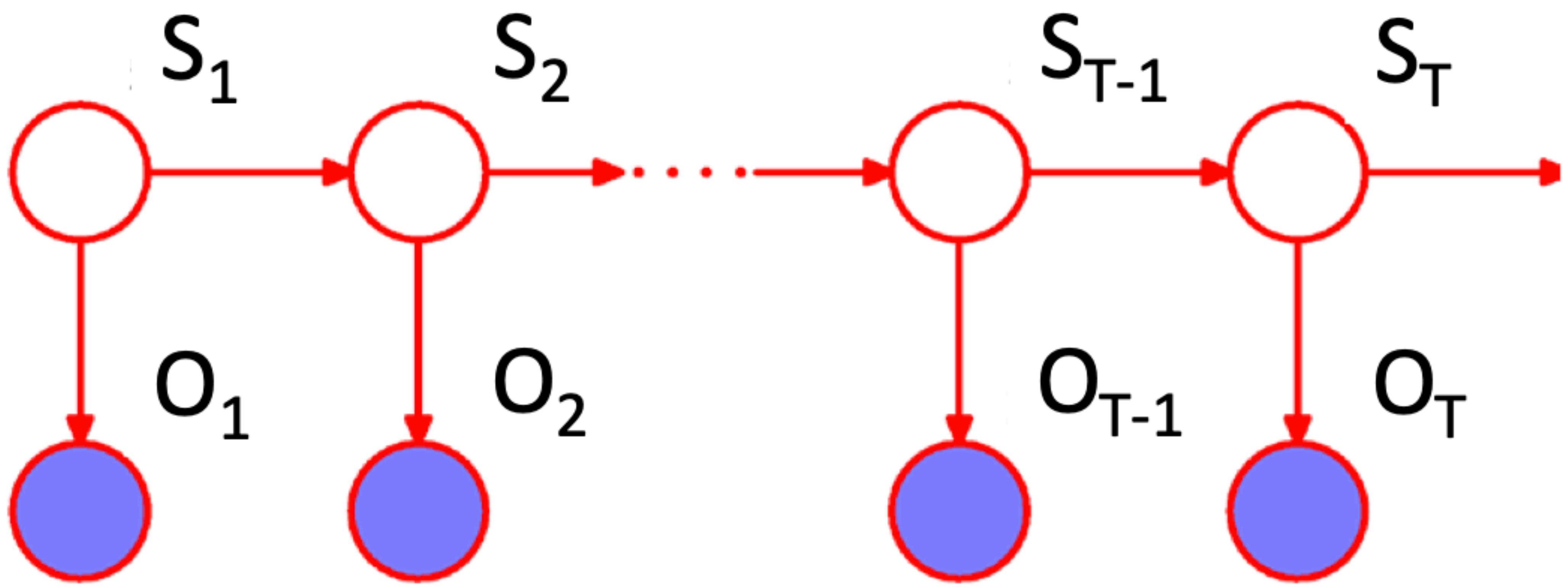
1st order $p(\mathbf{X}) = \prod_{i=1}^{n} p(X_n | X_{n-1})$ $O(K^2)$

mth order $p(\mathbf{X}) = \prod_{i=1}^{n} p(X_n | X_{n-1}, \ldots, X_{n-m})$ $O(K^{m+1})$

n-1th order $p(\mathbf{X}) = \prod_{i=1}^{n} p(X_n | X_{n-1}, \ldots, X_1)$ $O(K^n)$

≡ no assumptions – complete (but directed) graph

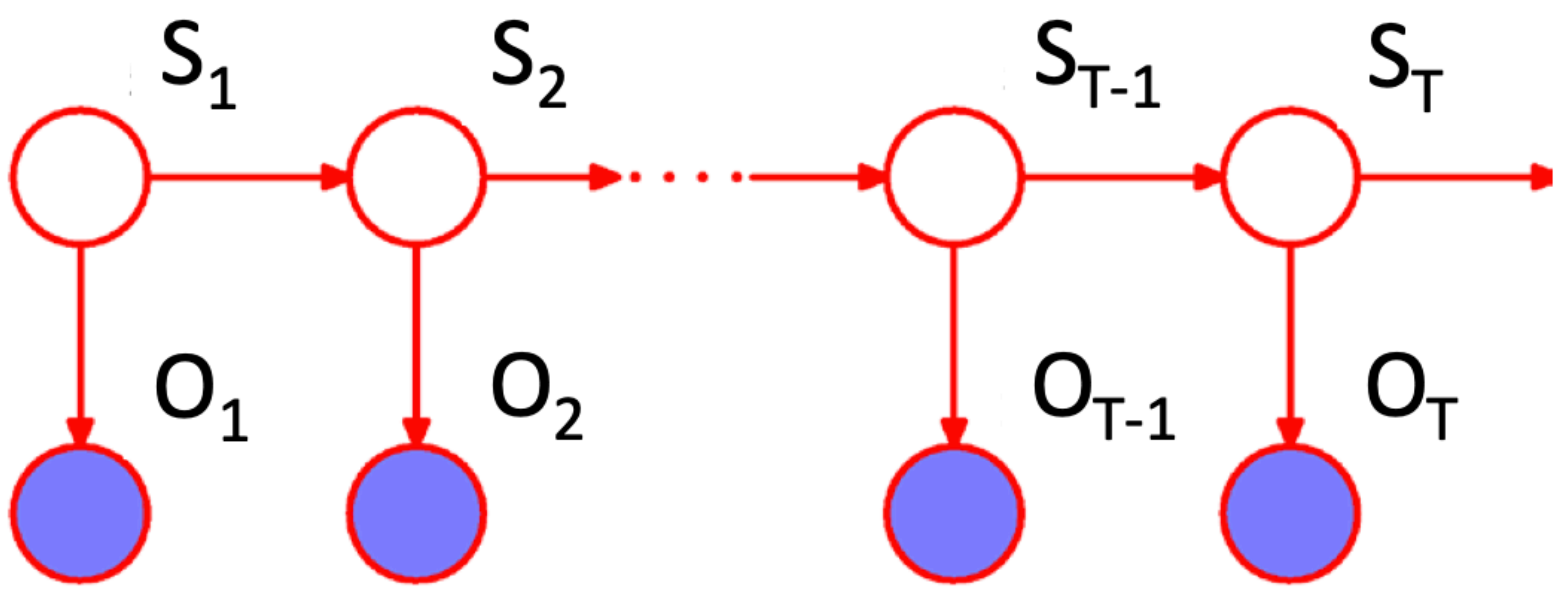
Hidden Markov Models



Observation space $O_t \in \{y_1, y_2, \ldots, y_K\}$

Hidden states $S_t \in \{1, \ldots, I\}$

Hidden Markov Models



$$p(S_1, \ldots, S_T, O_1, \ldots, O_T) = \prod_{t=1}^{T} p(O_t|S_t) \prod_{t=1}^{T} p(S_t|S_{t-1})$$

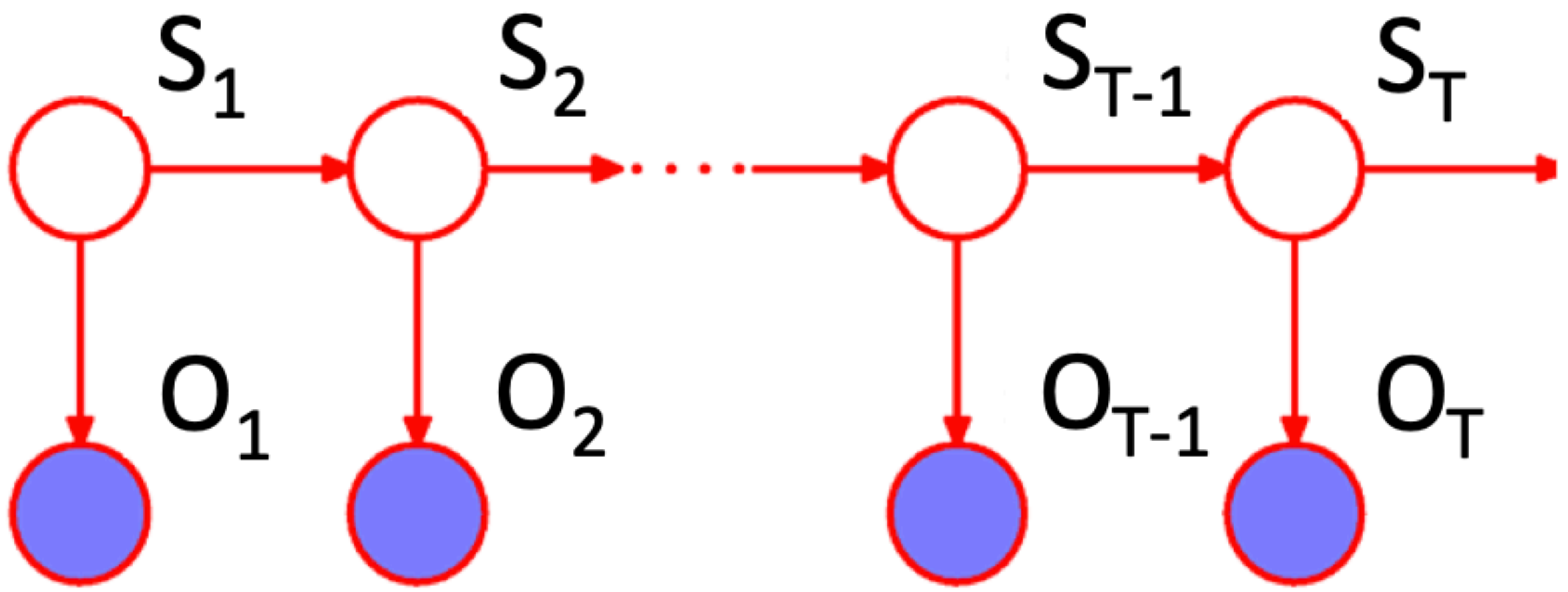
1st order Markov assumption on hidden states $\{S_t\}$ t = 1, …, T

(can be extended to higher order).

Is O_T and O_2 independent?

Hidden Markov Models

- Parameters – stationary/homogeneous markov model (independent of time t)



Initial probabilities

$$p(S_1 = i) = \pi_i$$

Transition probabilities

$$p(S_t = j | S_{t-1} = i) = p_{ij}$$

Emission probabilities

$$p(O_t = y | S_t = i) = q_i^y$$

$$p(\{S_t\}_{t=1}^T, \{O_t\}_{t=1}^T) \quad = \quad p(S_1) \prod_{t=2}^{T} p(S_t|S_{t-1}) \prod_{t=1}^{T} p(O_t|S_t)$$

HMM Example

- The Dishonest Casino

A casino has two dices:

Fair dice

P(1) = P(2) = P(3) = P(5) = P(6) = 1/6

Loaded dice

P(1) = P(2) = P(3) = P(5) = 1/10
P(6) = ½

Casino player switches back-&-forth between fair and loaded die with 5% probability

HMM Example

GIVEN: A sequence of rolls by the casino player

1245526462146146136136661664661636616366163616515615115146123562344

Question

1. How likely is the sequence given our model?

   This is the evaluation problem in HMMs

2. What portion of the sequence was generated with the fair die, and what portion with the loaded die

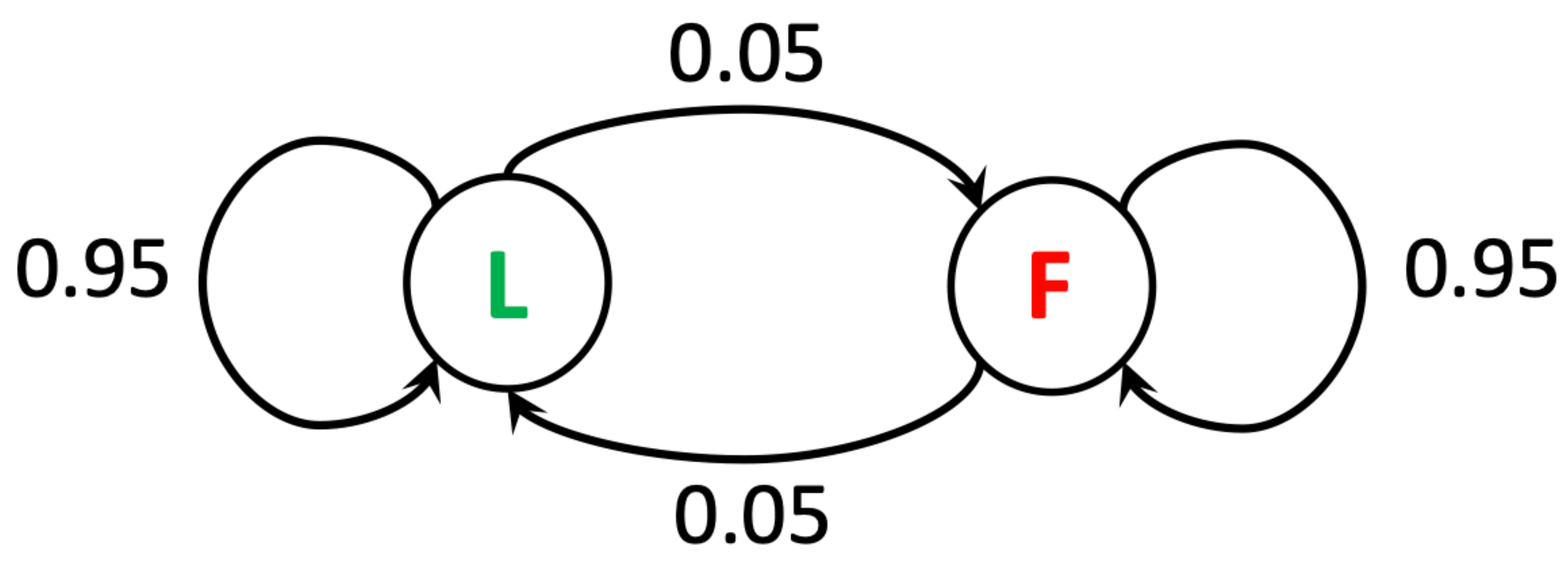
   This is the decoding question in HMMs

3. How “loaded” is the loaded die? How “fair” is the fair die? How often does the casino player change from fair to loaded, and back?

   This is the learning question in HMMs

State Space Representation

- Switch between F and L with 5% probability



- **HMM Parameters**

Initial probs $P(S_1 = L) = 0.5 = P(S_1 = F)$

Transition probs $P(S_t = L/F | S_{t-1} = L/F) = 0.95$

$P(S_t = F/L | S_{t-1} = L/F) = 0.05$

Emission probabilities $P(O_t = y | S_t = F) = 1/6 \qquad y = 1,2,3,4,5,6$

$P(O_t = y | S_t = L) = 1/10 \qquad y = 1,2,3,4,5$

$= 1/2 \qquad y = 6$

Three Main Problems in HMMs

- Evaluation – Given HMM parameters & observation seqn $\{O_t\}_{t=1}^T$

  find $p(\{O_t\}_{t=1}^T \,|\theta)$ prob of observed sequence

- Decoding – Given HMM parameters & observation seqn $\{O_t\}_{t=1}^T$

  find $\arg \max_{s_1,\ldots,s_T} p(\{S_t\}_{t=1}^T | \{O_t\}_{t=1}^T, \theta)$ most probable

  sequence of hidden states

- Learning – Given HMM with unknown parameters and $\{O_t\}_{t=1}^T$ observation sequence

  find $\arg\max_\theta p(\{O_t\}_{t=1}^T | \theta)$ parameters that maximize

  likelihood of observed data

HMM Algorithms

- Evaluation – What is the probability of the observed sequence? Forward Algorithm

- Decoding – What is the probability that the third roll was loaded given the observed sequence? Forward-Backward Algorithm

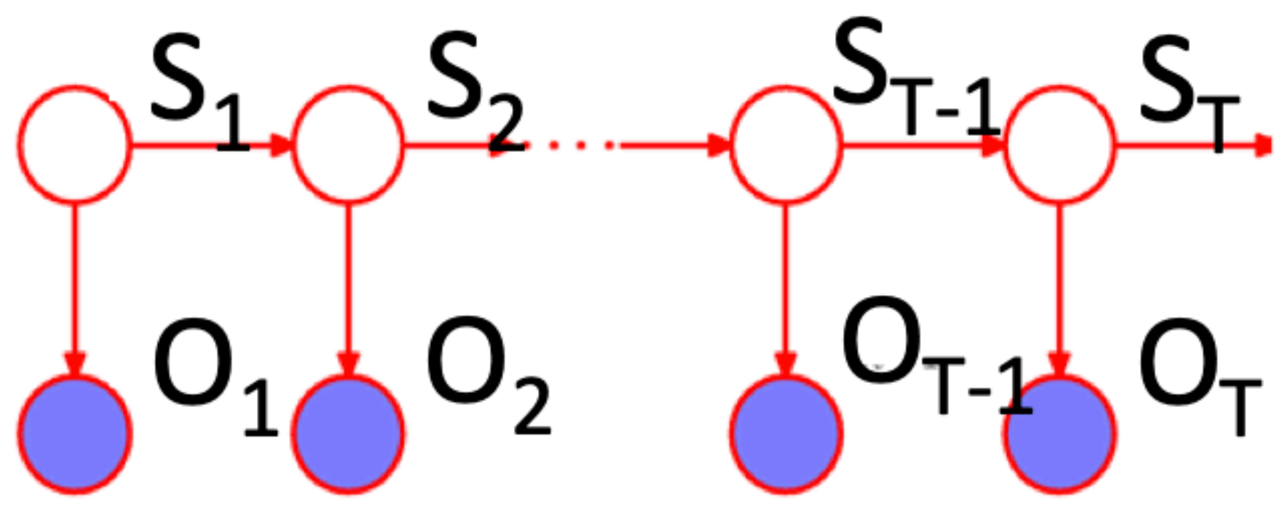
  – What is the most likely die sequence given the observed sequence? Viterbi Algorithm

- Learning – Under what parameterization is the observed sequence most probable? Baum-Welch Algorithm (EM)

Evaluation Problem

- Given HMM parameters $p(S_1), p(S_t|S_{t-1}), p(O_t|S_t)$ & observation sequence $\{O_t\}_{t=1}^T$

find probability of observed sequence



$$p(\{O_t\}_{t=1}^T) = \sum_{S_1,\dots,S_T} p(\{O_t\}_{t=1}^T, \{S_t\}_{t=1}^T)$$

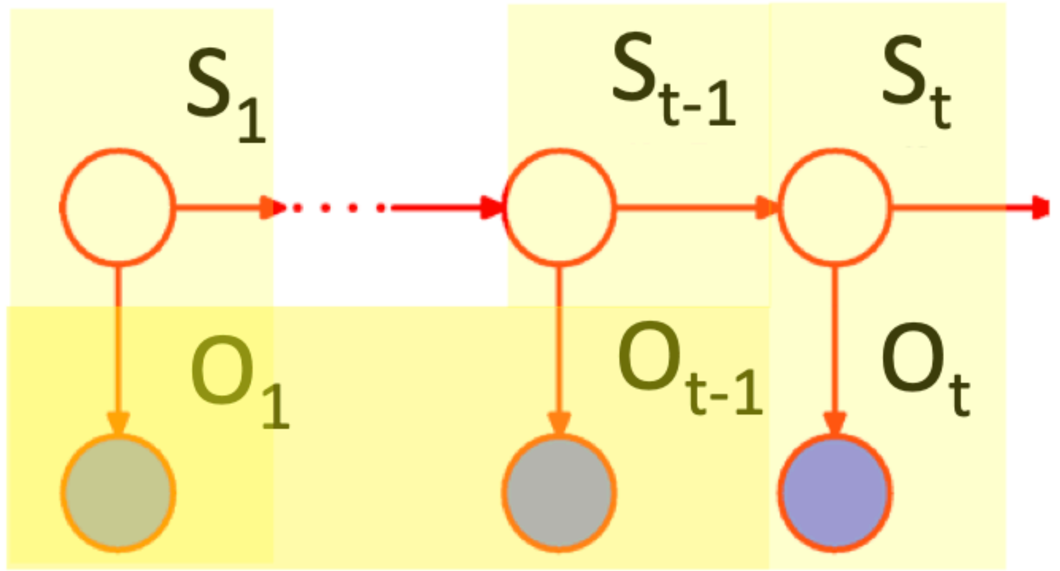
$$= \sum_{S_1,\dots,S_T} p(S_1) \prod_{t=2}^T p(S_t|S_{t-1}) \prod_{t=1}^T p(O_t|S_t)$$

requires summing over all possible hidden state values at all times – K^T exponential # terms!

Forward Probability

$$p(\{O_t\}_{t=1}^T) = \sum_k p(\{O_t\}_{t=1}^T, S_T = k) = \sum_k \alpha_T^k$$

Compute forward probability α_t^k recursively over t



$$\alpha_t^k \quad := \quad p(O_1, \ldots, O_t, S_t = k)$$

Introduce S_{t-1}

Chain rule

Markov assumption

$$= \quad p(O_t | S_t = k) \sum_i \alpha_{t-1}^i p(S_t = k | S_{t-1} = i)$$

Forward Algorithm

Can compute α_t^k for all k, t using dynamic programming:

- Initialize: $\alpha_1^k = p(O_1 | S_1 = k)\, p(S_1 = k)$ for all k

- Iterate: for t = 2, …, T

$$\alpha_t^k = p(O_t | S_t = k) \sum_i \alpha_{t-1}^i\, p(S_t = k | S_{t-1} = i) \qquad \text{for all } k$$

- Termination: $p(\{O_t\}_{t=1}^T) = \sum_k \alpha_T^k$

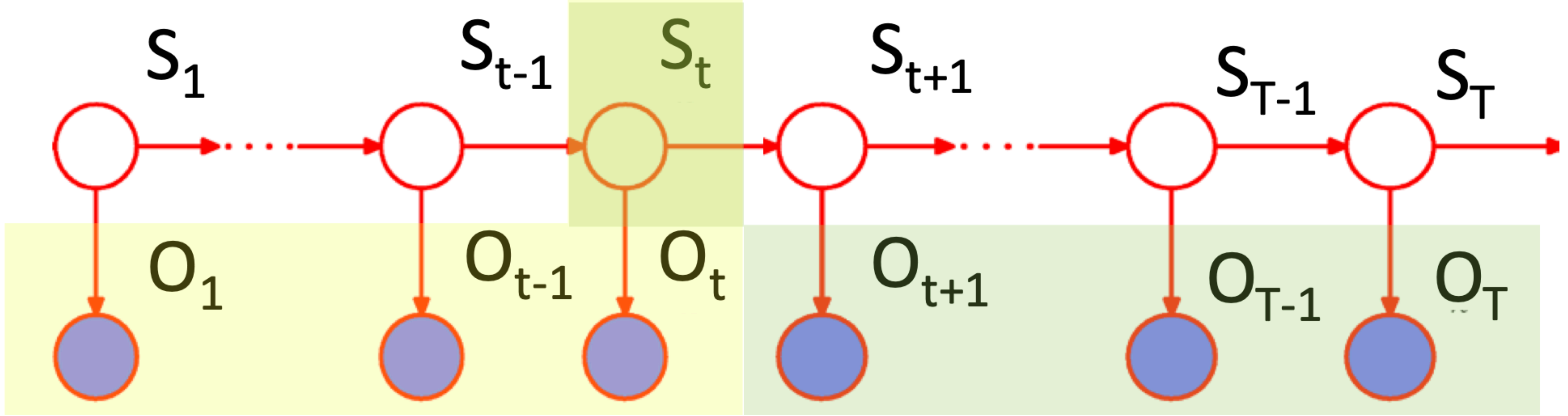
Decoding Problem 1

- Given HMM parameters $p(S_1), p(S_t|S_{t-1}), p(O_t|S_t)$ & observation sequence $\{O_t\}_{t=1}^T$

  find probability that hidden state at time t was k $\;p(S_t = k|\{O_t\}_{t=1}^T)$

$$\begin{aligned} p(S_t = k, \{O_t\}_{t=1}^T) &= p(O_1, \ldots, O_t, S_t = k, O_{t+1}, \ldots, O_T) \\ &= \underbrace{p(O_1, \ldots, O_t, S_t = k)}_{\alpha_t^k}\underbrace{p(O_{t+1}, \ldots, O_T|S_t = k)}_{\beta_t^k} \end{aligned}$$

Compute recursively



Backward Algorithm

Can compute β_t^k for all k, t using dynamic programming:

- Initialize: $\beta_T^k = 1$ for all k

- Iterate: for t = T-1, …, 1

$$\beta_t^k = \sum_i p(S_{t+1} = i | S_t = k) p(O_{t+1} | S_{t+1} = i) \beta_{t+1}^i \quad \text{for all k}$$

- Termination: $p(S_t = k, \{O_t\}_{t=1}^T) = \alpha_t^k \beta_t^k$

$$p(S_t = k | \{O_t\}_{t=1}^T) = \frac{p(S_t = k, \{O_t\}_{t=1}^T)}{p(\{O_t\}_{t=1}^T)} = \frac{\alpha_t^k \beta_t^k}{\sum_i \alpha_t^i \beta_t^i}$$

Most Likely State vs. Most Likely Sequence

- Most likely state assignment at time t

$$\arg\max_k p(S_t = k | \{O_t\}_{t=1}^T) = \arg\max_k \alpha_t^k \beta_t^k$$

E.g. Which die was most likely used by the casino in the third roll given the observed sequence?

- Most likely assignment of state sequence

$$\arg\max_{\{S_t\}_{t=1}^T} p(\{S_t\}_{t=1}^T | \{O_t\}_{t=1}^T)$$

Are the solutions the same?

Decoding Problem 2

- Given HMM parameters $p(S_1), p(S_t|S_{t-1}), p(O_t|S_t)$ & observation sequence $\{O_t\}_{t=1}^T$

  find most likely assignment of state sequence

$$\arg\max_{\{S_t\}_{t=1}^T} p(\{S_t\}_{t=1}^T | \{O_t\}_{t=1}^T) = \arg\max_{\{S_t\}_{t=1}^T} p(\{S_t\}_{t=1}^T, \{O_t\}_{t=1}^T)$$

$$= \arg\max_k \underbrace{\max_{\{S_t\}_{t=1}^{T-1}} p(S_T = k, \{S_t\}_{t=1}^{T-1}, \{O_t\}_{t=1}^T)}_{V_T^k}$$

Compute recursively

V_T^k - probability of most likely sequence of states ending at state $S_T = k$

Viterbi Decoding

$$\max_{\{S_t\}_{t=1}^T} p(\{S_t\}_{t=1}^T, \{O_t\}_{t=1}^T) = \max_k V_T^k$$

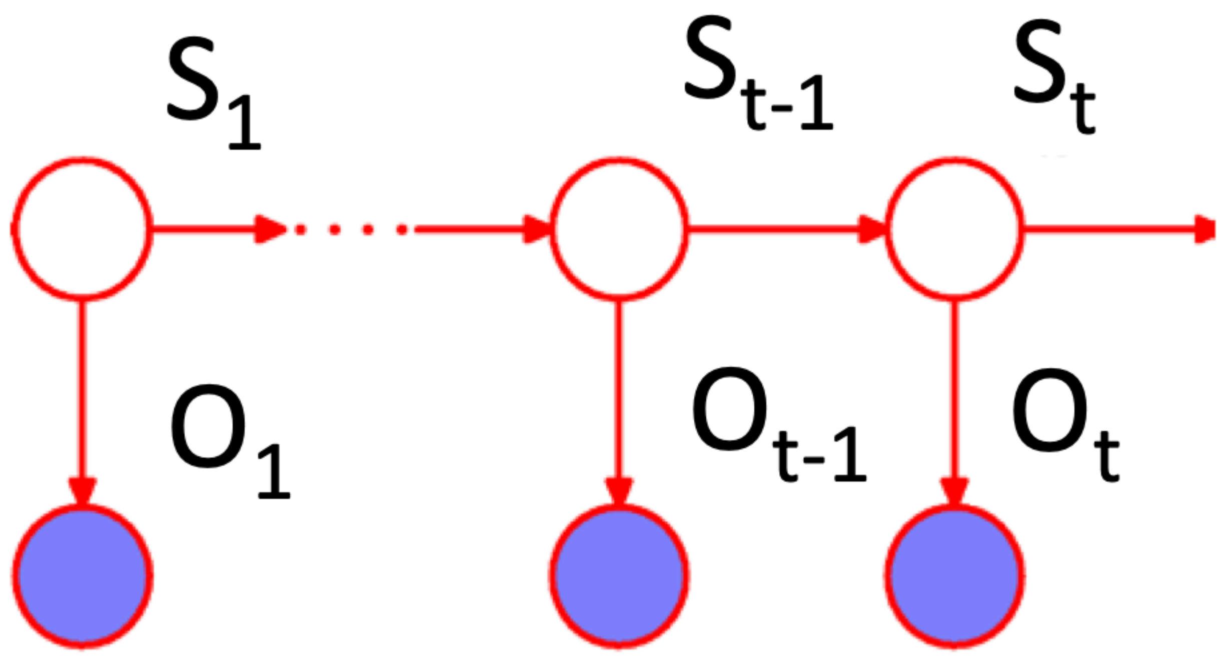
Compute probability V_t^k recursively over t

$$V_t^k \quad := \quad \max_{S_1,\ldots,S_{t-1}} p(S_t = k, S_1, \ldots, S_{t-1}, O_1, \ldots, O_t)$$

$$\vdots$$

Bayes rule

Markov assumption



$$= \quad p(O_t|S_t = k) \max_i p(S_t = k|S_{t-1} = i) V_{t-1}^i$$

Viterbi Algorithm

Can compute V_t^k for all k, t using dynamic programming:

- Initialize: $V_1^k = p(O_1|S_1=k)p(S_1 = k)$ for all k

- Iterate: for t = 2, …, T

$$V_t^k = p(O_t|S_t = k) \max_i p(S_t = k|S_{t-1} = i)V_{t-1}^i$$ for all k

- Termination: $$\max_{\{S_t\}_{t=1}^T} p(\{S_t\}_{t=1}^T, \{O_t\}_{t=1}^T) = \max_k V_T^k$$

Traceback:

$$S_T^* = \arg\max_k V_T^k$$

$$S_{t-1}^* = \arg\max_i p(S_t^*|S_{t-1} = i)V_{t-1}^i$$

Can we do in the backward direction?

Computational Complexity

- What is the running time for Forward, Backward, Viterbi?

$$\alpha_t^k = q_k^{O_t} \sum_i \alpha_{t-1}^i \, p_{i,k}$$

$$\beta_t^k = \sum_i p_{k,i} \, q_i^{O_{t+1}} \, \beta_{t+1}^i$$

$$V_t^k = q_k^{O_t} \, \max_i p_{i,k} \, V_{t-1}^i$$

$O(K^2T)$ linear in T instead of $O(K^T)$ exponential in T!

Learning with EM

- Start with random initialization of parameters
- **E-step** – Fix parameters, find expected state assignments

$$\gamma_i(t) = p(S_t = i|O,\theta) = \frac{\alpha_t^i \beta_t^i}{\sum_j \alpha_t^j \beta_t^j} \qquad \mathbf{O} = \{O_t\}_{t=1}^T$$

Forward-Backward algorithm

$$\xi_{ij}(t) = p(S_{t-1} = i, S_t = j|O,\theta)$$

$$= \frac{p(S_{t-1} = i|O,\theta)p(S_t = j, O_t, \ldots, O_T|S_{t-1} = i, \theta)}{p(O_t, \ldots, O_T|S_{t-1} = i, \theta)}$$

$$= \frac{\gamma_i(t-1)\ p_{ij}\ q_j^{O_t}\ \beta_t^j}{\beta_{t-1}^i}$$

You will derive the EM in your HW

Thank You!
Q & A