

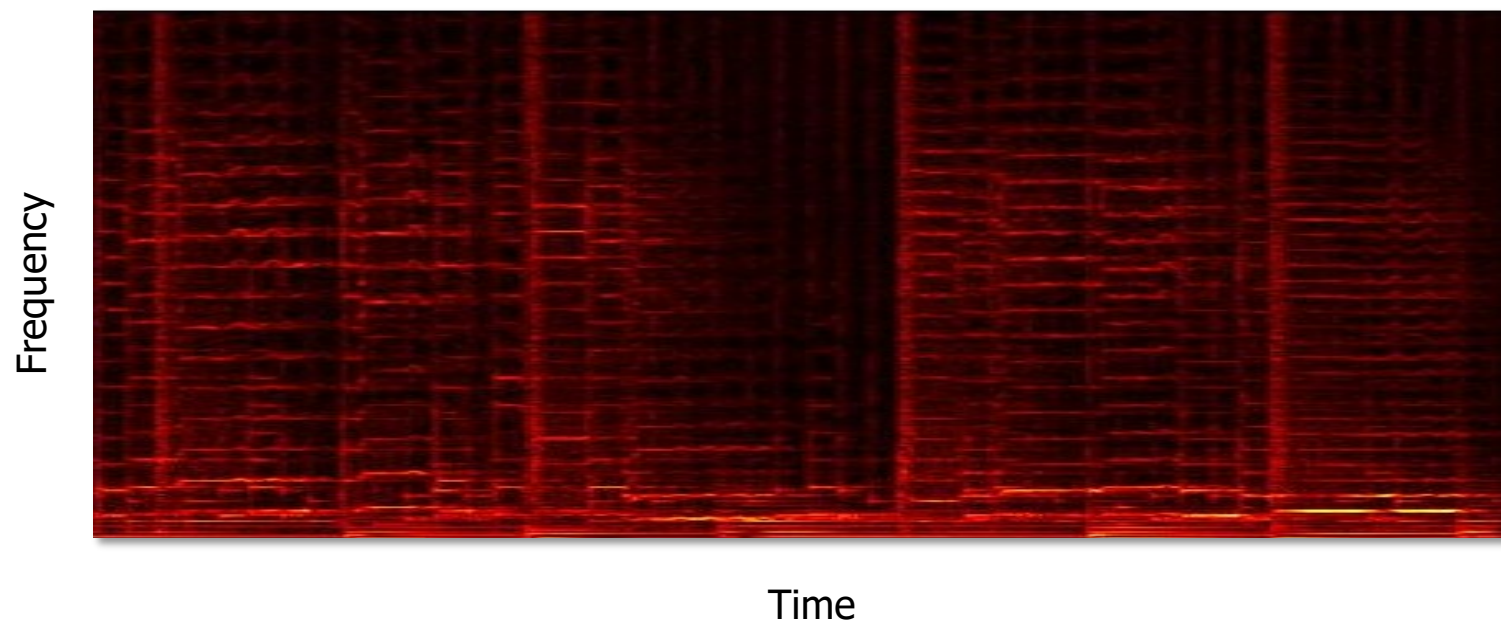
# Topic 8

Audio Modeling by  
Hidden Markov Models

# Structure in Spectrograms

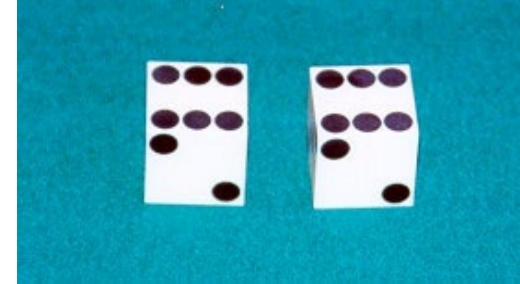
---

- Spectral structure
- Temporal structure



# An HMM Example

---



- A dishonest casino has two dice:

- A fair dice

- $P(1) = P(2) = P(3) = P(4) = P(5) = P(6) = 1/6$

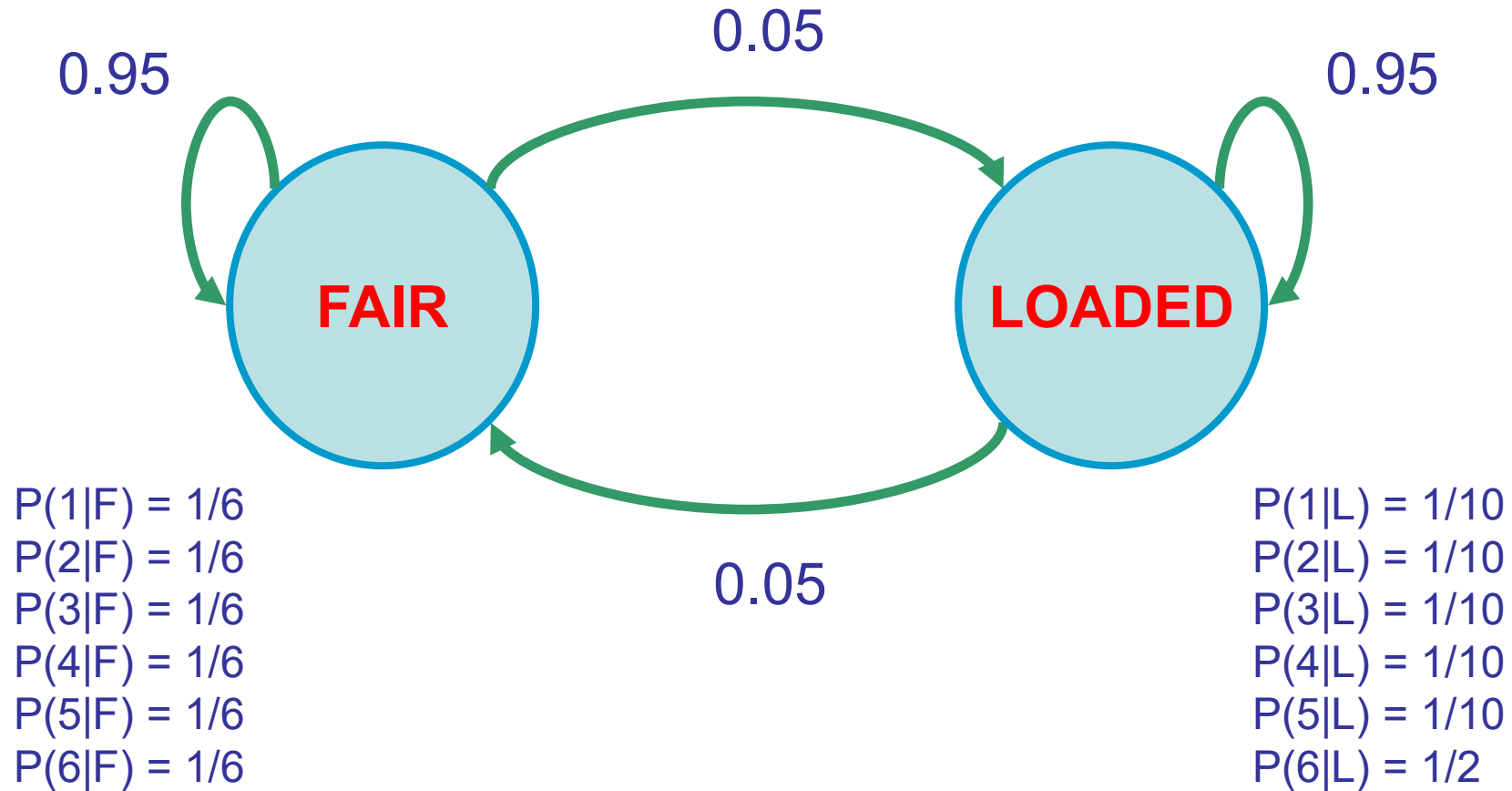
- A loaded dice

- $P(1) = P(2) = P(3) = P(4) = P(5) = 1/10; P(6) = 1/2$

- The casino randomly starts with one dice.
- The casino randomly switches the dice once every 20 turns, on average.

# My Dishonest Casino Model

$P(\text{first dice} = F) = 0.5$ ;  $P(\text{first dice} = L) = 0.5$



# Finite-state HMM

---

- A finite set of states  $\{1, \dots, N\}$
- The **initial probability** of states  $\boldsymbol{\Pi} = \{\pi_1, \dots, \pi_N\}$ 
  - $\pi_i$  is the probability of starting with state  $i$ .
  - $\sum_i \pi_i = 1$
- State **transition probabilities**,  $\boldsymbol{A} = \{a_{ij}\}$ 
  - $a_{ij}$  is the probability of going from state  $i$  to  $j$
  - $\sum_j a_{ij} = 1$
- An emission (observation) alphabet  $\{e_1, \dots, e_M\}$
- **Emission probabilities**,  $\boldsymbol{B} = \{b_{ij}\}$ 
  - $b_{ij}$  is the probability of observing  $e_j$  when at state  $i$
  - $\sum_j b_{ij} = 1$

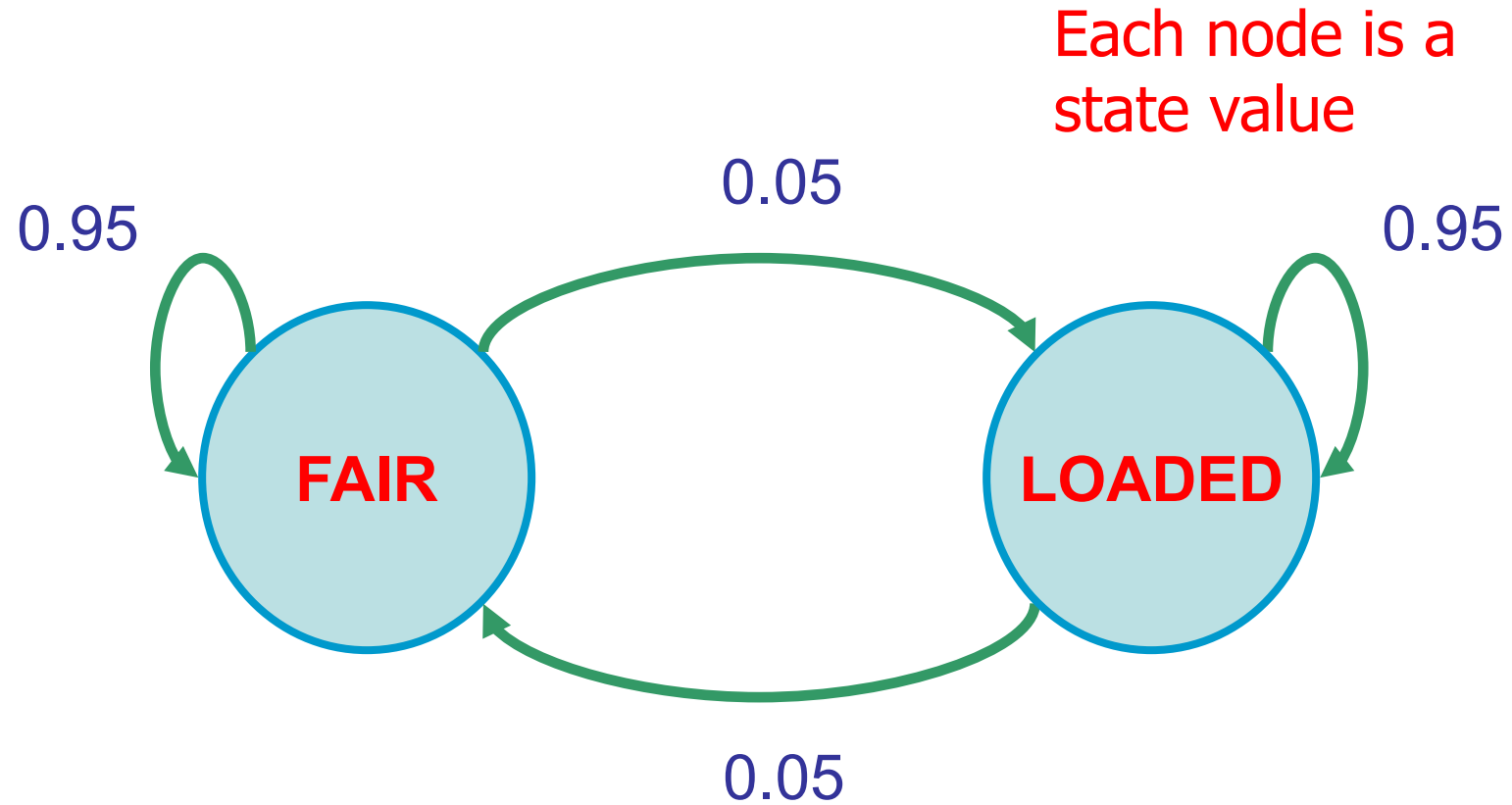
# Markovian Property

---

- If the current state is known, future states do not depend on previous states.
- I.e., what I'm going to do next depends only on where I am now, NOT on how I got here.
- Memory-less

# State Space Representation

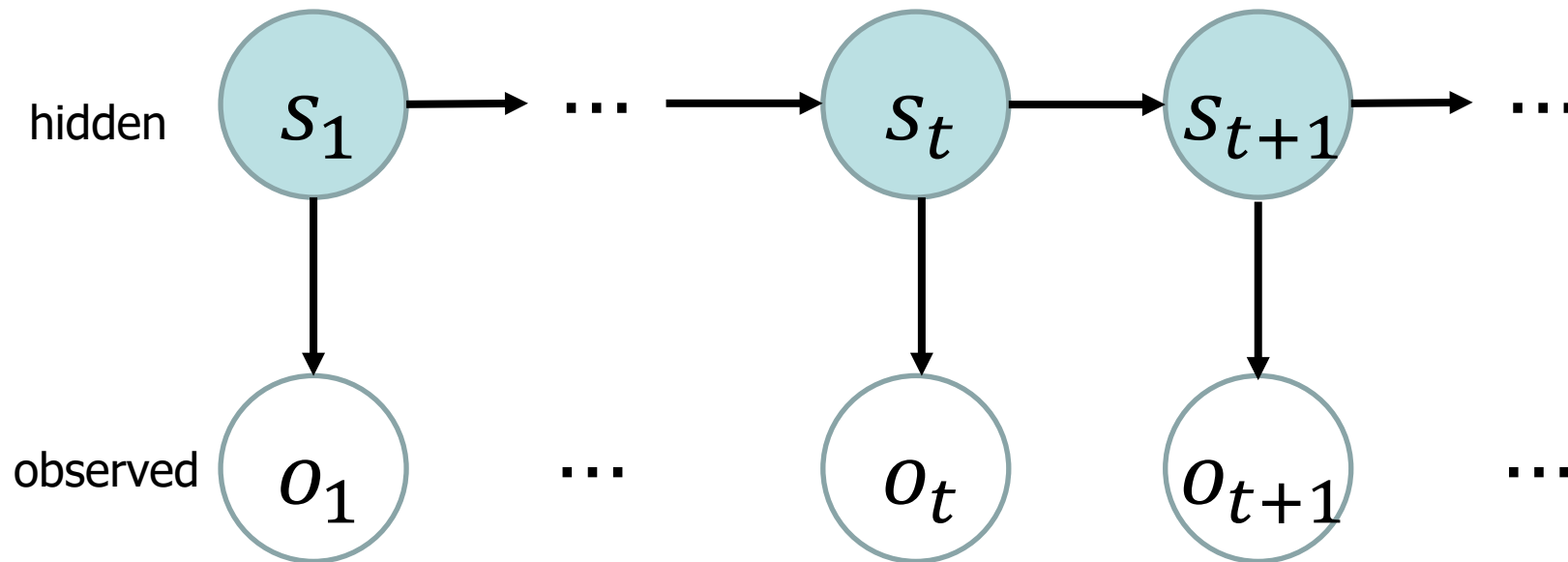
---



# Probabilistic Graphical Model Representation

---

- Let  $s_t$  be the state at time  $t$ ,  $t = 1, \dots, T$ .
  - $s_t$  takes values of  $\{1, \dots, N\}$
- Let  $o_t$  be the observation at time  $t$ .
  - $o_t$  takes values of  $\{e_1, \dots, e_M\}$

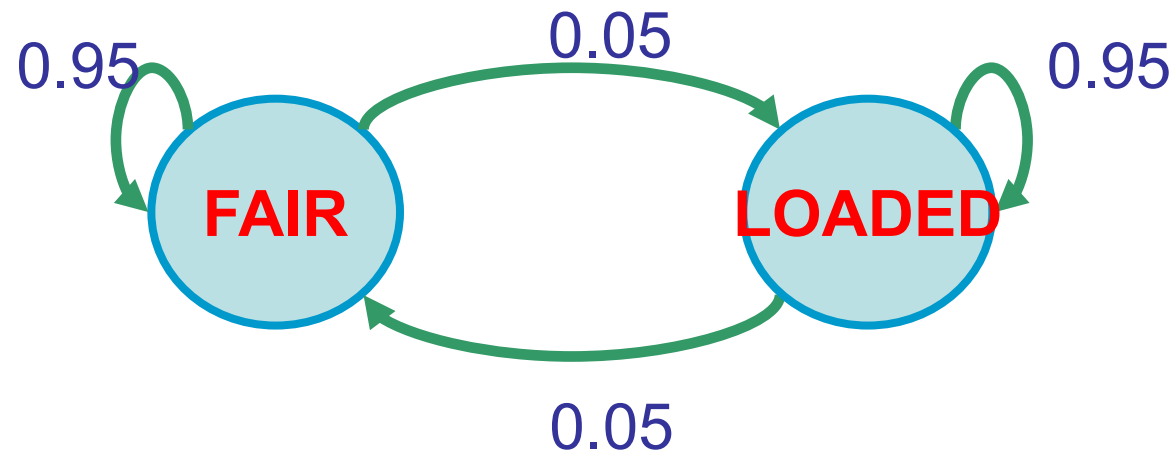


Each node is a  
random variable

# My Dishonest Casino Model

---

- The states (i.e., which dice is used) are hidden.
- We only observe a sequence of rolls, say  
 $O = (3, 6, 5, 1, 6, 6, 3, 6)$
- If the fair dice is red and the loaded dice is blue, then the states are not hidden anymore.



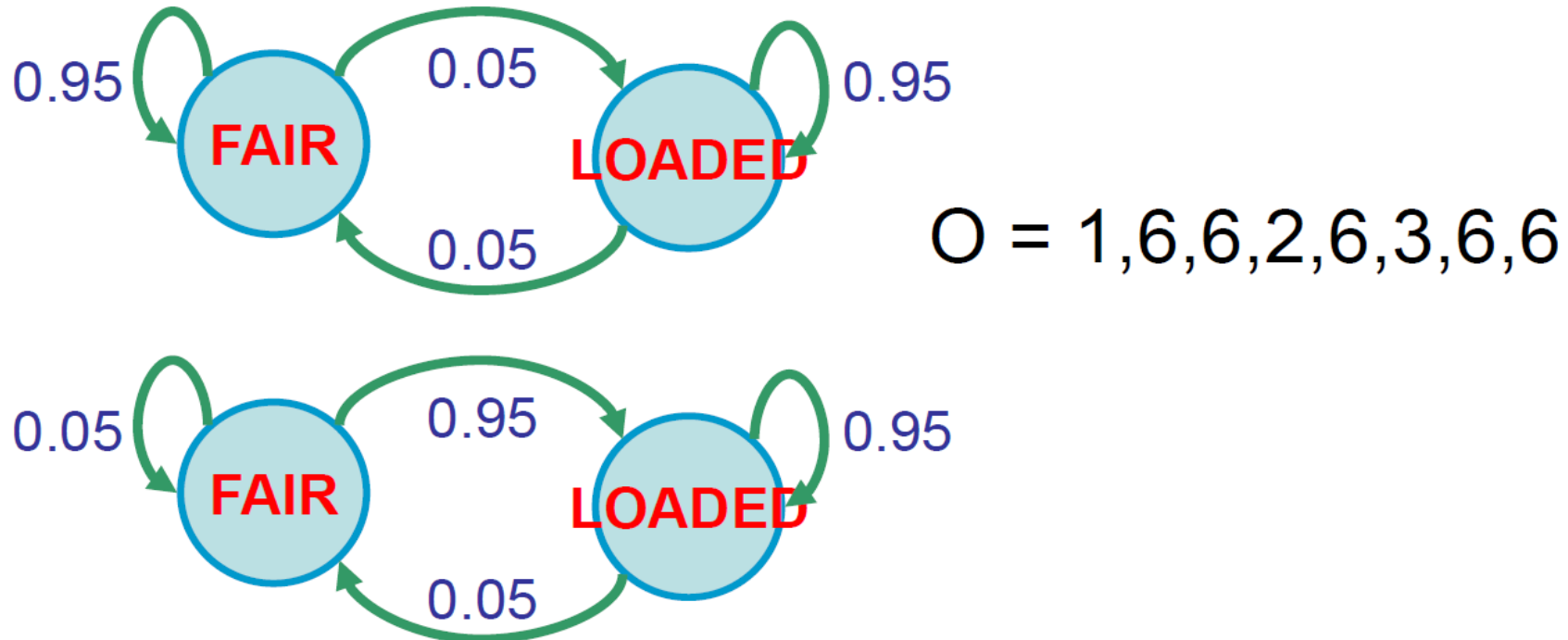
# Key Problems for HMM

---

- Given: observation sequence  $O = (o_1, \dots, o_T)$ , and HMM model  $\lambda = \langle \Pi, A, B \rangle$
- 1) **Evaluation**
  - What is the probability of the observation sequence,  $P(O; \lambda)$ , given the model  $\lambda$ ? Also called the **likelihood** of model to explain the observation.
- 2) **Decoding**
  - What sequence of states  $S = (s_1, \dots, s_T)$  best explains the observation, i.e., maximizes  $P(O, S; \lambda)$ ?
- 3) **Learning**
  - Which model  $\lambda = \langle \Pi, A, B \rangle$  can maximize  $P(O; \lambda)$ ?

# Evaluation

- Given observation  $O$  and HMM  $\lambda = \langle \Pi, A, B \rangle$ , evaluate  $P(O; \lambda)$
- Helps choose the best HMM model



# Naïve way to calculate $P(O; \lambda)$

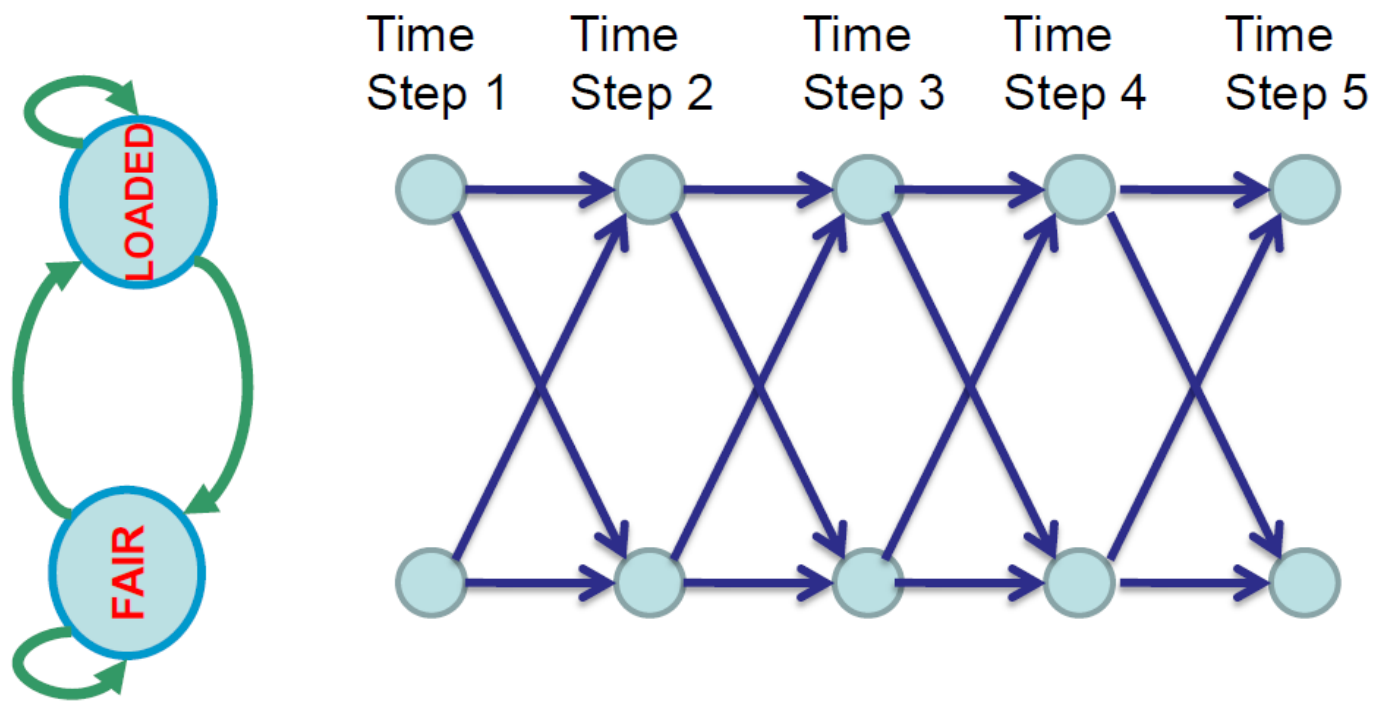
---

$$P(O; \lambda) = \sum_{\text{all possible state sequences } S} P(O, S; \lambda)$$

- How many possible sequences?
  - Sequence length =  $T$ ; state space size =  $N$
  - $N^T$
- Too slow, often intractable!
- We use the **forward algorithm**:  $O(N^2T)$

# The Forward Algorithm

- Idea: Build a trellis that captures all paths through the model so we can reuse probabilities from shared path segments.



# The Idea in Math

---

$$\begin{aligned} P(O_{1:T}) &= \sum_{s_T} P(O_{1:T}, s_T) \\ &= \sum_{s_T} \sum_{s_{T-1}} P(O_{1:T-1}, o_T, s_T, s_{T-1}) \\ &= \sum_{s_T} \sum_{s_{T-1}} P(O_{1:T-1}, s_{T-1}) P(s_T | s_{T-1}) P(o_T | s_T) \end{aligned}$$

Recursion!

Transition probability

Emission probability

# The Forward Algorithm

---

- We compute it by induction
- Let  $\alpha_t(j) = P(O_{1:t}, s_t = j)$ 
  - Initialization:  $\alpha_1(j) = \pi_j P(o_1 | s_1 = j)$ , for  $j = 1, \dots, N$
  - (equivalently:  $\alpha_1(j) = \pi_j b_{j o_1}$ , for  $j = 1, \dots, N$ )
  - Induction: for  $t = 2, \dots, T$  and  $j = 1, \dots, N$ 
$$\alpha_t(j) = \left[ \sum_{i=1}^N \alpha_{t-1}(i) a_{ij} \right] b_{j o_t}$$
  - Termination:  $P(O; \lambda) = \sum_{j=1}^N \alpha_T(j)$

# Decoding

---

- Given observation  $O = (o_1, \dots, o_T)$  and an HMM model  $\lambda = \langle \Pi, A, B \rangle$ , find the state sequence  $S = (s_1, \dots, s_T)$  that best explains the observation, i.e., maximizes  $P(O, S)$ .
- Naïve algorithm
  - Try all possible sequences and choose the best one
  - Too many possible sequences:  $N^T$
- **Viterbi algorithm**
  - Reuse probabilities from shared paths
  - $O(N^2T)$

# The Idea in Math

---

- Very similar to the forward algorithm

$$\begin{aligned} & \max_{S_{1:T}} P(O_{1:T}, S_{1:T}) && \text{Recursion!} \\ = & \max_{S_{1:T}} P(o_T, s_T | O_{1:T-1}, S_{1:T-1}) P(O_{1:T-1}, S_{1:T-1}) \\ = & \max_{S_{1:T}} P(o_T, s_T | s_{T-1}) P(O_{1:T-1}, S_{1:T-1}) \\ = & \max_{S_T} P(o_T | s_T) \max_{S_{1:T-1}} P(s_T | s_{T-1}) P(O_{1:T-1}, S_{1:T-1}) \end{aligned}$$

↑      ↑  
Emission      Transition  
probability      probability

# The Viterbi Algorithm

---

- Let  $v_t(j) = \max_{s_{1:t-1}} P(O_{1:t}, s_{1:t-1}, s_t = j)$
- Initialization:  $v_1(j) = \pi_j P(o_1 | s_1 = j)$ , for  $j = 1, \dots, N$ 
  - (equivalently:  $v_1(j) = \pi_j b_{j o_1}$ , for  $j = 1, \dots, N$ )
  - Induction: for  $t = 2, \dots, T$  and  $j = 1, \dots, N$ 
$$v_t(j) = \left[ \max_i v_{t-1}(i) a_{ij} \right] b_{j o_t}$$
$$prev_t(j) = \arg \max_i v_{t-1}(i) a_{ij}$$
  - Termination:  $P(O, S; \lambda) = \max_j v_T(j)$
  - Trace back from  $\arg \max_j v_T(j)$  to get the best path

# Learning

---

- Given observation  $O = (o_1, \dots, o_T)$ , what are the best parameters of an HMM model  $\lambda = \langle \boldsymbol{\Pi}, \mathbf{A}, \mathbf{B} \rangle$  that can maximize  $P(O; \lambda)$ ?
- The parameters  $\lambda = \langle \boldsymbol{\Pi}, \mathbf{A}, \mathbf{B} \rangle$  are unknown
- The hidden states  $S = (s_1, \dots, s_T)$  are unknown
- Baum-Welch algorithm
  - EM algorithm!

# Continuous Observations

---

- In the previous slides, we assumed a **discrete** emission (observation) alphabet  $\{e_1, \dots, e_M\}$ .
- What if the observation alphabet is **continuous**, e.g., real-valued?
- How do we represent emission probabilities  $B$ ?
- **Parameterized** model  $p(o_t|s_t)$

# Audio Modeling by HMMs

---

- Speech recognition
  - States: phonemes
  - Observation: MFCC features of audio frames
  - Transition probabilities: phonemes transition
  - Emission probabilities: phoneme -> audio spectrum
  - Recognition: decoding states from observed audio frames

# Audio Modeling by HMMs

---

- Chord recognition
  - States: chords
  - Observation: some feature representation of audio spectra
  - Transition probabilities: chord progression
  - Emission probabilities: chord -> audio spectrum
  - Recognition: decoding chord labels from observed audio frames

# Audio Modeling by HMMs

---

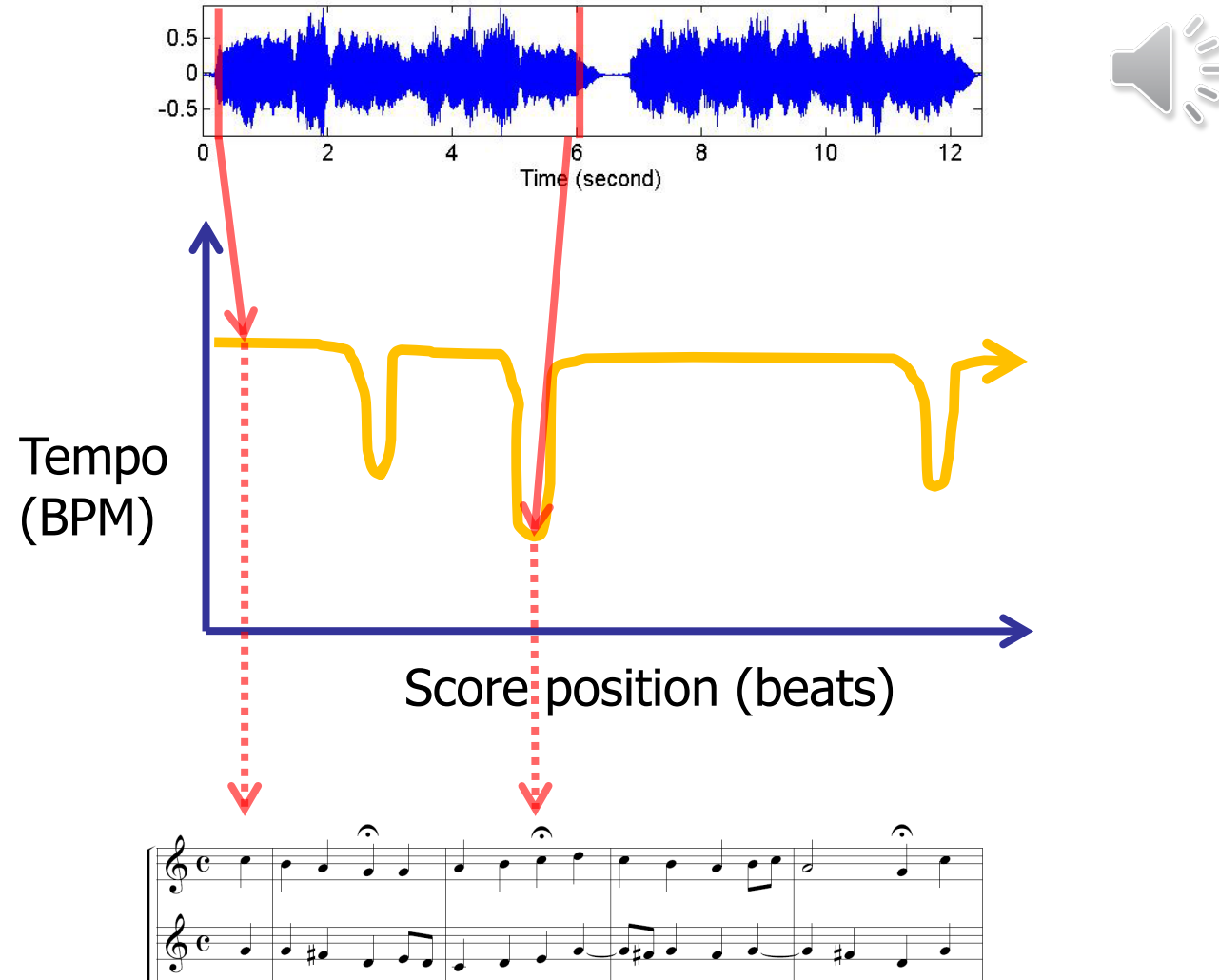
- Refining pitch detection results
  - States: pitch candidates (e.g., all discretized freq. between 65Hz-370Hz)
  - Observation: audio spectra
  - Transition probabilities: pitches tend to change smoothly
  - Emission probabilities: the likelihood of each pitch candidate,  $P(\text{audio frame} \mid \text{pitch candidate})$
  - Refinement: decoding pitches from observation

# Infinite-state HMM

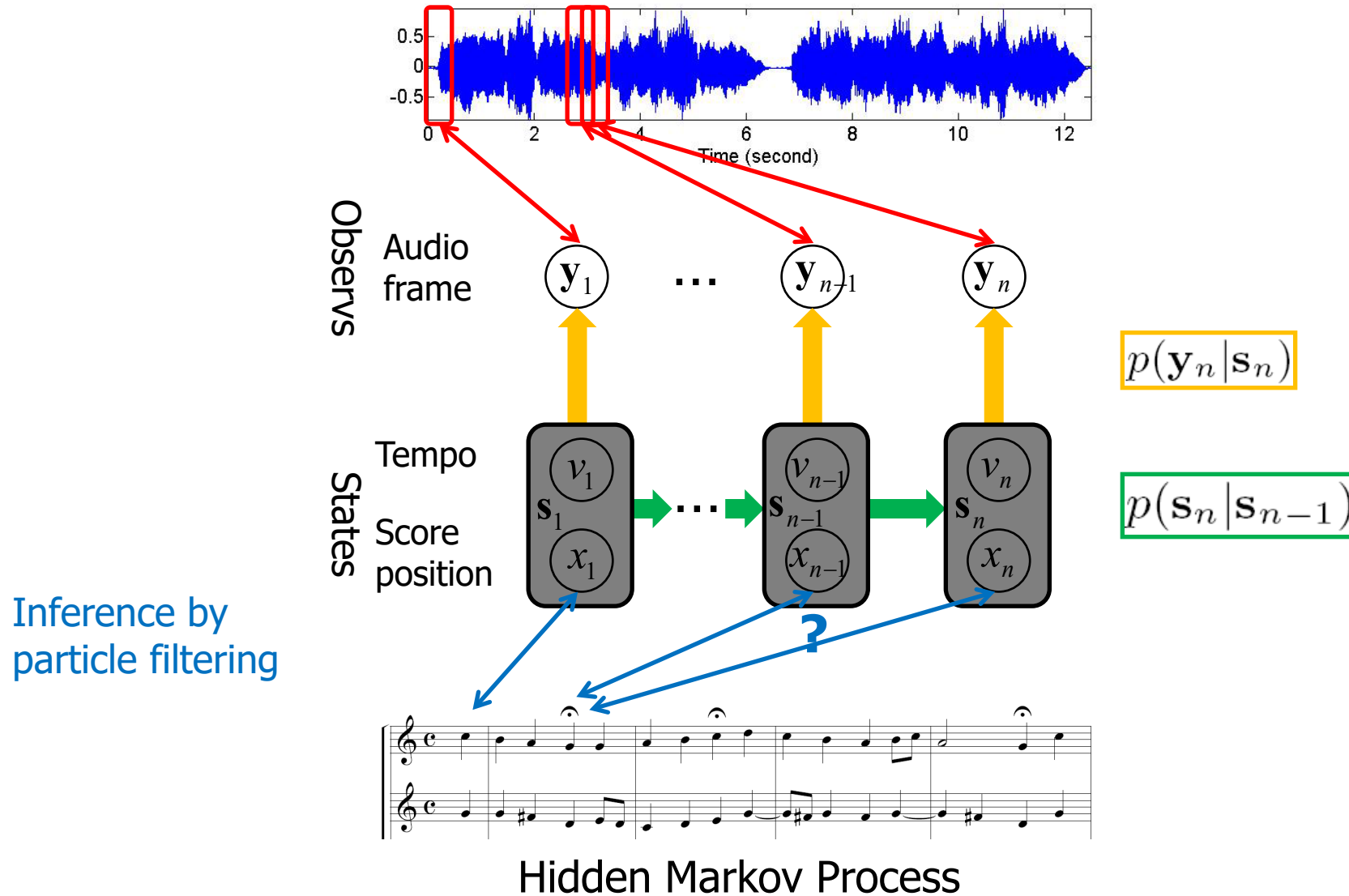
---

- There are infinitely many states, also called **hidden Markov process**.
- Summations over states in finite-state HMMs become to integrations over states.
- When the states are high-dimensional, integration is not easy.
  - Use Monte Carlo methods instead

# An Example: Audio-score Alignment



# An Example: Audio-score Alignment



# Limitations of HMM

---

- Only models short-time dependencies
  - Audio signals can have longer dependencies, e.g., rhythmic structure
  - Higher-order HMM
- Only one sequence of states
  - Audio with multiple sound sources?
  - Factorial HMM
- Generative model
  - May not be ideal for some tasks
  - Conditional Random Field (CRF)