HMMs for Acoustic Modeling (Part I)

Lecture 2

CS 753

Instructor: Preethi Jyothi

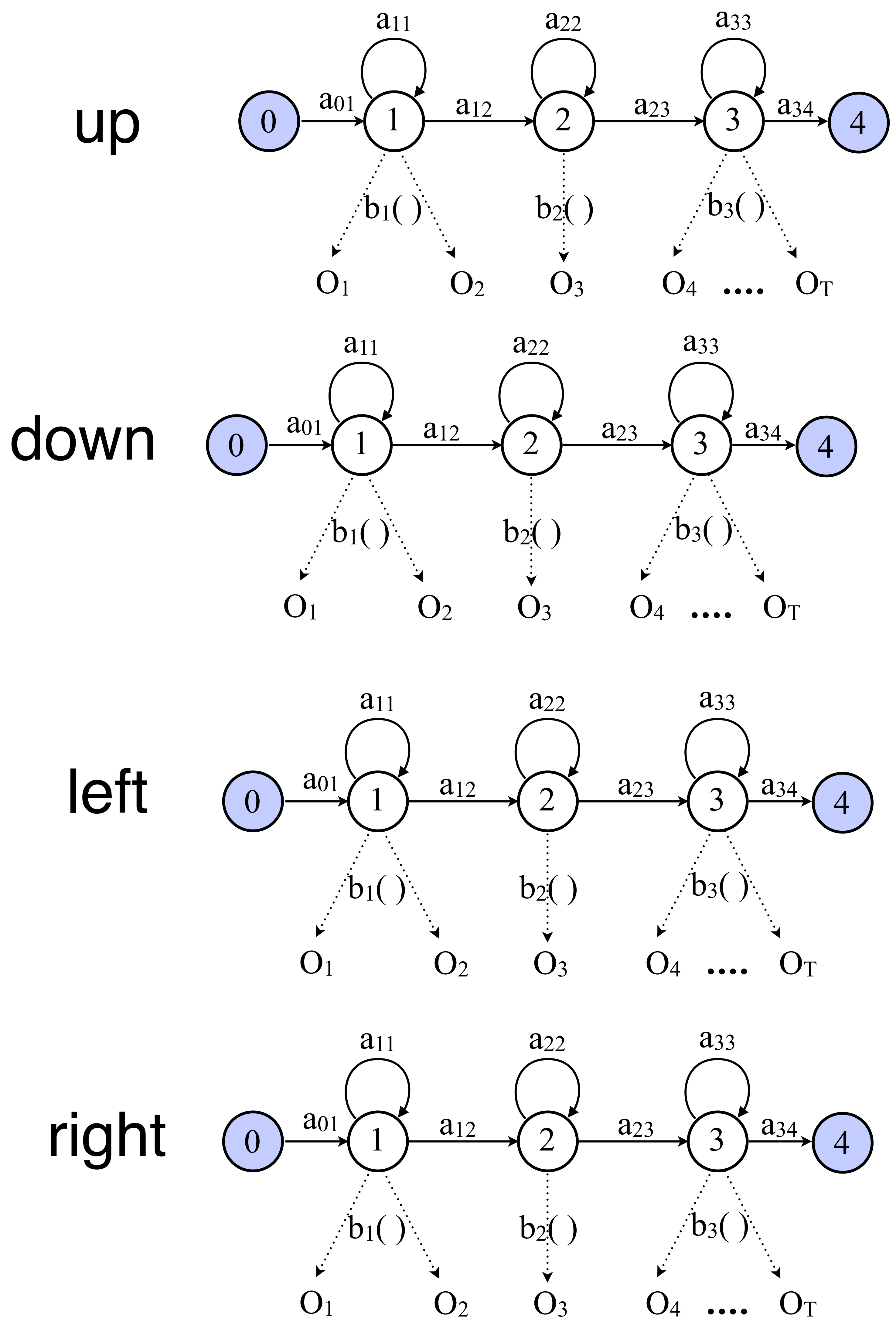
Recall: Statistical ASR

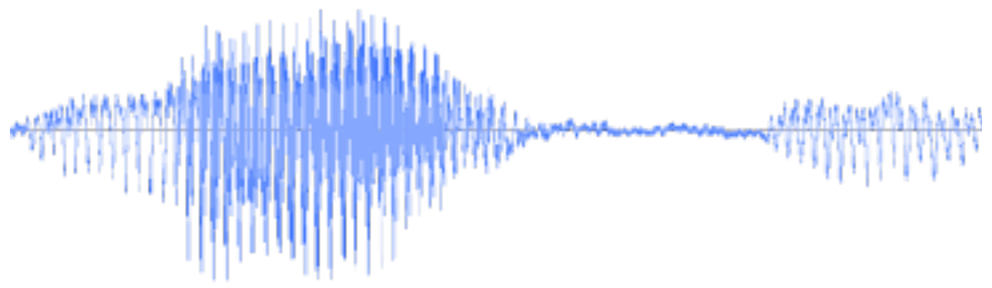
Let $\mathbf{O}$ be a sequence of acoustic features corresponding to a speech signal. That is, $\mathbf{O} = \{O_1, \ldots, O_T\}$, where $O_i \in \mathbb{R}^d$ refers to a d-dimensional acoustic feature vector and T is the length of the sequence.

Let $\mathbf{W}$ denote a word sequence. An ASR decoder solves the foll. problem:

$$
\begin{aligned}
\mathbf{W}^* &= \arg\max_{W} \Pr(\mathbf{W} \mid \mathbf{O}) \\
&= \arg\max_{W} \underbrace{\Pr(\mathbf{O} \mid \mathbf{W})}_{\text{Acoustic Model}} \overbrace{\Pr(\mathbf{W})}^{\text{Language Model}}
\end{aligned}
$$

Isolated word recognition





acoustic
features
$\mathbf{O}$

$\Pr(\mathbf{O}\,|\,\text{"}\mathit{up}\text{"})$

$\Pr(\mathbf{O}\,|\,\text{"}\mathit{down}\text{"})$

Compute $\arg\max_{w} \Pr(\mathbf{O}\,|\,w)$

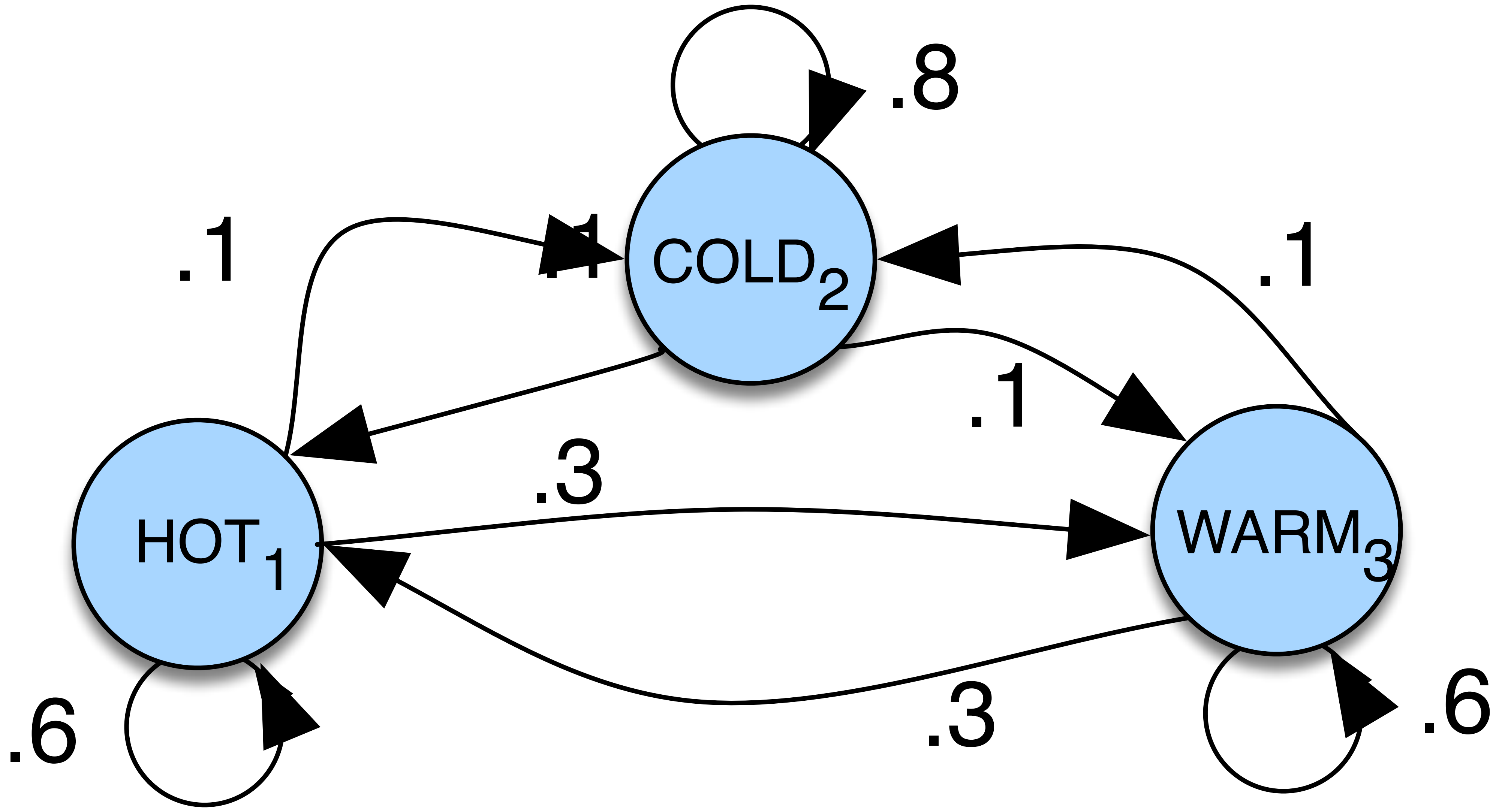
$\Pr(\mathbf{O}\,|\,\text{"}\mathit{left}\text{"})$

$\Pr(\mathbf{O}\,|\,\text{"}\mathit{right}\text{"})$

What are Hidden Markov Models (HMMs)?

Following slides contain figures/material from “Hidden Markov Models”, “Speech and Language Processing”, D. Jurafsky and J. H. Martin, 2019. (https://web.stanford.edu/~jurafsky/slp3/A.pdf)

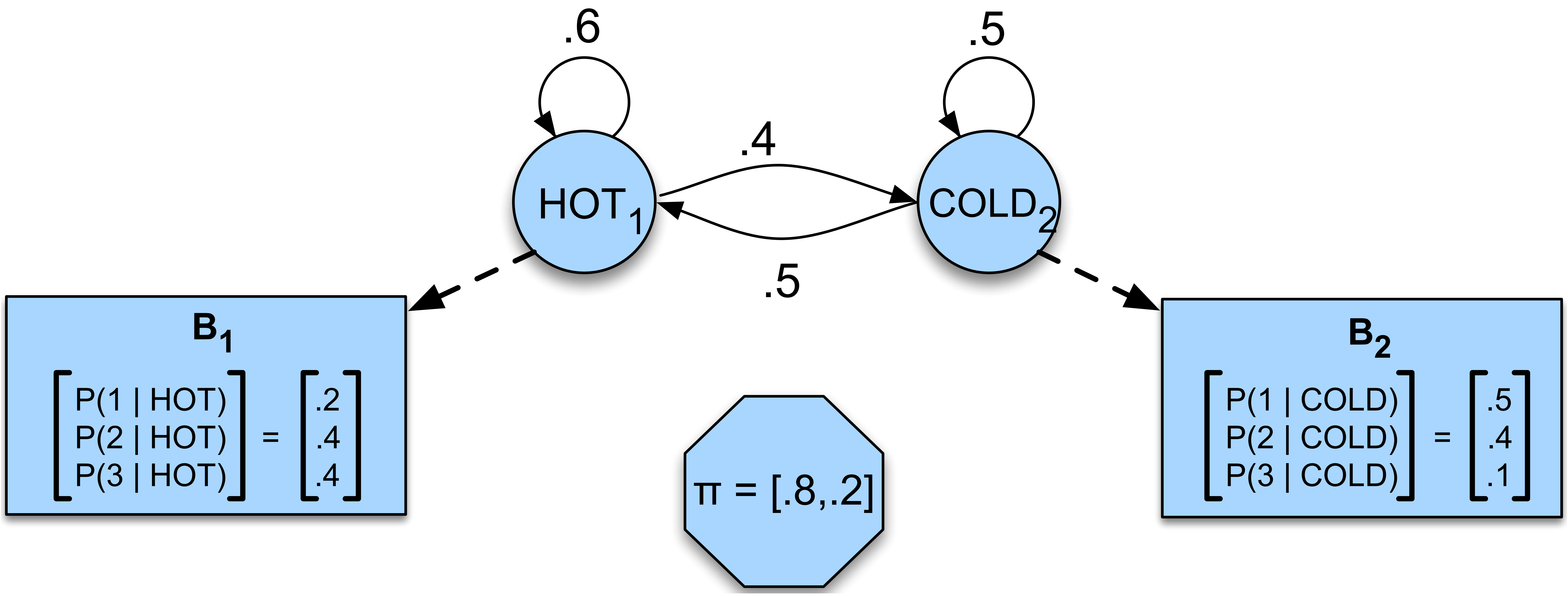
Markov Chains



$$\pi = [0.1, 0.7, 0.2]$$

$Q = q_1 q_2 \ldots q_N$	a set of N **states**
$A = a_{11} a_{12} \ldots a_{n1} \ldots a_{nn}$	a **transition probability matrix** A, each a_{ij} representing the probability of moving from state i to state j, s.t. $\sum_{j=1}^{n} a_{ij} = 1 \quad \forall i$
$\pi = \pi_1, \pi_2, ..., \pi_N$	an **initial probability distribution** over states. π_i is the probability that the Markov chain will start in state i. Some states j may have $\pi_j = 0$, meaning that they cannot be initial states. Also, $\sum_{i=1}^{n} \pi_i = 1$

HMM Assumptions



Markov Assumption: $P(q_i|q_1 ... q_{i-1}) = P(q_i|q_{i-1})$

Output Independence: $P(o_i|q_1 \ldots q_i, \ldots, q_T, o_1, \ldots, o_i, \ldots, o_T) = P(o_i|q_i)$

Hidden Markov Model

$Q = q_1 q_2 \ldots q_N$	a set of N **states**
$A = a_{11} \ldots a_{ij} \ldots a_{NN}$	a **transition probability matrix** A, each a_{ij} representing the probability of moving from state i to state j, s.t. $\sum_{j=1}^{N} a_{ij} = 1 \quad \forall i$
$O = o_1 o_2 \ldots o_T$	a sequence of T **observations**, each one drawn from a vocabulary $V = v_1, v_2, ..., v_V$
$B = b_i(o_t)$	a sequence of **observation likelihoods**, also called **emission probabilities**, each expressing the probability of an observation o_t being generated from a state i
$\pi = \pi_1, \pi_2, ..., \pi_N$	an **initial probability distribution** over states. π_i is the probability that the Markov chain will start in state i. Some states j may have $\pi_j = 0$, meaning that they cannot be initial states. Also, $\sum_{i=1}^{n} \pi_i = 1$

Three problems for HMMs

Problem 1 (Likelihood):	Given an HMM $\lambda = (A, B)$ and an observation sequence O, determine the likelihood $P(O\|\lambda)$.
Problem 2 (Decoding):	Given an observation sequence O and an HMM $\lambda = (A, B)$, discover the best hidden state sequence Q.
Problem 3 (Learning):	Given an observation sequence O and the set of states in the HMM, learn the HMM parameters A and B.

Computing Likelihood: Given an HMM $\lambda = (A, B)$ and an observation sequence O, determine the likelihood $P(O|\lambda)$.

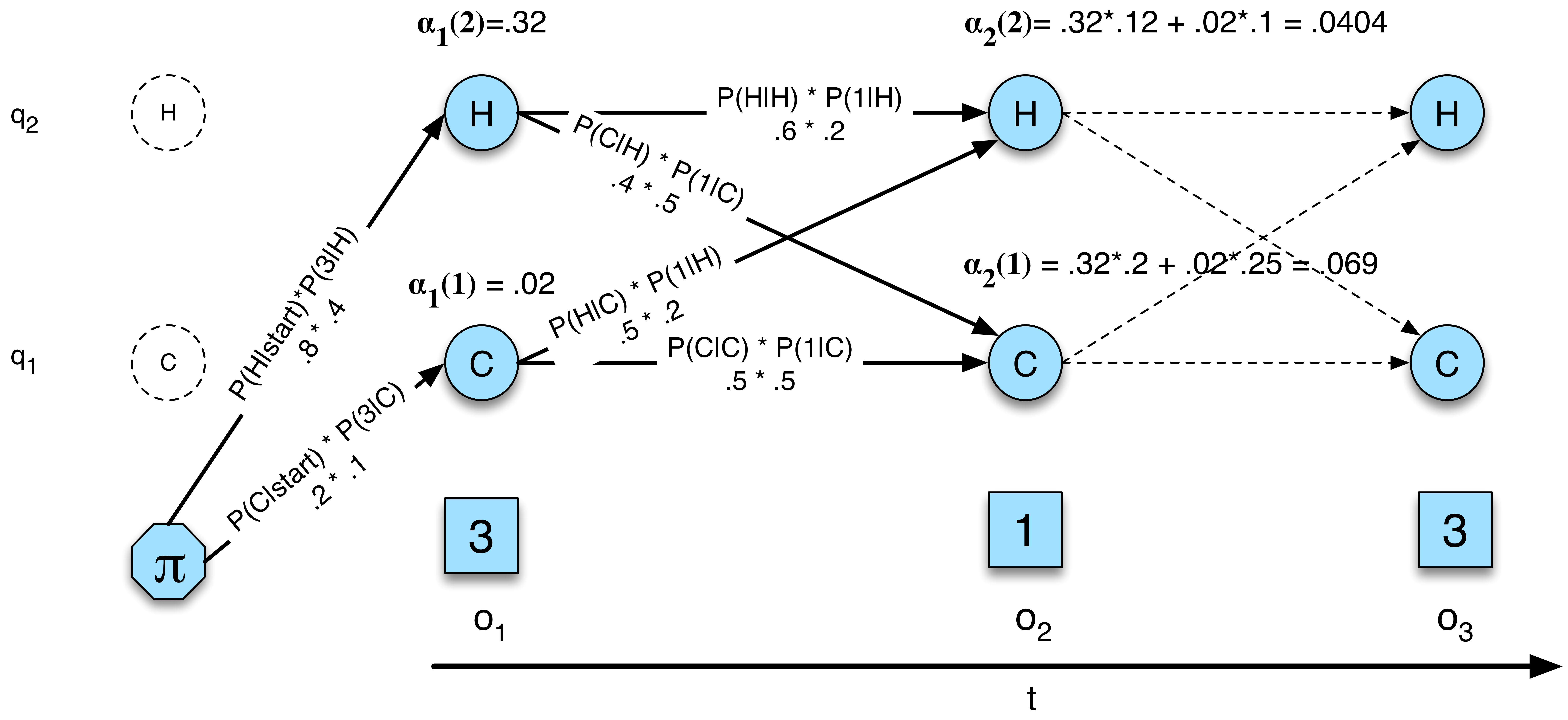
"A tutorial on hidden Markov models and selected applications in speech recognition", Rabiner, 1989

Forward Algorithm

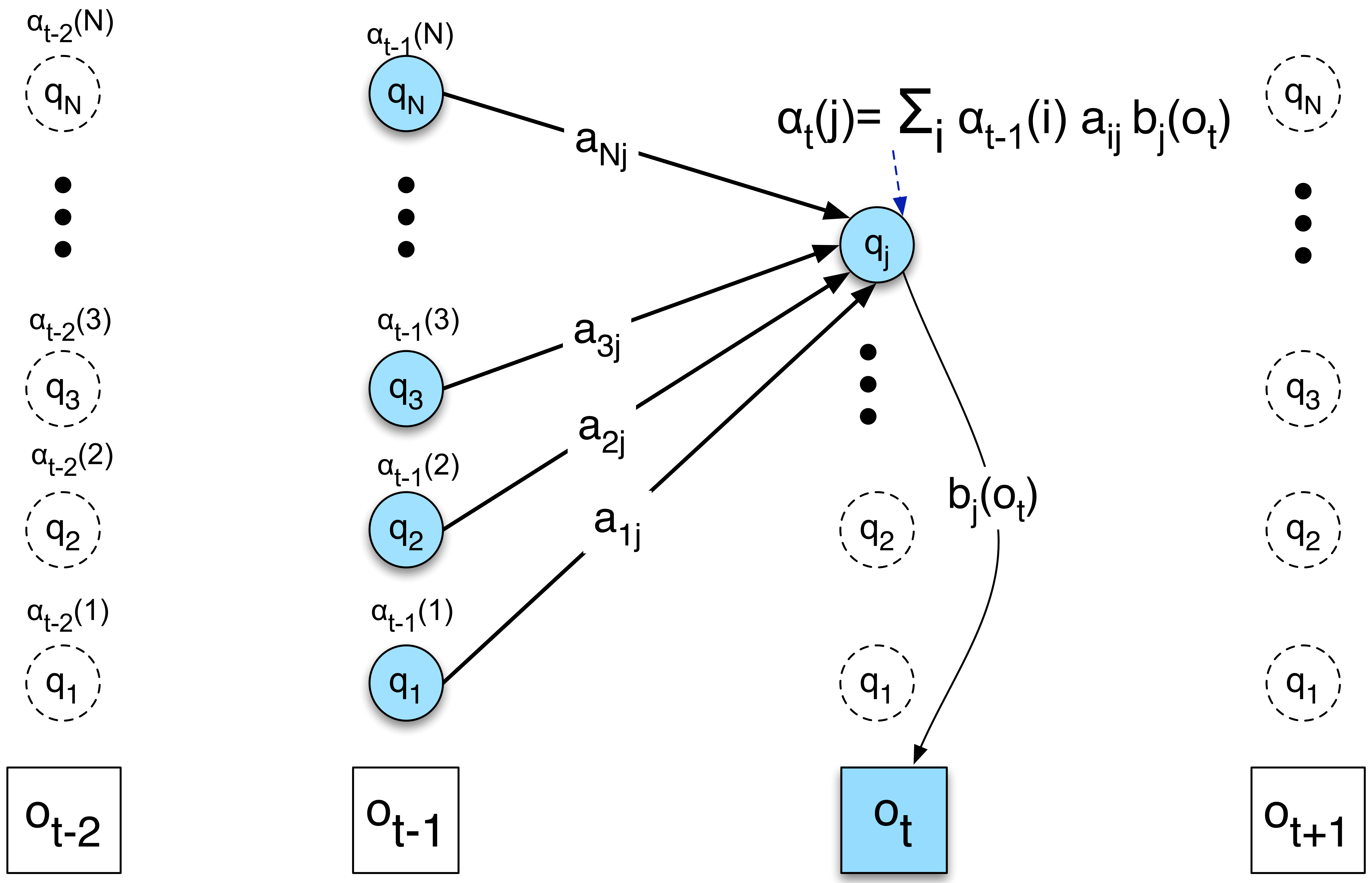
Forward Probability

$$\alpha_t(j) = P(o_1, o_2 \ldots o_t, q_t = j|\lambda)$$

$$\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}(i)a_{ij}b_j(o_t)$$



Visualizing the forward recursion



Forward Algorithm

1. Initialization:

$$\alpha_1(j) \;=\; \pi_j b_j(o_1) \;\; 1 \leq j \leq N$$

2. Recursion:

$$\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} b_j(o_t); \quad 1 \leq j \leq N, 1 < t \leq T$$

3. Termination:

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

Three problems for HMMs

Problem 1 (Likelihood):	Given an HMM $\lambda = (A, B)$ and an observation sequence O, determine the likelihood $P(O\|\lambda)$.
Problem 2 (Decoding):	Given an observation sequence O and an HMM $\lambda = (A, B)$, discover the best hidden state sequence Q.
Problem 3 (Learning):	Given an observation sequence O and the set of states in the HMM, learn the HMM parameters A and B.

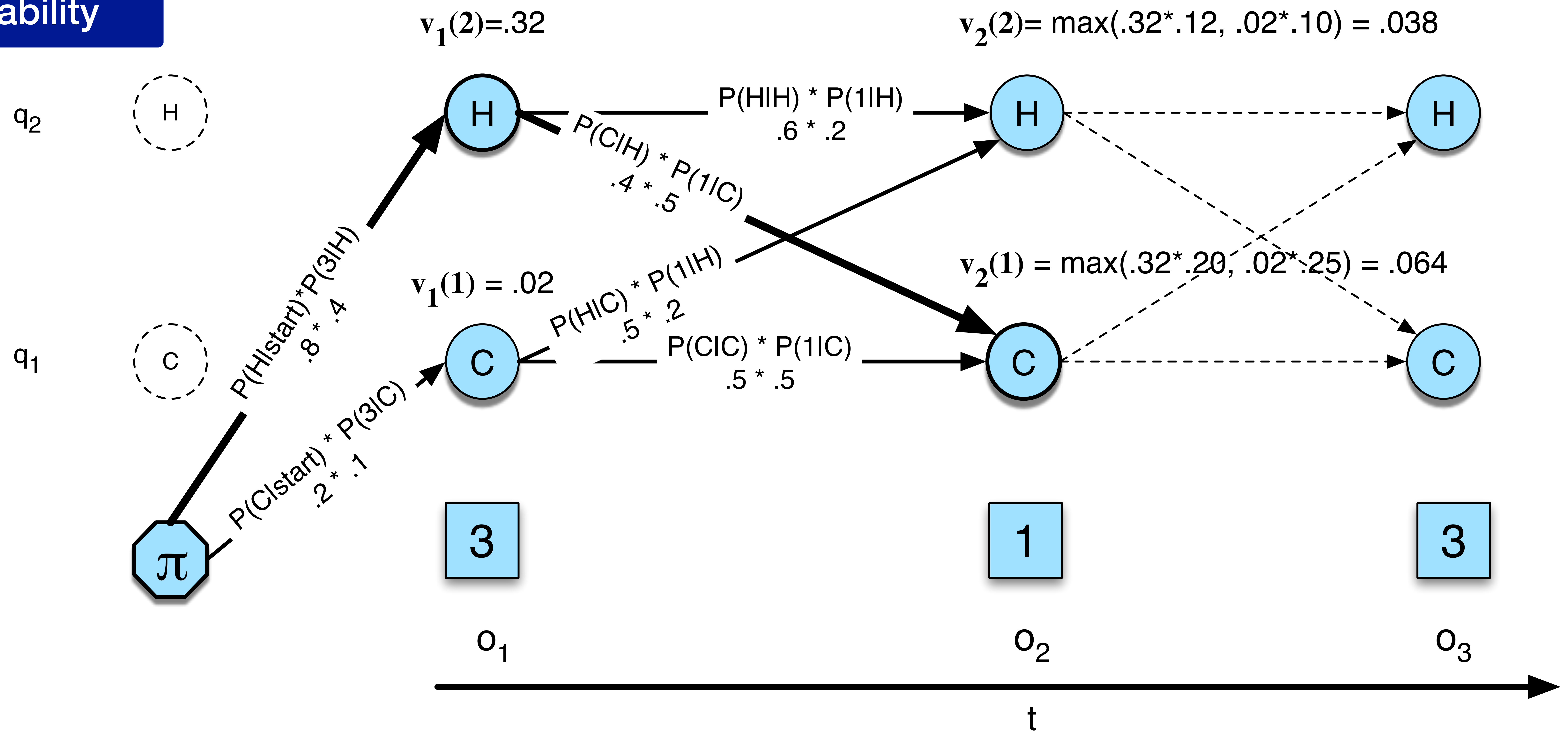
Decoding: Given as input an HMM $\lambda = (A, B)$ and a sequence of observations $O = o_1, o_2, ..., o_T$, find the most probable sequence of states $Q = q_1 q_2 q_3 \ldots q_T$.

Viterbi Trellis

$$v_t(j) = \max_{q_1,\ldots,q_{t-1}} P(q_1 \ldots q_{t-1}, o_1, o_2 \ldots o_t, q_t = j|\lambda)$$

$$v_t(j) \;=\; \max_{i=1}^{N} v_{t-1}(i)\, a_{ij}\, b_j(o_t)$$

Viterbi Path Probability



Viterbi recursion

1. **Initialization:**

$$
\begin{aligned}
v_1(j) &= \pi_j b_j(o_1) \qquad 1 \leq j \leq N \\
bt_1(j) &= 0 \qquad\qquad\quad 1 \leq j \leq N
\end{aligned}
$$

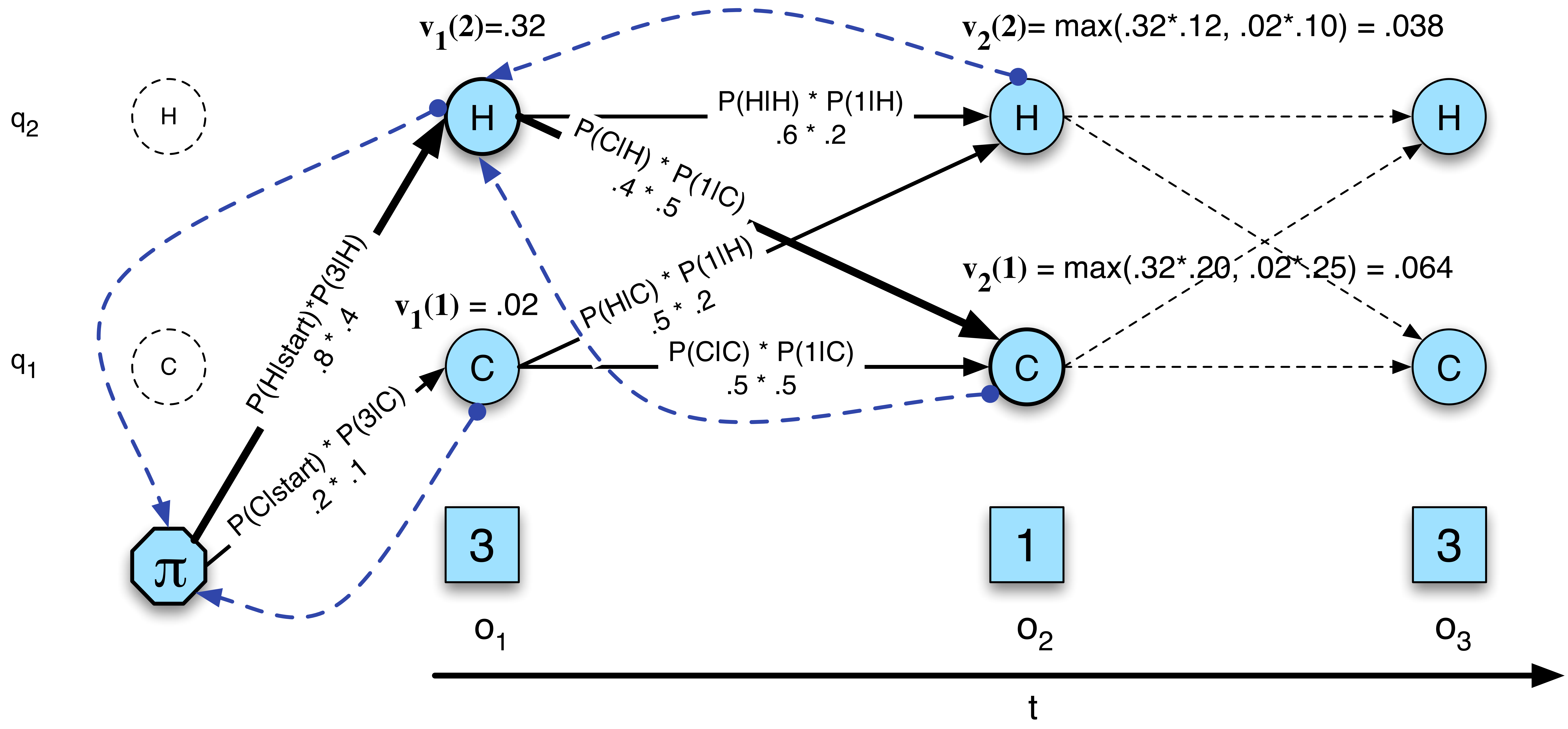
2. **Recursion**

$$
\begin{aligned}
v_t(j) &= \max_{i=1}^{N} v_{t-1}(i)\, a_{ij}\, b_j(o_t); \quad 1 \leq j \leq N, 1 < t \leq T \\
bt_t(j) &= \operatorname*{argmax}_{i=1}^{N} v_{t-1}(i)\, a_{ij}\, b_j(o_t); \quad 1 \leq j \leq N, 1 < t \leq T
\end{aligned}
$$

3. **Termination:**

$$
\begin{aligned}
\text{The best score:} \quad P* &= \max_{i=1}^{N} \; v_T(i) \\
\text{The start of backtrace:} \quad q_T* &= \operatorname*{argmax}_{i=1}^{N} \; v_T(i)
\end{aligned}
$$

Viterbi backtrace



Gaussian Observation Model

- So far, we considered HMMs with discrete outputs
- In acoustic models, HMMs output real valued vectors
- Hence, observation probabilities are defined using probability density functions
- A widely used model: Gaussian distribution

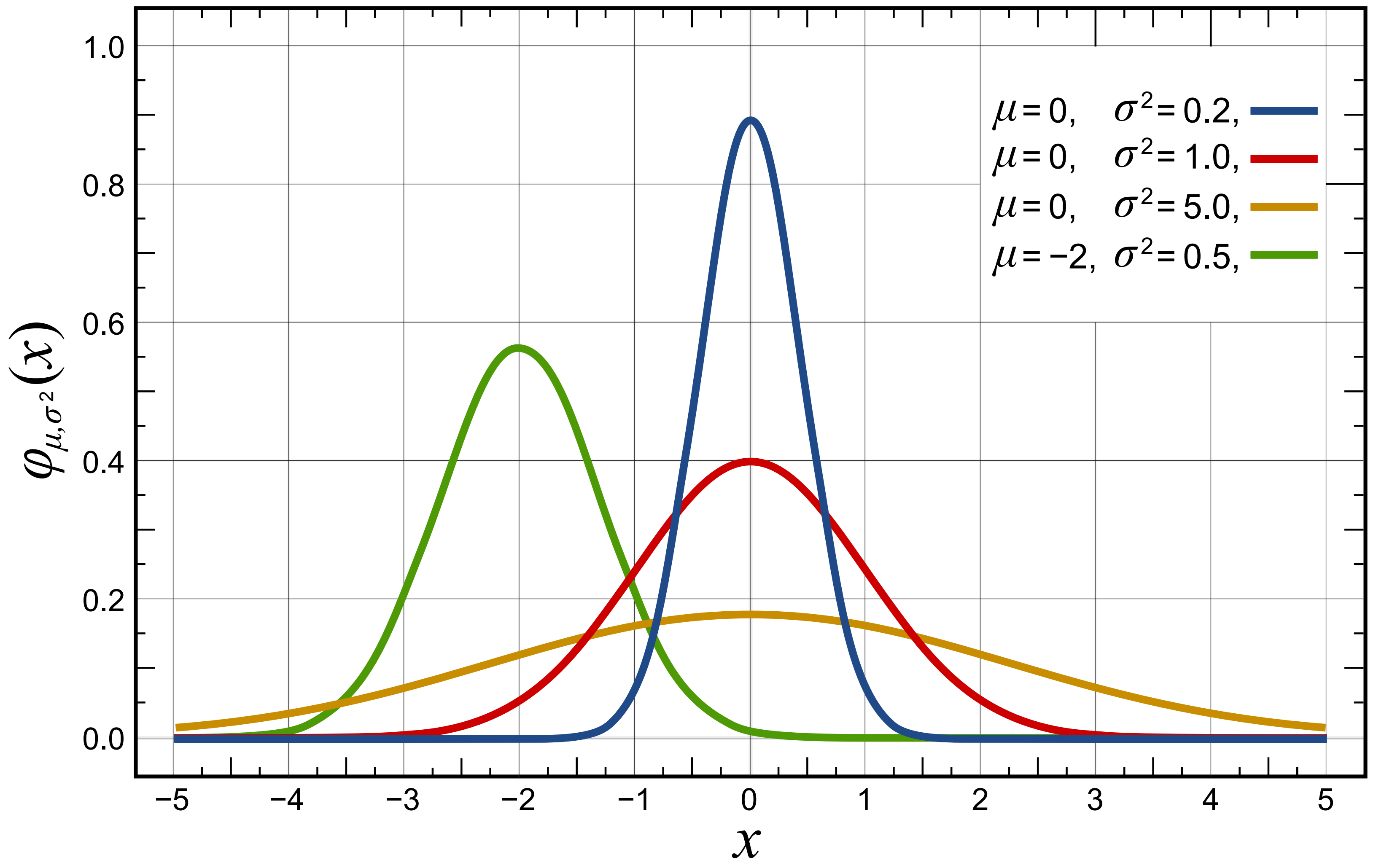
$$\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}$$

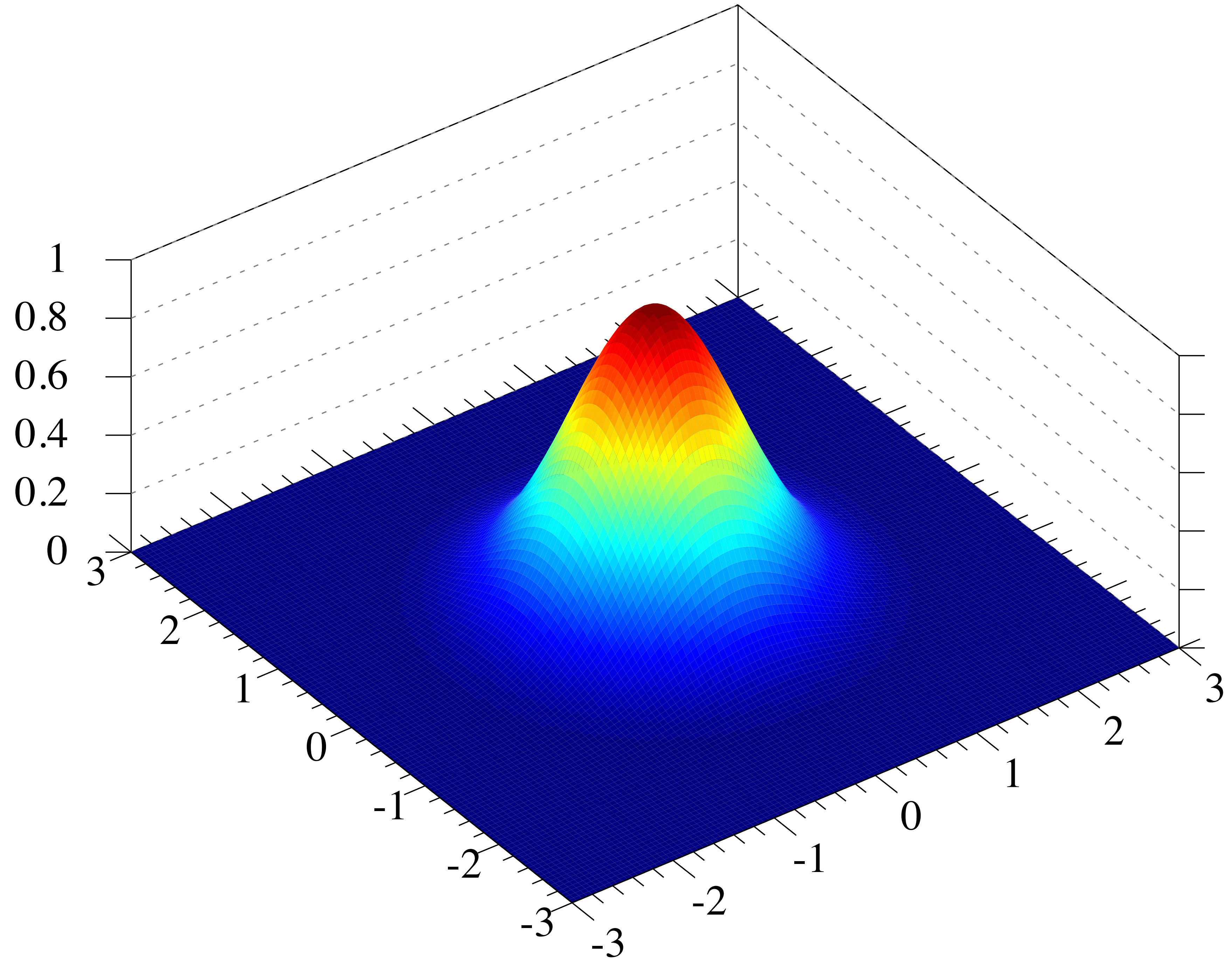
- HMM emission/observation probabilities $b_j(x) = \mathcal{N}(x \mid \mu_j, \sigma_j^2)$ where μ_j is the mean associated with state j and σ_j^2 is its variance
- For multivariate Gaussians, $b_j(\mathbf{x}) = \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_j, \Sigma_j)$ where Σ_j is the covariance matrix associated with state j

Gaussian Mixture Model

- A single Gaussian observation model assumes that the observed acoustic feature vectors are unimodal

Unimodal

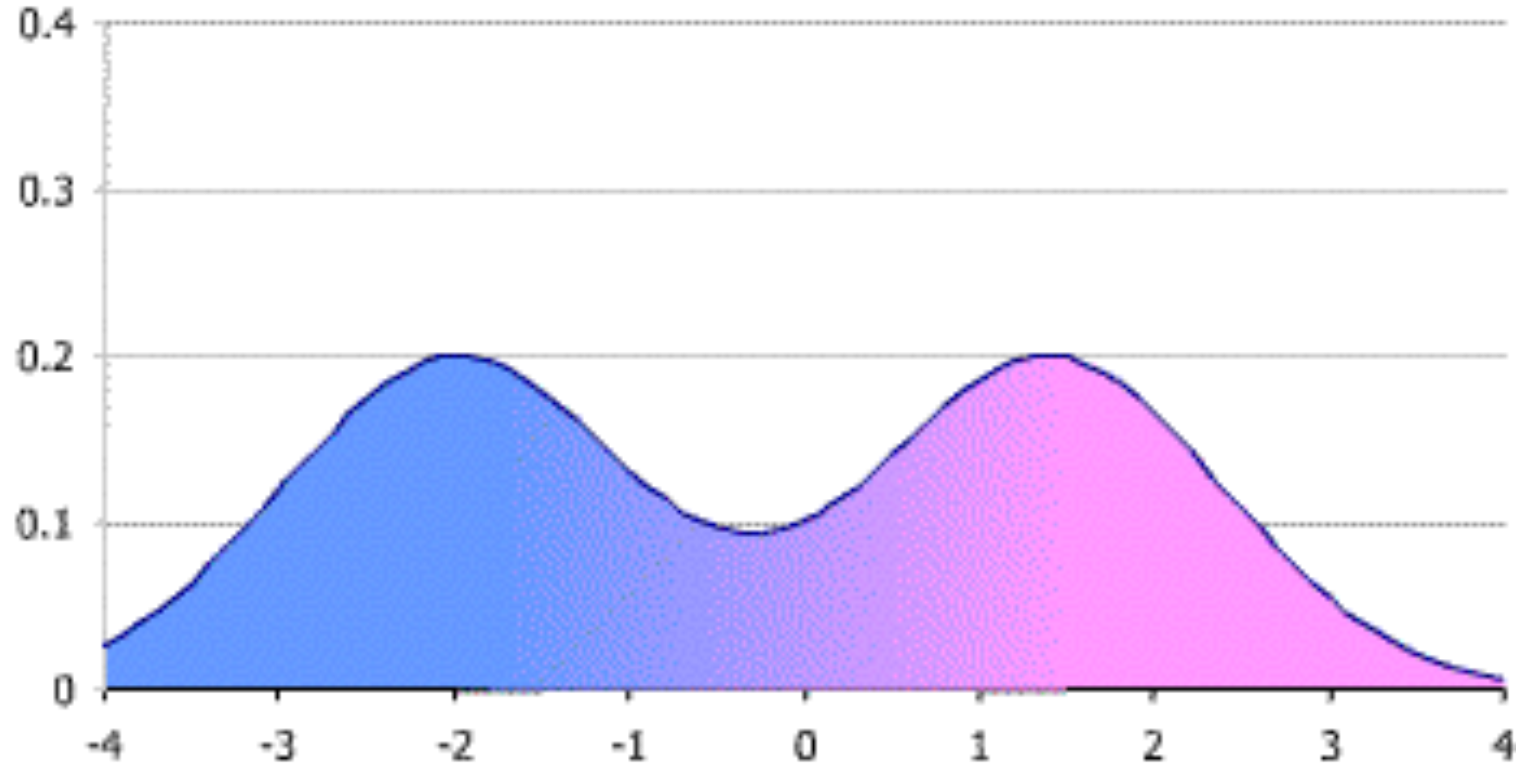

$\mu=0,\quad \sigma^2=0.2,$
$\mu=0,\quad \sigma^2=1.0,$
$\mu=0,\quad \sigma^2=5.0,$
$\mu=-2,\ \sigma^2=0.5,$
$\varphi_{\mu,\sigma^2}(x)$
x
1.0
0.8
0.6
0.4
0.2
0.0
−5
−4
−3
−2
−1
0
1
2
3
4
5

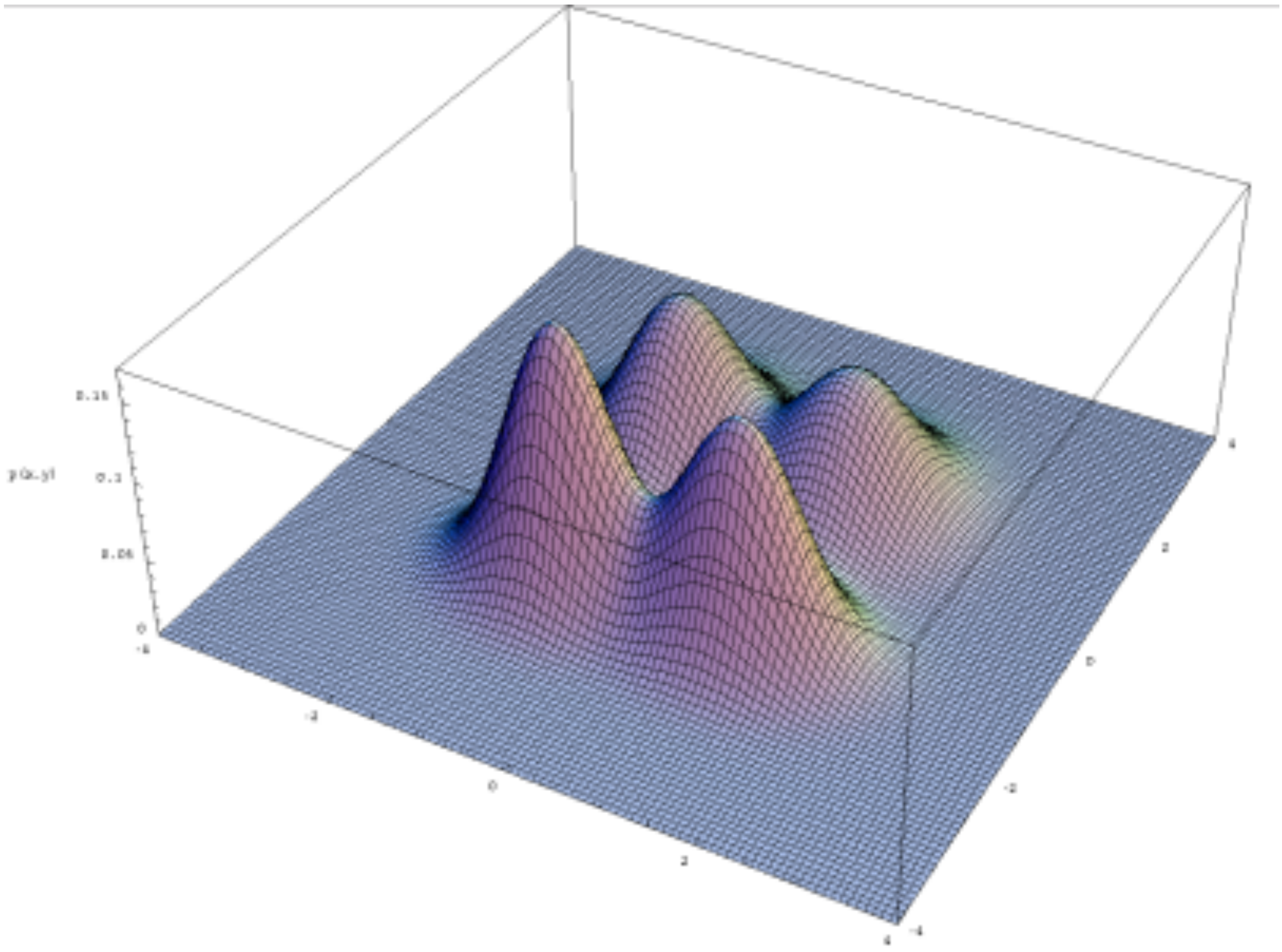

1
0.8
0.6
0.4
0.2
0
3
2
1
0
-1
-2
-3
-3
-2
-1
0
1
2
3

Gaussian Mixture Model

- A single Gaussian observation model assumes that the observed acoustic feature vectors are unimodal
- More generally, we use a “mixture of Gaussians” to model multiple modes in the data

Mixture Models





Gaussian Mixture Model

- A single Gaussian observation model assumes that the observed acoustic feature vectors are unimodal
- More generally, we use a "mixture of Gaussians" to model multiple modes in the data
- Instead of $b_j(\mathbf{x}) = \mathcal{N}(\mathbf{x} \,|\, \boldsymbol{\mu}_j, \Sigma_j)$ in the single Gaussian case, $b_j(\mathbf{x})$ now becomes:

$$b_j(\mathbf{x}) = \sum_{m=1}^{M} c_{jm} \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_{jm}, \Sigma_{jm})$$

where c_{jm} is the mixing probability for Gaussian component m of state j

$$\sum_{m=1}^{M} c_{jm} = 1, \quad c_{jm} \geq 0$$