Machine Learning for Signal Processing

Hidden Markov Models

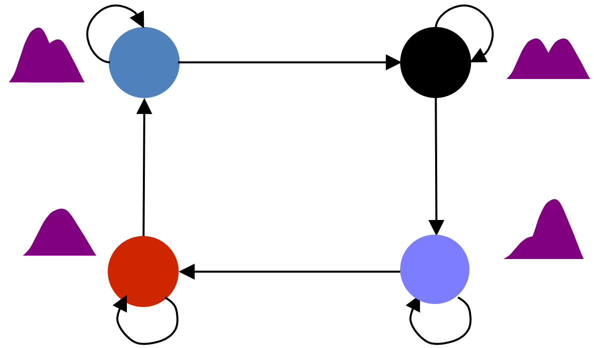
Yuxuan Wu and Zhongyuan Zhai

Problem 2: State segmentation

- Given only a sequence of observations, how do we determine which sequence of states was followed in producing it?

The HMM as a generator

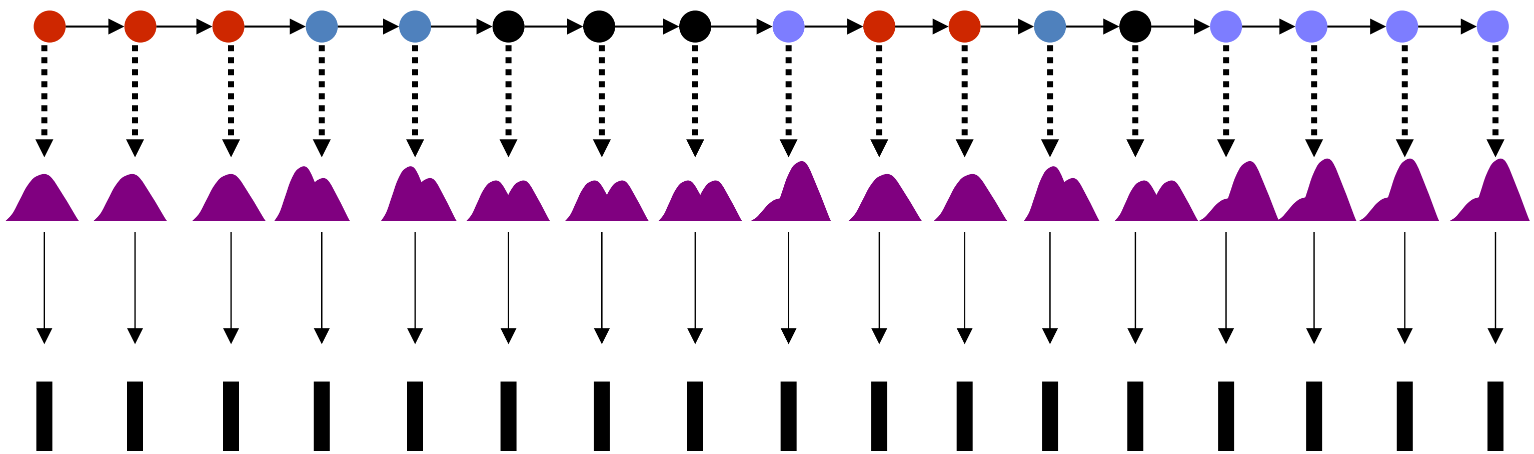
HMM assumed to be generating data



state sequence

state distributions

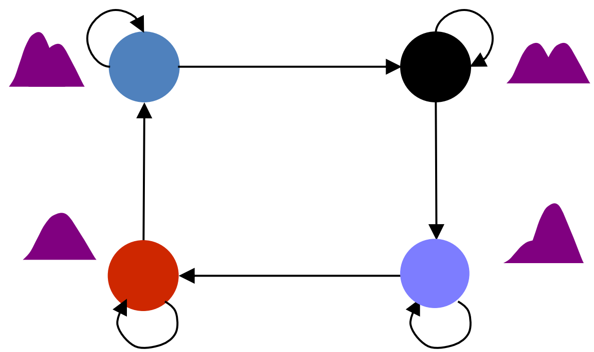
observation sequence



- The process goes through a series of states and produces observations from them

States are hidden

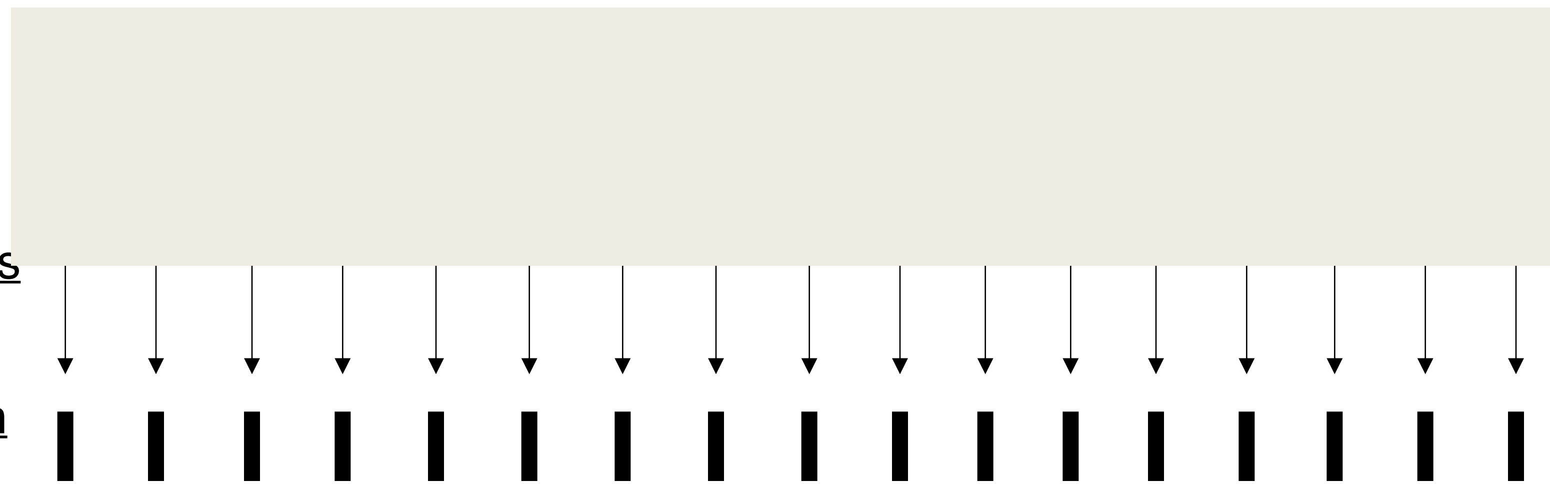
HMM assumed to be generating data



state sequence

state distributions

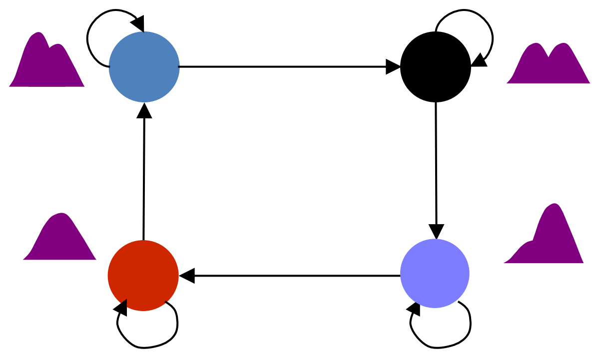
observation sequence



- The observations do not reveal the underlying state

The state segmentation problem

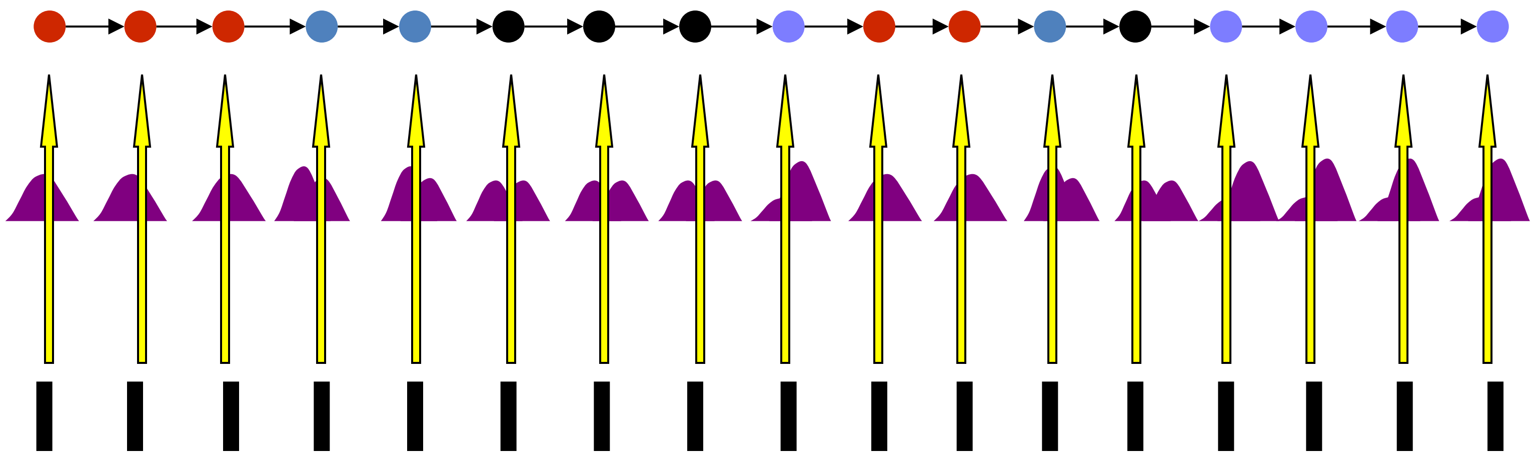
HMM assumed to be generating data



state sequence

state distributions

observation sequence



- State segmentation: Estimate state sequence given observations

Estimating the State Sequence

- Many different state sequences are capable of producing the observation

- Solution: Identify the most *probable* state sequence
  - The state sequence for which the probability of progressing through that sequence and generating the observation sequence is maximum
  - i.e $P(o_1, o_2, o_3, \ldots, s_1, s_2, s_3, \ldots)$ is maximum

Estimating the state sequence

- Once again, exhaustive evaluation is impossibly expensive
- But once again a simple dynamic-programming solution is available

$$P(o_1, o_2, o_3, \ldots, s_1, s_2, s_3, \ldots) =$$

$$\underline{P(o_1|s_1)P(o_2|s_2)P(o_3|s_3)\ldots}\;\underline{P(s_1)P(s_2|s_1)P(s_3|s_2)\ldots}$$

- Needed:

$$\arg\max_{s_1, s_2, s_3, \ldots} P(o_1 \mid s_1)P(s_1)P(o_2 \mid s_2)P(s_2 \mid s_1)P(o_3 \mid s_3)P(s_3 \mid s_2)$$

Estimating the state sequence

- Once again, exhaustive evaluation is impossibly expensive
- But once again a simple dynamic-programming solution is available

$$P(o_1, o_2, o_3, \ldots, s_1, s_2, s_3, \ldots) =$$

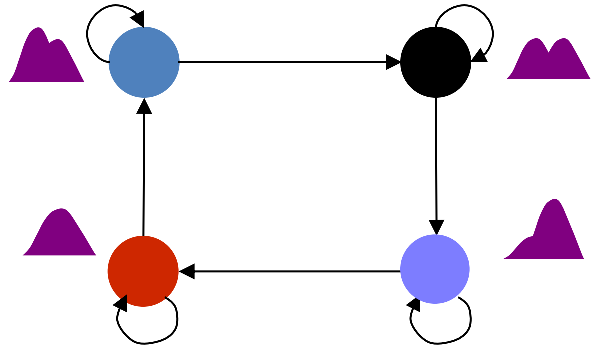
$$\underline{P(o_1|s_1)P(o_2|s_2)P(o_3|s_3)\ldots}\,\underline{P(s_1)P(s_2|s_1)P(s_3|s_2)\ldots}$$

- Needed:

$$\arg\max_{s_1, s_2, s_3, \ldots} P(o_1 | s_1)P(s_1)P(o_2 | s_2)P(s_2 | s_1)P(o_3 | s_3)P(s_3 | s_2)$$

The HMM as a generator

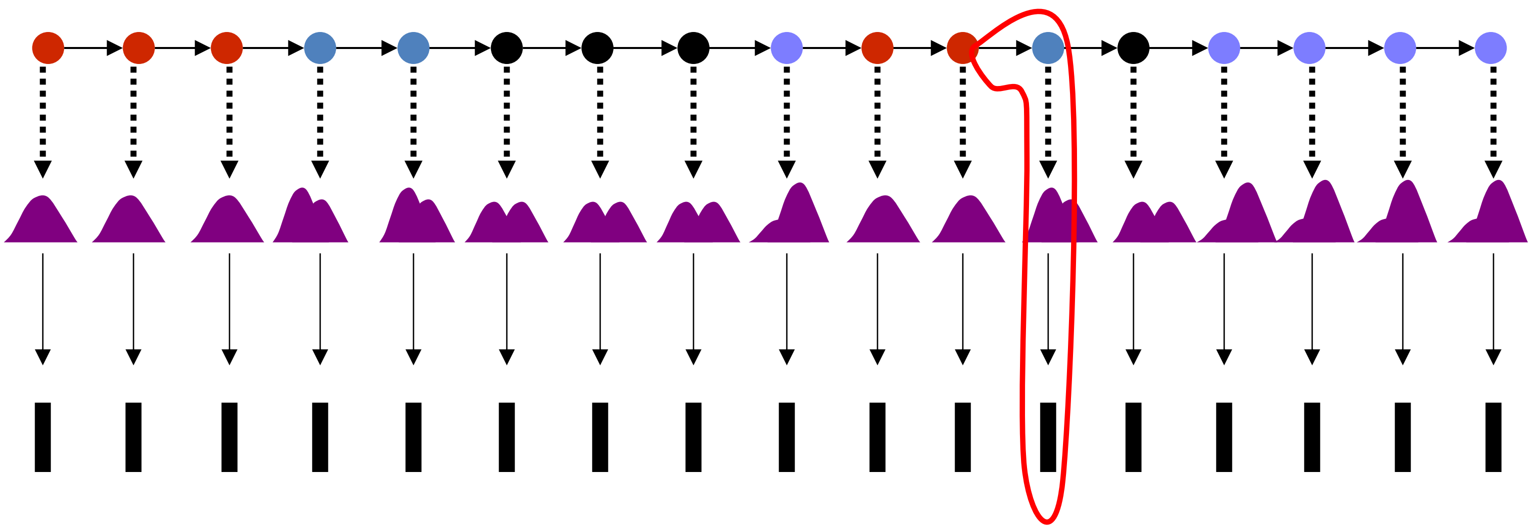
HMM assumed to be generating data



state sequence

state distributions

observation sequence



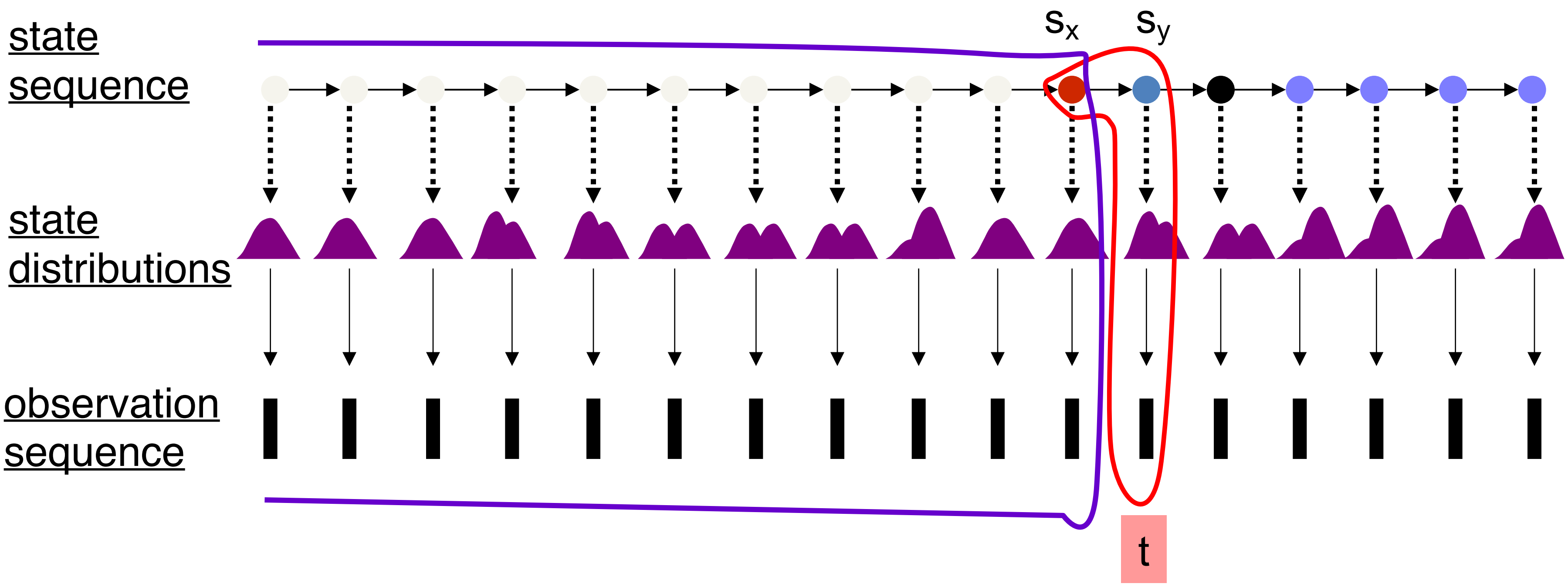
- Each enclosed term represents one forward transition and a subsequent emission

The state sequence

- The probability of a state sequence ?,?,?,?,s_x,s_y ending at time t , and producing all observations until o_t
  - $P(o_{1..t-1}, ?,?,?,?, s_x , o_t, s_y) = P(\underline{o_{1..t-1},?,?,?,?, s_x}) P(o_t|s_y)P(s_y|s_x)$

- The *best* state sequence that ends with s_x,s_y at t will have a probability equal to the probability of the best state sequence ending at t-1 at s_x times $P(o_t|s_y)P(s_y|s_x)$

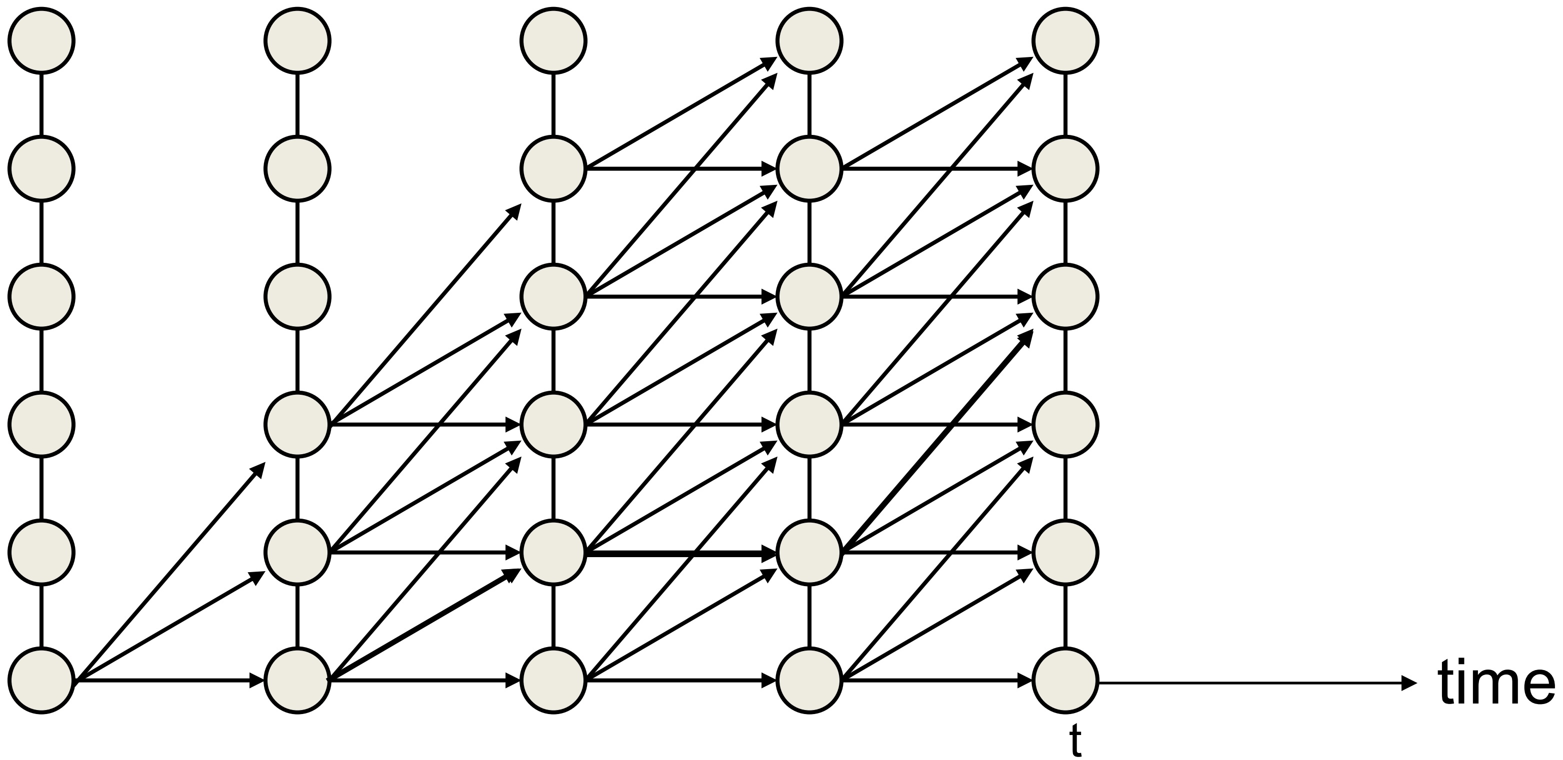
Extending the state sequence



- The probability of a state sequence ?,?,?,?,s_x,s_y ending at time *t* and producing observations until o_t
  - $P(o_{1..t-1}, o_t, ?,?,?,?, s_x, s_y) = P(o_{1..t-1},?,?,?,?, s_x)P(o_t|s_y)P(s_y|s_x)$

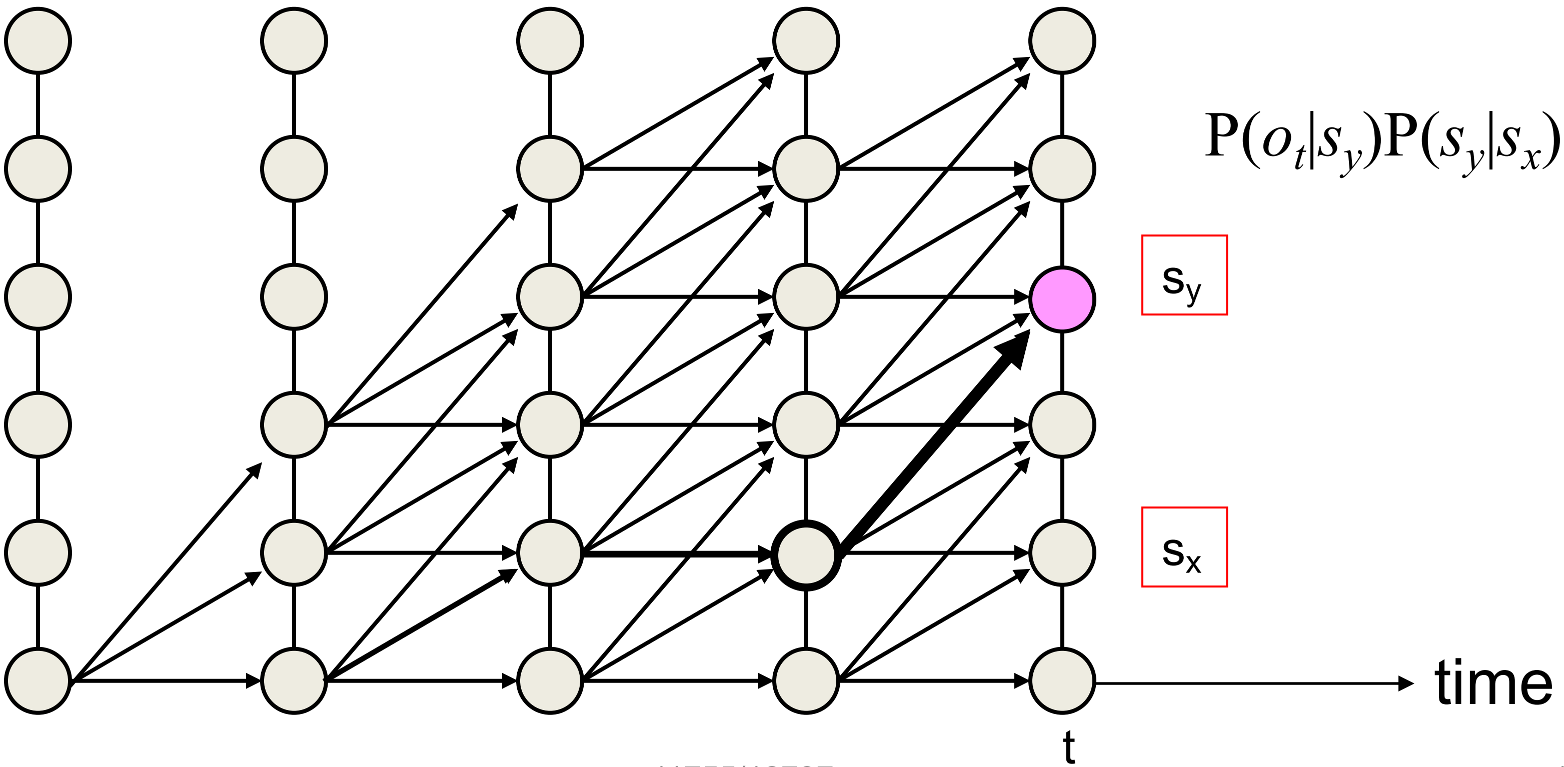
Trellis

- The graph below shows the set of all possible state sequences through this HMM in five time instants



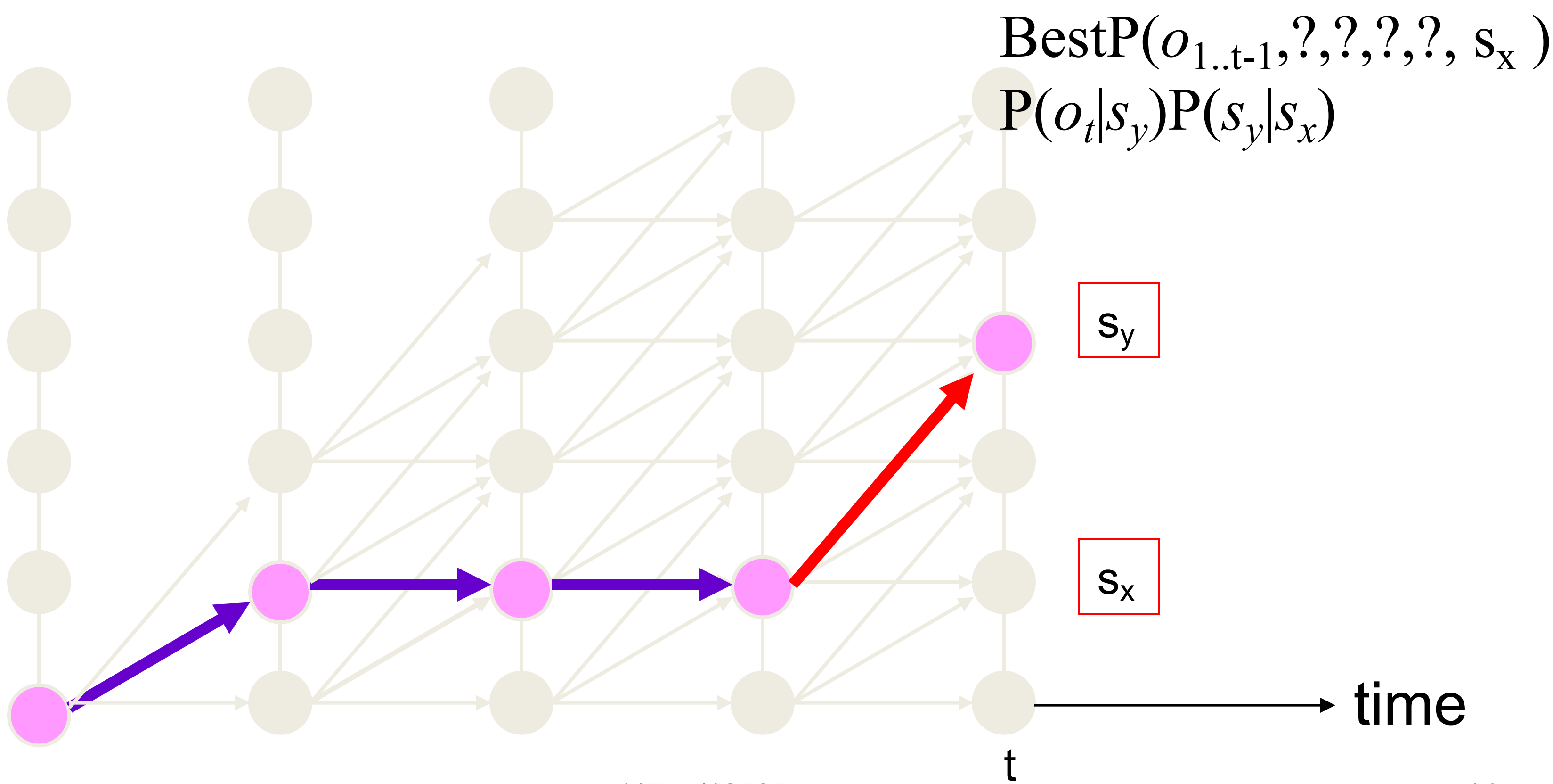
The cost of extending a state sequence

- The cost of *extending* a state sequence ending at s_x is only dependent on the transition from s_x to s_y, and the observation probability at s_y



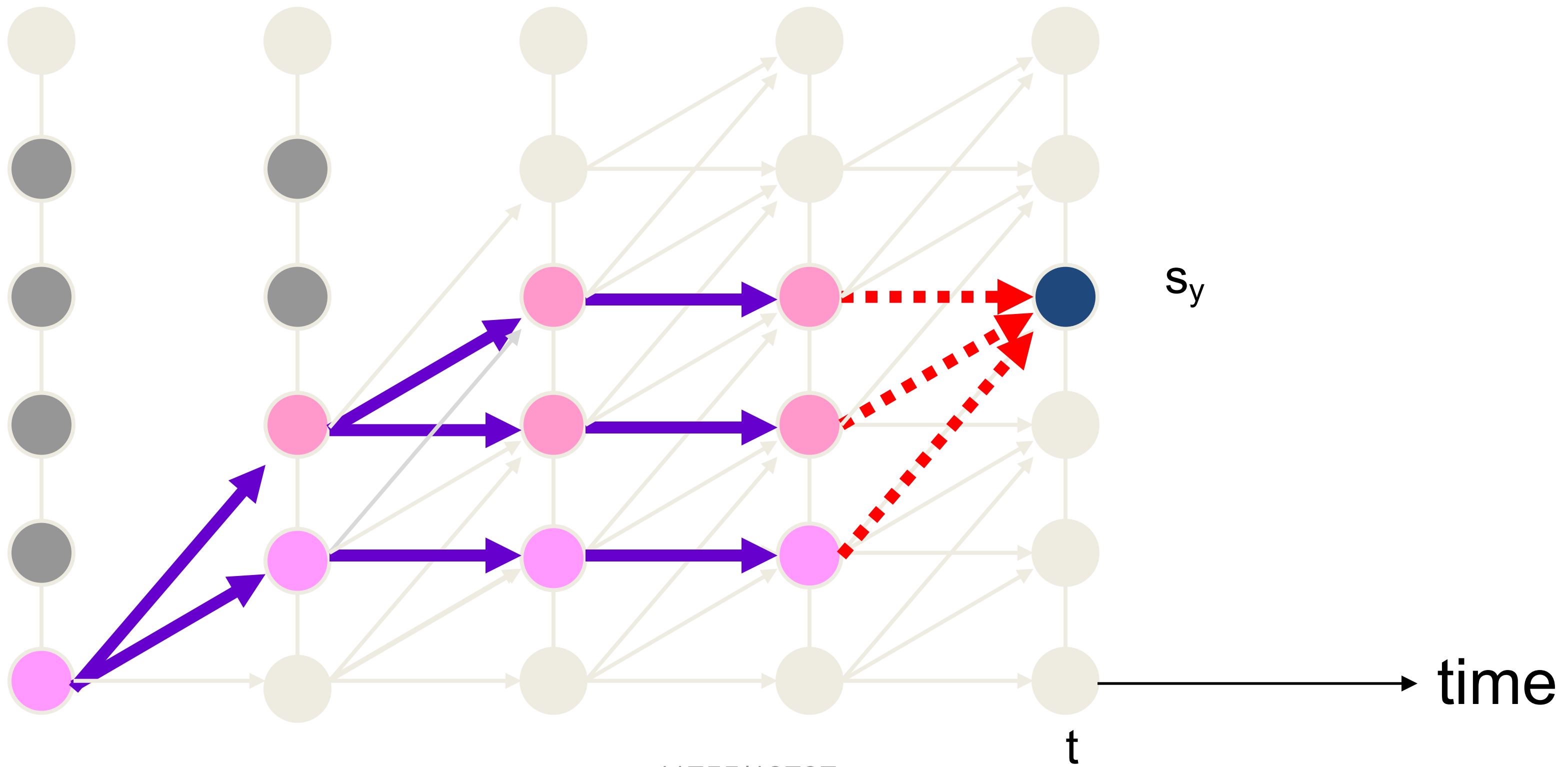
The cost of extending a state sequence

- The best path to s_y through s_x is simply an extension of the best path to s_x



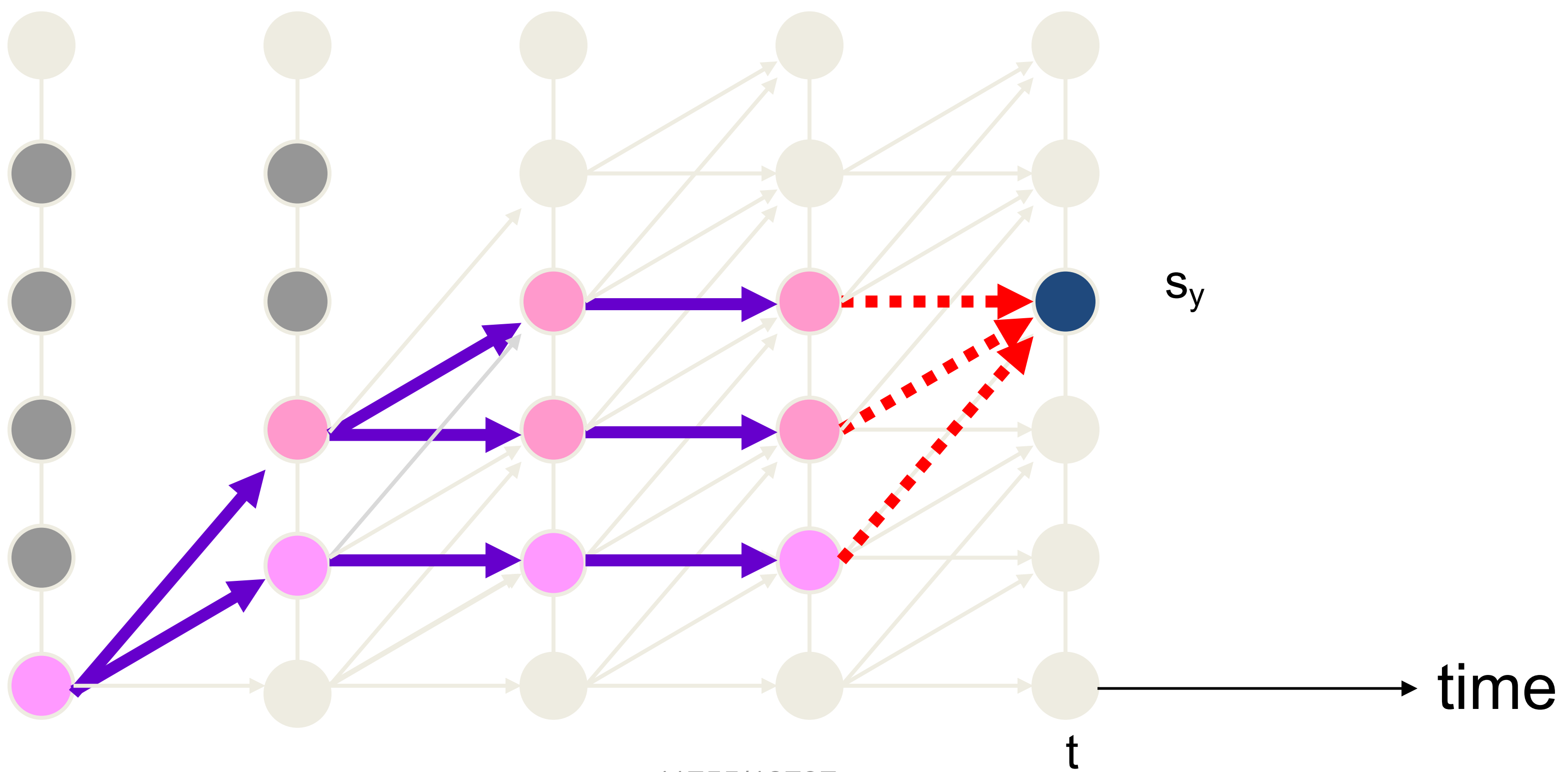
The Recursion

- The overall best path to s_y is an extension of the best path to one of the states at the previous time



The Recursion

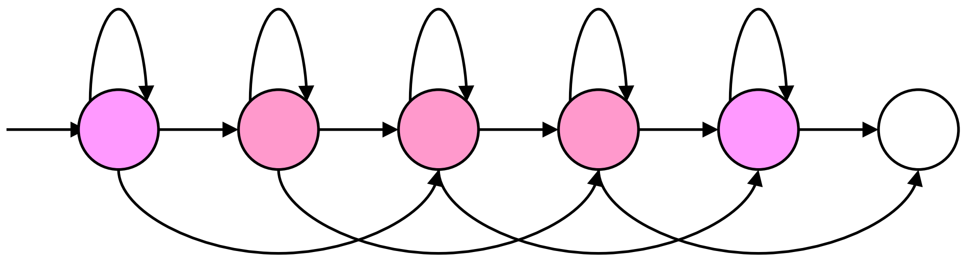
- Prob. of best path to s_y =
  Max_{s_x} $\text{BestP}(o_{1..t-1},?,?,?,?,\ s_x\ )\ P(o_t|s_y)P(s_y|s_x)$

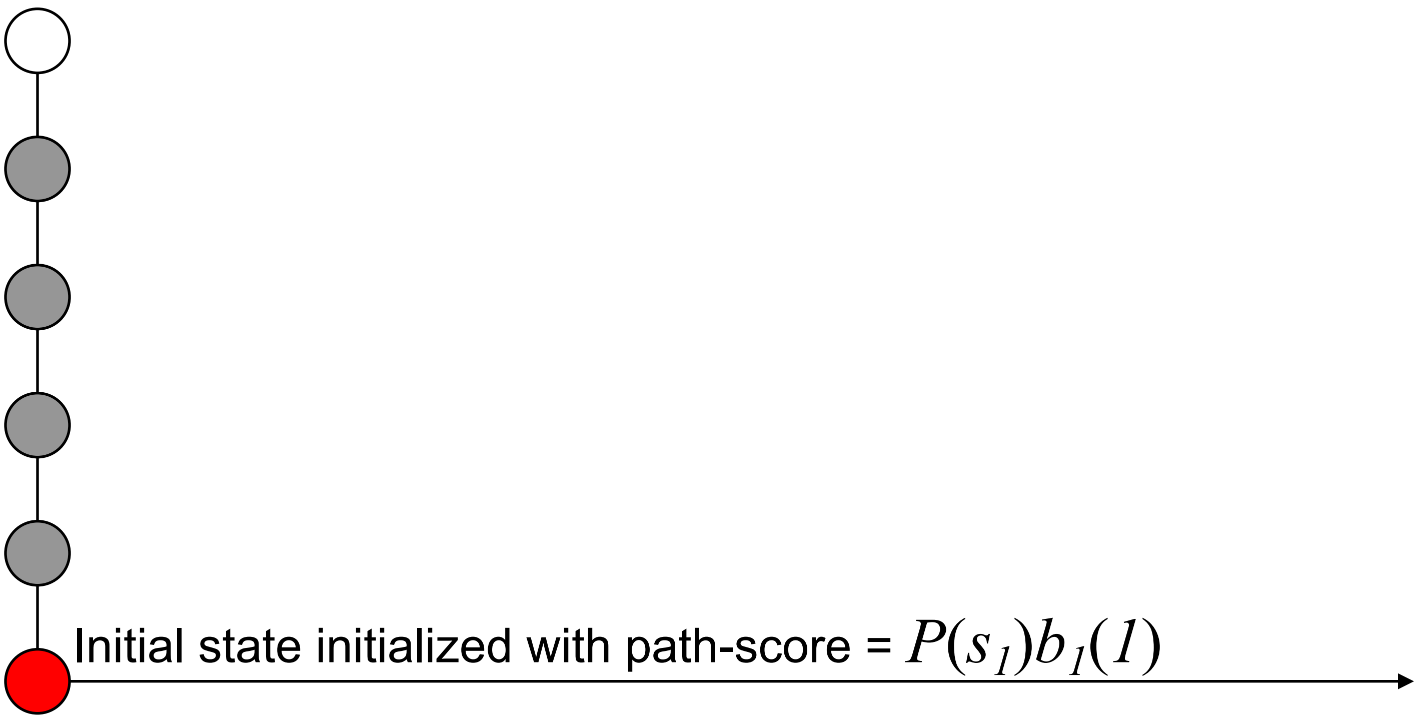


Finding the best state sequence

- The simple algorithm just presented is called the VITERBI algorithm in the literature
  - After A.J.Viterbi, who invented this dynamic programming algorithm for a completely different purpose: decoding error correction codes!

Viterbi Search (contd.)



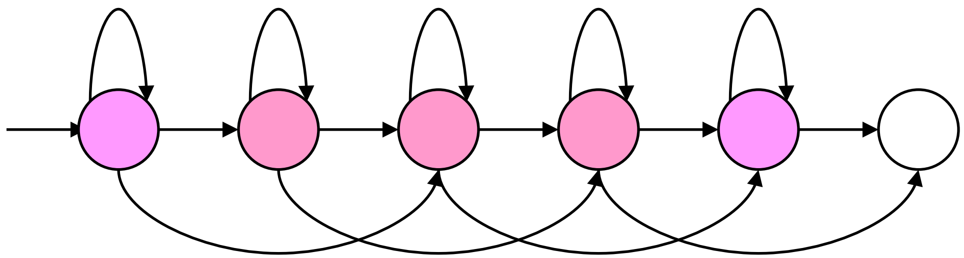


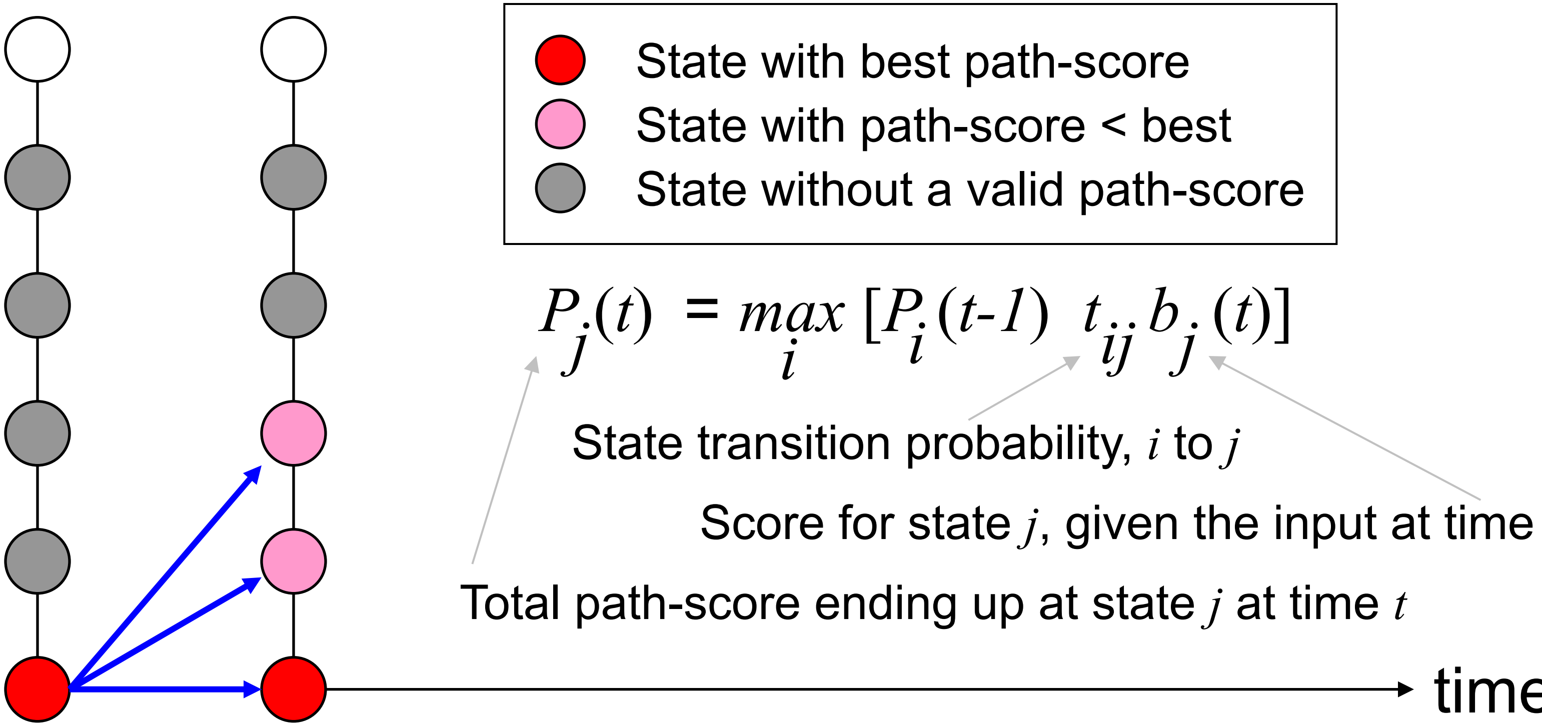
Initial state initialized with path-score = $P(s_1)b_1(1)$

time

In this example all other states have score 0 since $P(s_i) = 0$ for them

Viterbi Search (contd.)





- State with best path-score
- State with path-score < best
- State without a valid path-score

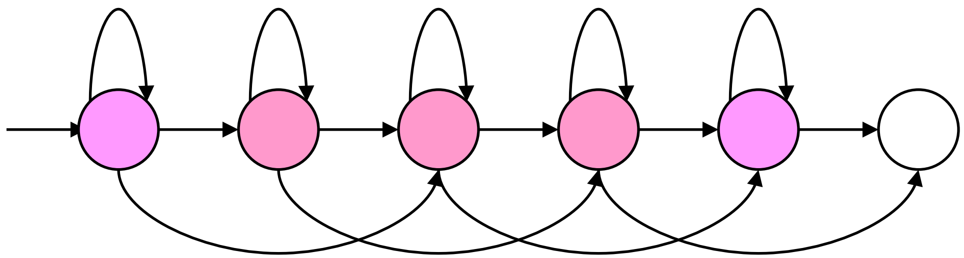
$$P_j(t) = \max_i \left[P_i(t-1)\; t_{ij}\; b_j(t) \right]$$

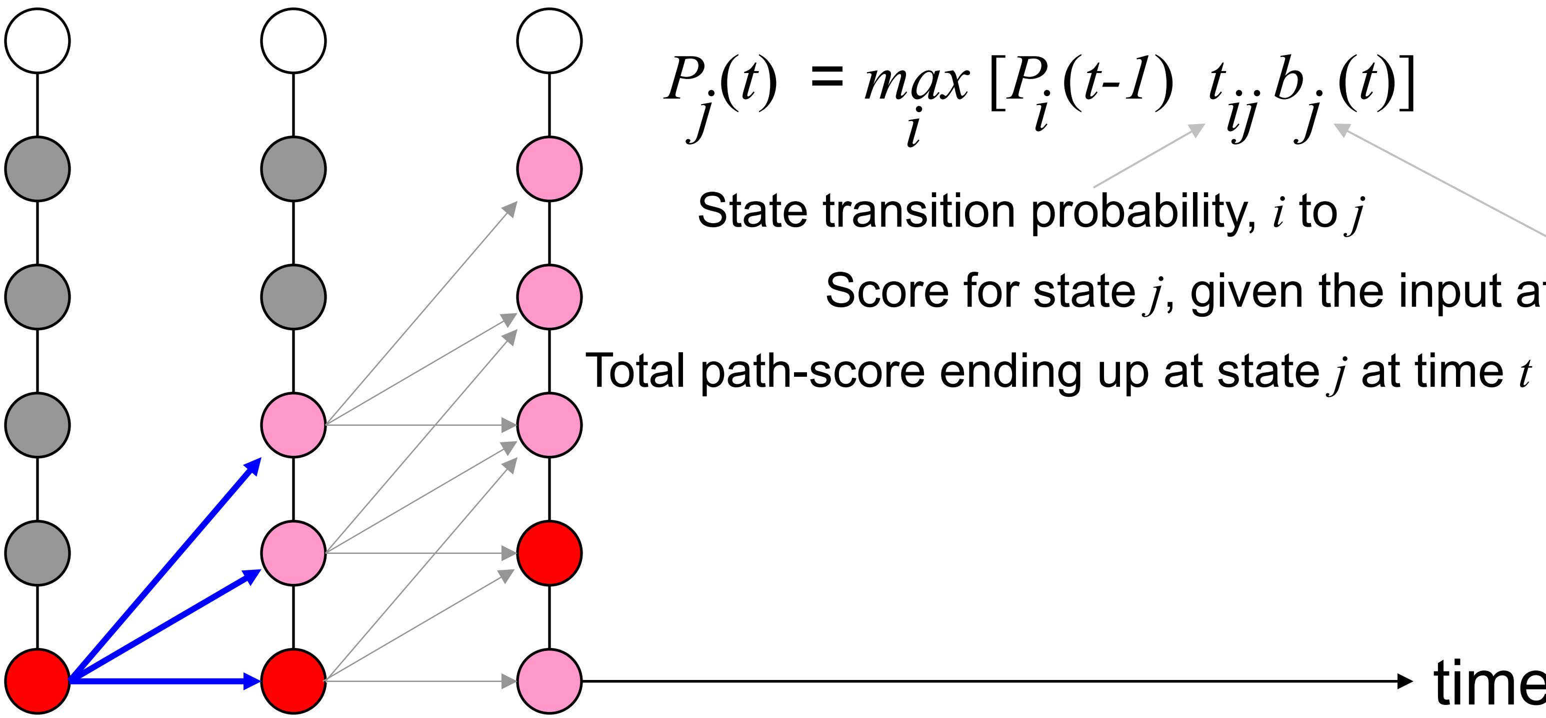
State transition probability, i to j

Score for state j, given the input at time t

Total path-score ending up at state j at time t

Viterbi Search (contd.)





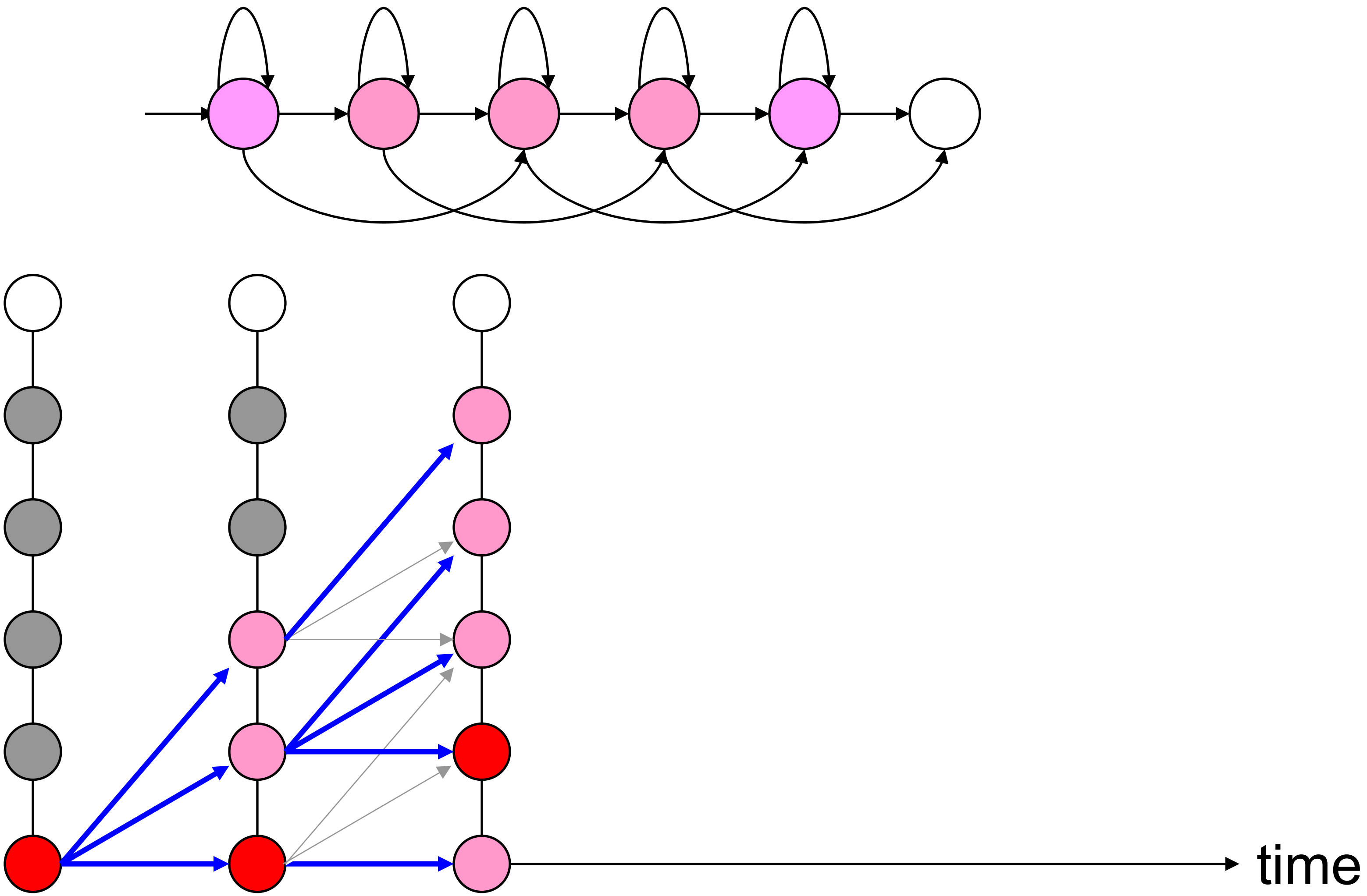
$$P_j(t) = \max_i \left[P_i(t\text{-}1)\ t_{ij}\ b_j(t)\right]$$

State transition probability, i to j

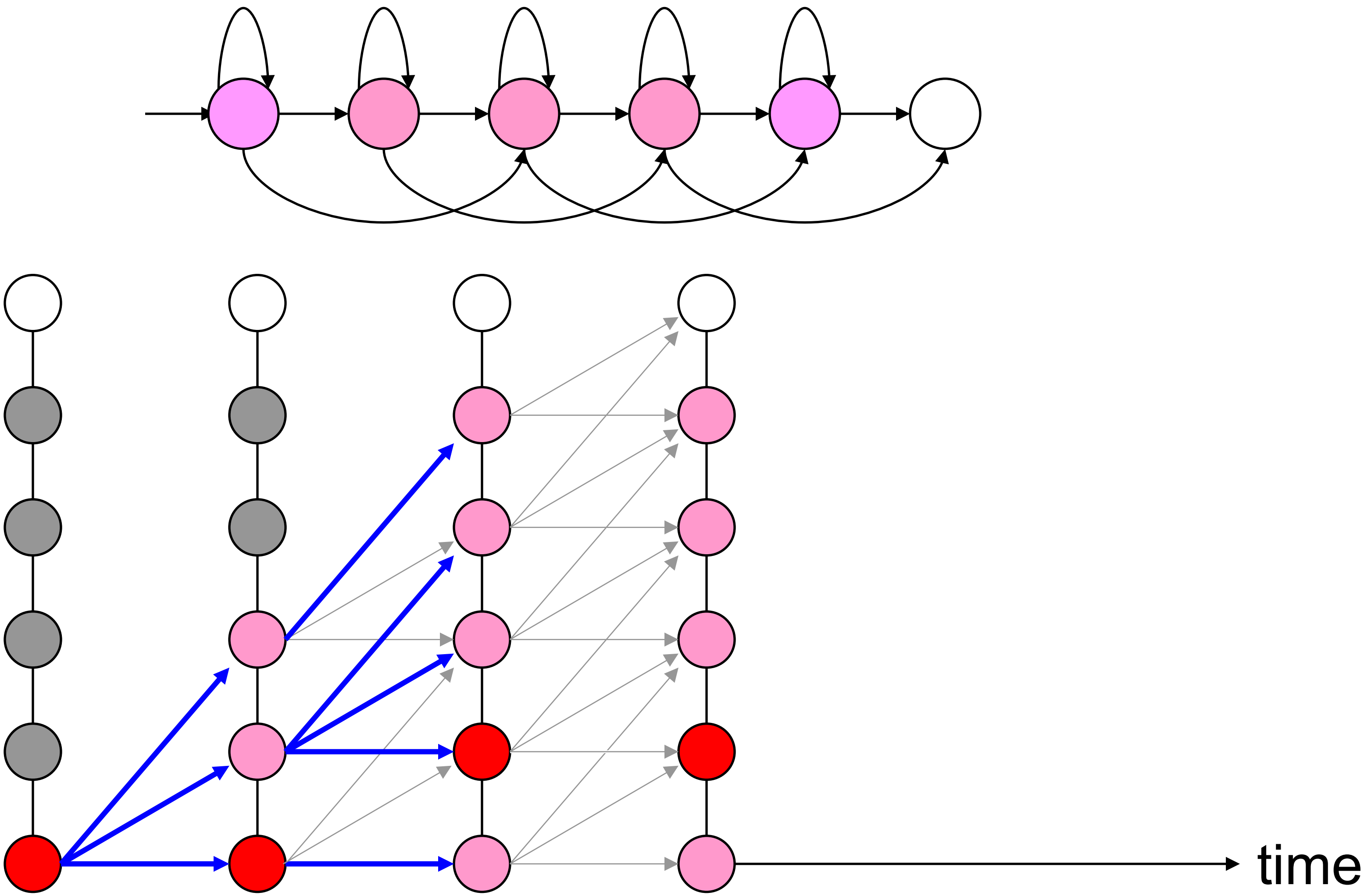
Score for state j, given the input at time t

Total path-score ending up at state j at time t

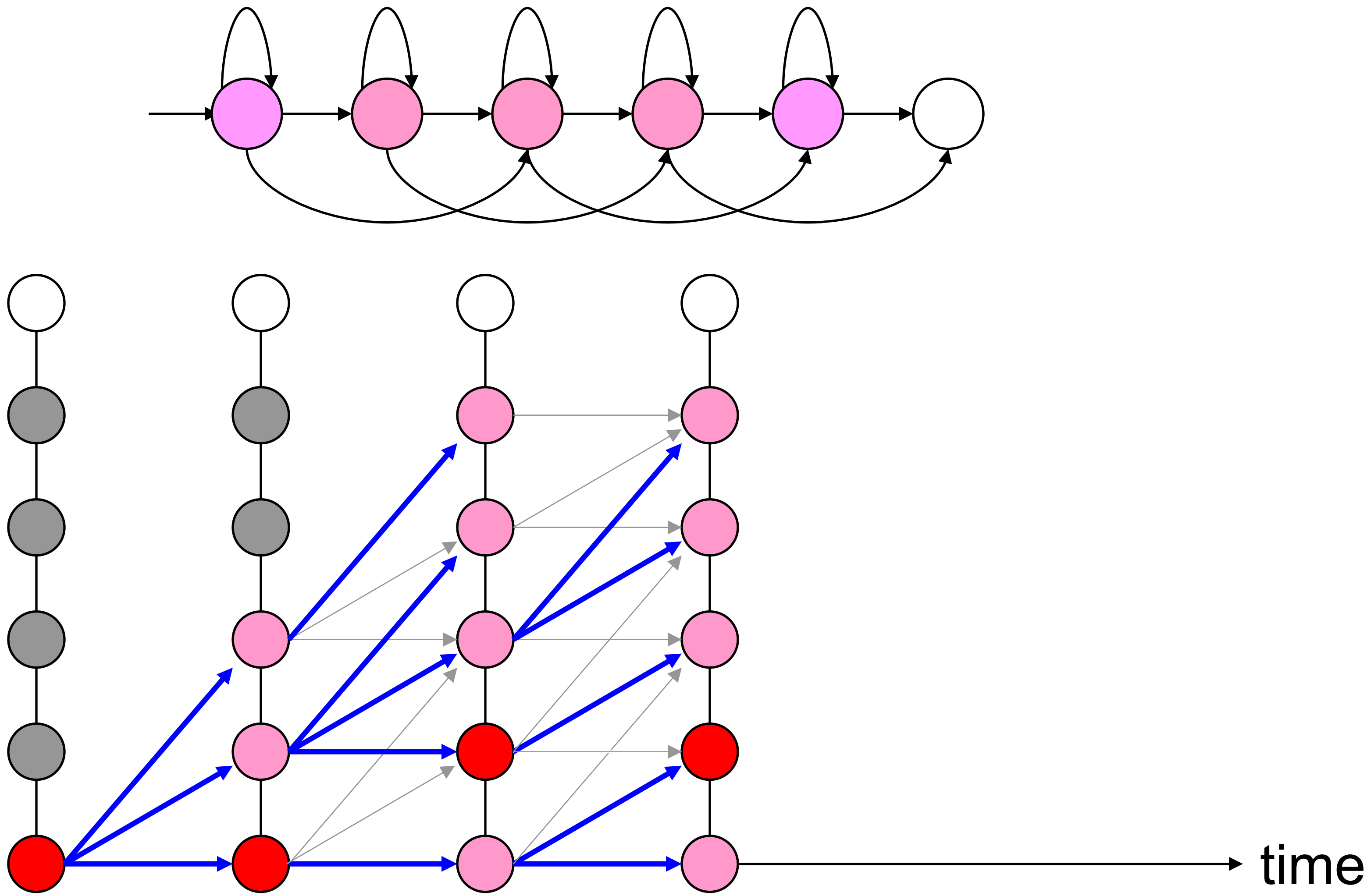
Viterbi Search (contd.)



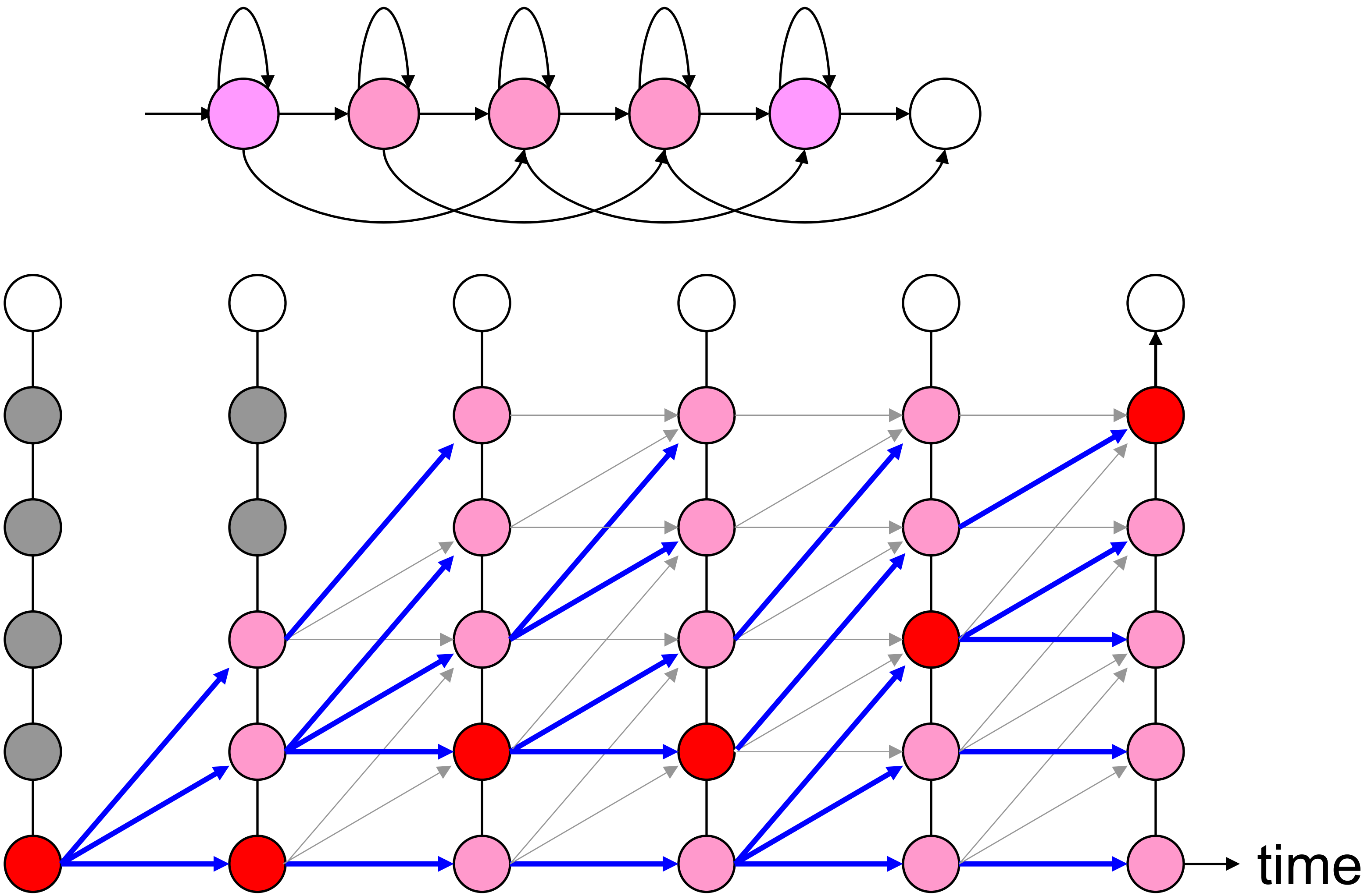
Viterbi Search (contd.)



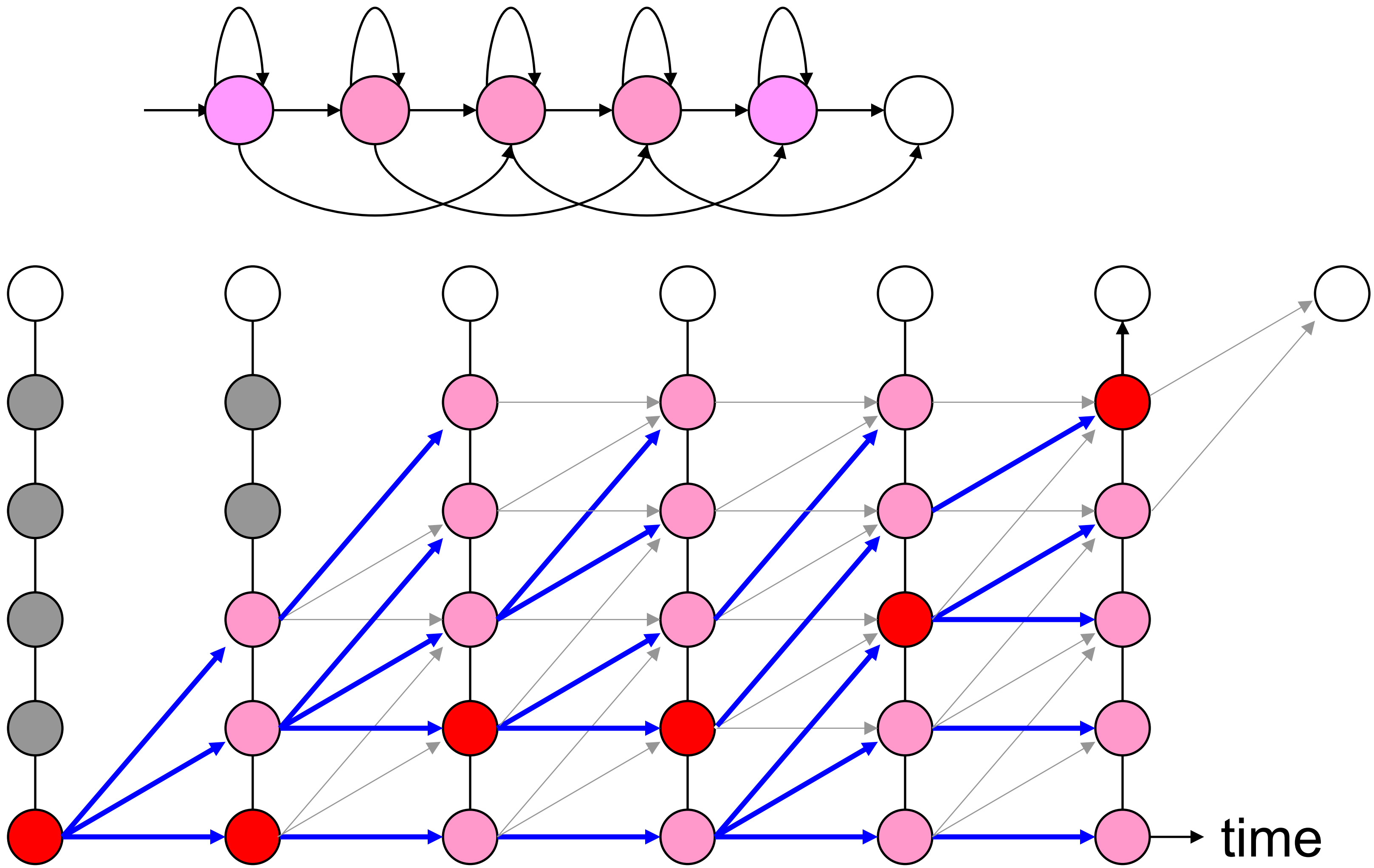
Viterbi Search (contd.)



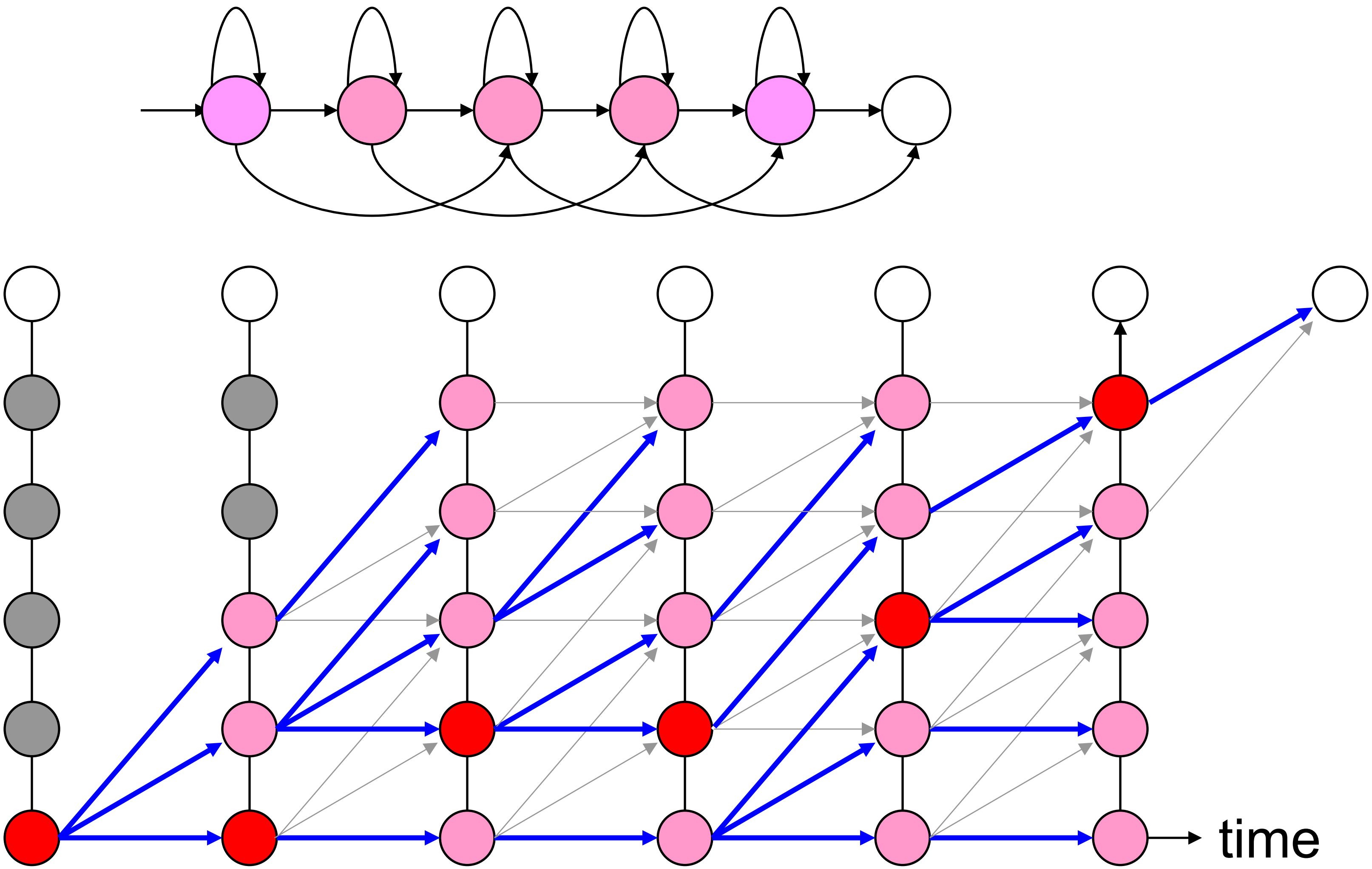
Viterbi Search (contd.)



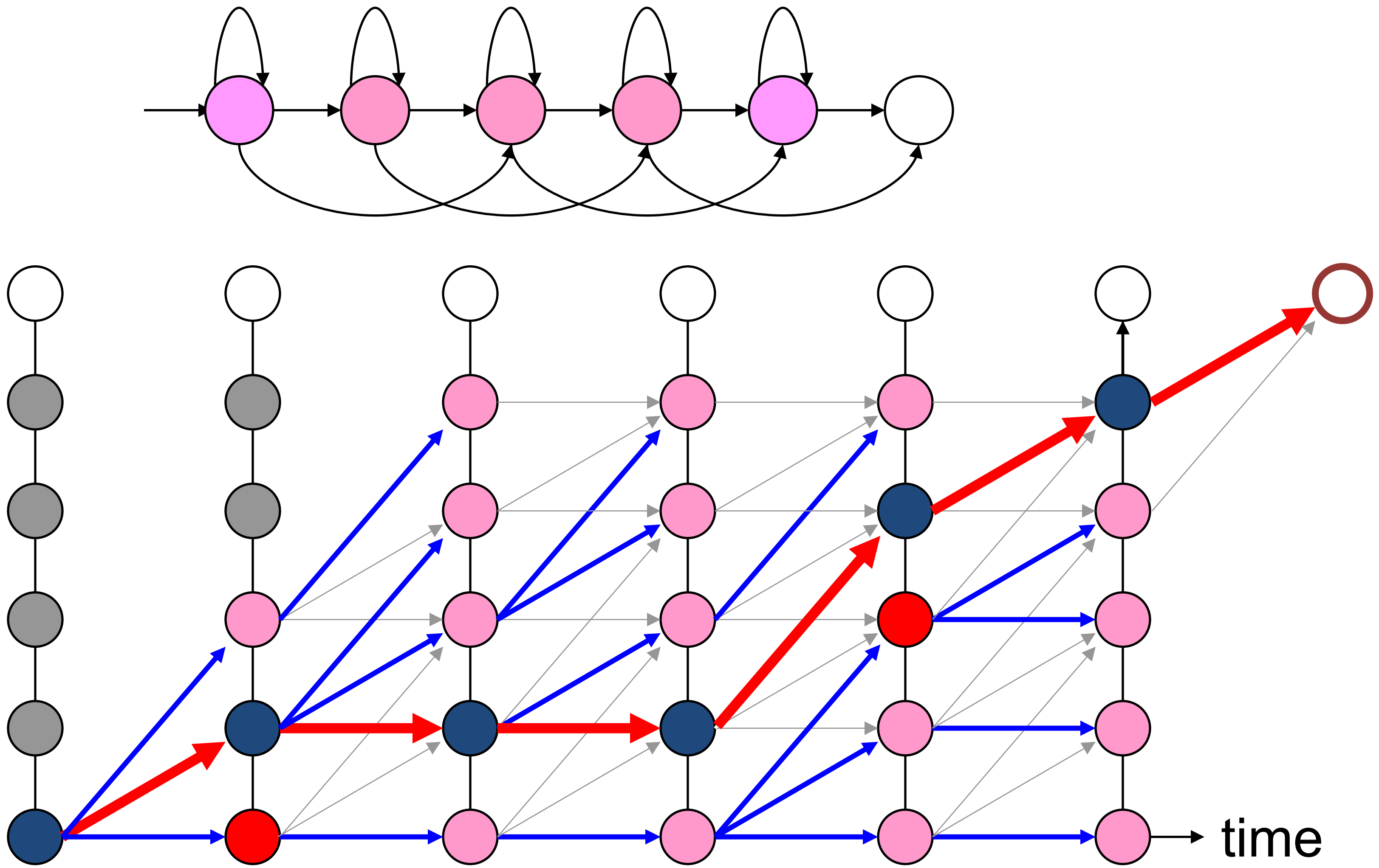
Viterbi Search (contd.)



Viterbi Search (contd.)

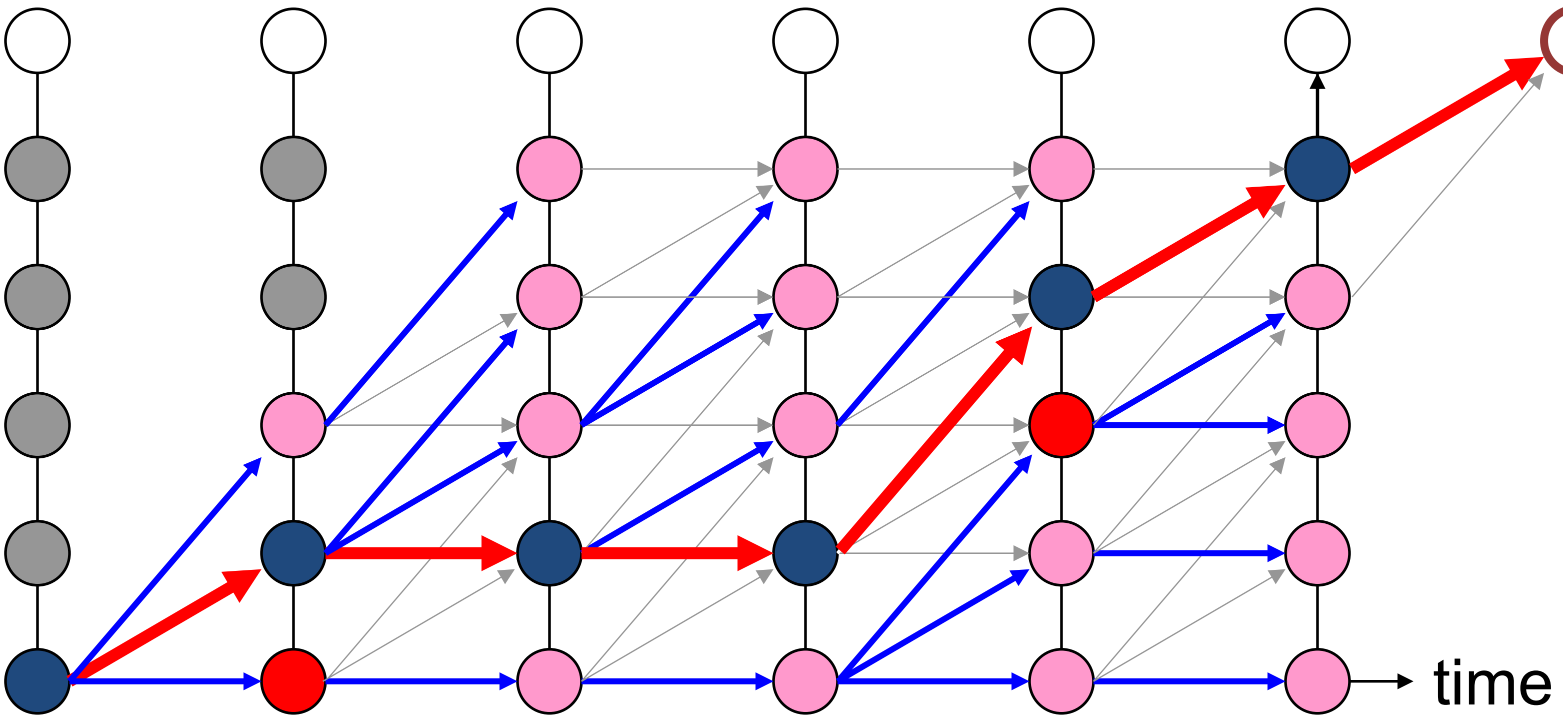


Viterbi Search (contd.)



Viterbi Search (contd.)

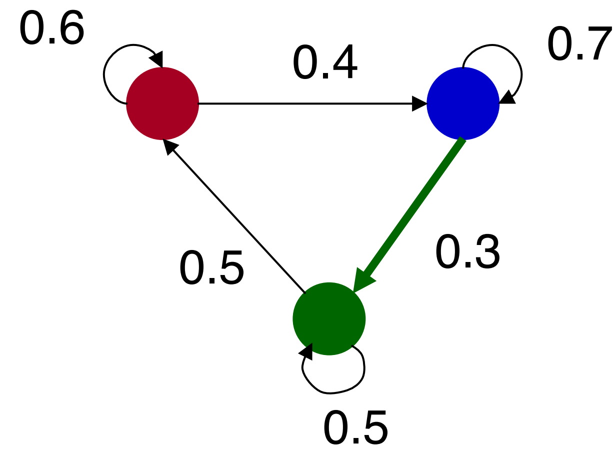
THE BEST STATE SEQUENCE IS THE ESTIMATE OF THE STATE SEQUENCE FOLLOWED IN GENERATING THE OBSERVATION



Problem3: Training HMM parameters

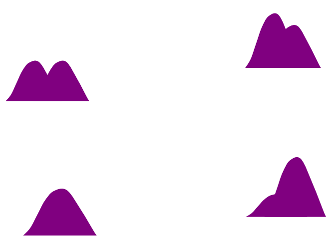
- We can compute the probability of an observation, and the best state sequence given an observation, using the HMM's parameters
- But where do the HMM parameters come from?
- They must be learned from a collection of observation sequences

HMM Parameters



- The transition probabilities
  - Often represented as a matrix as here
  - T_{ij} is the probability that when in state i, the process will move to j
- The probability π_i of beginning at any state s_i
  - The complete set is represented as π
- The *state output distributions*
  - Typically histograms, Gaussians, or Gaussian mixtures
  - Assuming Gaussian
    - Parameters are mean and variance

$$T = \begin{pmatrix} .6 & .4 & 0 \\ 0 & .7 & .3 \\ .5 & 0 & .5 \end{pmatrix}$$



Learning HMM parameters: Simple procedure – counting

- Given a set of training instances
- Iteratively:

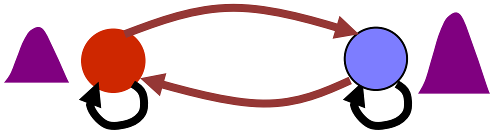
1. Initialize HMM parameters
2. Segment all training instances
3. Estimate transition probabilities and state output probability parameters by counting

Learning by counting example

- Explanation by example in next few slides
- 2-state HMM, Gaussian PDF at states, 3 observation sequences
- Example shows ONE iteration
  - How to count after state sequences are obtained

Example: Learning HMM Parameters

- We have an HMM with two states s1 and s2.
- Observations are vectors x_{ij}
  - i-th sequence, j-th vector
- We are given the following three observation sequences
  - And have already estimated state sequences



Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	X_{a1}	X_{a2}	X_{a3}	X_{a4}	X_{a5}	X_{a6}	X_{a7}	X_{a8}	X_{a9}	X_{a10}

Observation 2

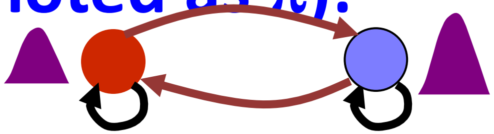
Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	X_{b1}	X_{b2}	X_{b3}	X_{b4}	X_{b5}	X_{b6}	X_{b7}	X_{b8}	X_{b9}

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	X_{c1}	X_{c2}	X_{c3}	X_{c4}	X_{c5}	X_{c6}	X_{c7}	X_{c8}

Example: Learning HMM Parameters

- **Initial state probabilities (usually denoted as π):**
  - We have 3 observations
  - 2 of these begin with S1, and one with S2
  - $\pi(S1) = 2/3$, $\pi(S2) = 1/3$



Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	X_{a1}	X_{a2}	X_{a3}	X_{a4}	X_{a5}	X_{a6}	X_{a7}	X_{a8}	X_{a9}	X_{a10}

Observation 2

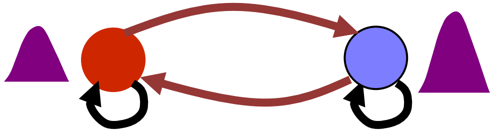
Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	X_{b1}	X_{b2}	X_{b3}	X_{b4}	X_{b5}	X_{b6}	X_{b7}	X_{b8}	X_{b9}

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	X_{c1}	X_{c2}	X_{c3}	X_{c4}	X_{c5}	X_{c6}	X_{c7}	X_{c8}

Example: Learning HMM Parameters

- **Transition probabilities:**
  - State S1 occurs 11 times in non-terminal locations



Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	X_{a1}	X_{a2}	X_{a3}	X_{a4}	X_{a5}	X_{a6}	X_{a7}	X_{a8}	X_{a9}	X_{a10}

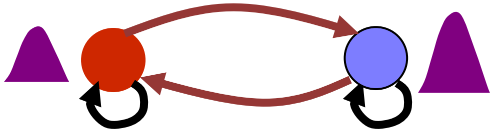
Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	X_{b1}	X_{b2}	X_{b3}	X_{b4}	X_{b5}	X_{b6}	X_{b7}	X_{b8}	X_{b9}

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	X_{c1}	X_{c2}	X_{c3}	X_{c4}	X_{c5}	X_{c6}	X_{c7}	X_{c8}

Example: Learning HMM Parameters



- **Transition probabilities:**
  - State S1 occurs 11 times in non-terminal locations
  - Of these, it is followed immediately by S1 6 times

Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	X_{a1}	X_{a2}	X_{a3}	X_{a4}	X_{a5}	X_{a6}	X_{a7}	X_{a8}	X_{a9}	X_{a10}

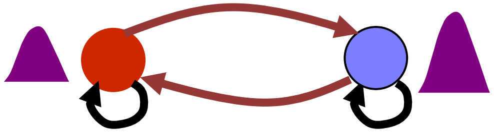
Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	X_{b1}	X_{b2}	X_{b3}	X_{b4}	X_{b5}	X_{b6}	X_{b7}	X_{b8}	X_{b9}

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	X_{c1}	X_{c2}	X_{c3}	X_{c4}	X_{c5}	X_{c6}	X_{c7}	X_{c8}

Example: Learning HMM Parameters



- **Transition probabilities:**
  - State S1 occurs 11 times in non-terminal locations
  - Of these, it is followed immediately by S1 6 times
  - It is followed immediately by S2 5 times

Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	X_{a1}	X_{a2}	X_{a3}	X_{a4}	X_{a5}	X_{a6}	X_{a7}	X_{a8}	X_{a9}	X_{a10}

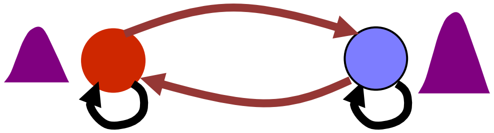
Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	X_{b1}	X_{b2}	X_{b3}	X_{b4}	X_{b5}	X_{b6}	X_{b7}	X_{b8}	X_{b9}

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	X_{c1}	X_{c2}	X_{c3}	X_{c4}	X_{c5}	X_{c6}	X_{c7}	X_{c8}

Example: Learning HMM Parameters



- **Transition probabilities:**
  - State S1 occurs 11 times in non-terminal locations
  - Of these, it is followed immediately by S1 6 times
  - It is followed immediately by S2 5 times
  - P(S1 | **S1**) = 6/ 11;   P(S2 | **S1**) = 5 / 11

Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	X_{a1}	X_{a2}	X_{a3}	X_{a4}	X_{a5}	X_{a6}	X_{a7}	X_{a8}	X_{a9}	X_{a10}

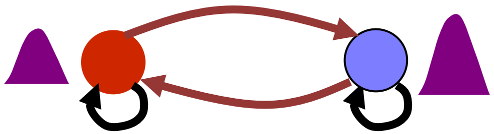
Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	X_{b1}	X_{b2}	X_{b3}	X_{b4}	X_{b5}	X_{b6}	X_{b7}	X_{b8}	X_{b9}

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	X_{c1}	X_{c2}	X_{c3}	X_{c4}	X_{c5}	X_{c6}	X_{c7}	X_{c8}

Example: Learning HMM Parameters



- **Transition probabilities:**
  - State S2 occurs 13 times in non-terminal locations

Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs.	X_{a1}	X_{a2}	X_{a3}	X_{a4}	X_{a5}	X_{a6}	X_{a7}	X_{a8}	X_{a9}	X_{a10}

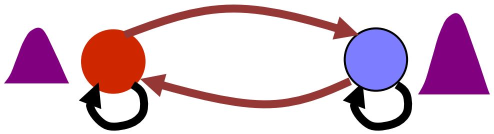
Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	X_{b1}	X_{b2}	X_{b3}	X_{b4}	X_{b5}	X_{b6}	X_{b7}	X_{b8}	X_{b9}

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	X_{c1}	X_{c2}	X_{c3}	X_{c4}	X_{c5}	X_{c6}	X_{c7}	X_{c8}

Example: Learning HMM Parameters



- **Transition probabilities:**
  - State S2 occurs 13 times in non-terminal locations
  - Of these, it is followed immediately by S1 5 times

Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	X_{a1}	X_{a2}	X_{a3}	X_{a4}	X_{a5}	X_{a6}	X_{a7}	X_{a8}	X_{a9}	X_{a10}

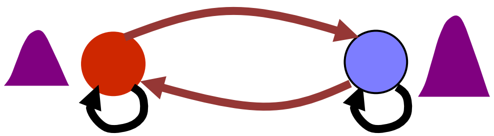
Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	X_{b1}	X_{b2}	X_{b3}	X_{b4}	X_{b5}	X_{b6}	X_{b7}	X_{b8}	X_{b9}

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	X_{c1}	X_{c2}	X_{c3}	X_{c4}	X_{c5}	X_{c6}	X_{c7}	X_{c8}

Example: Learning HMM Parameters



- **Transition probabilities:**
  - State S2 occurs 13 times in non-terminal locations
  - Of these, it is followed immediately by S1 5 times
  - It is followed immediately by S2 8 times

Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	X_{a1}	X_{a2}	X_{a3}	X_{a4}	X_{a5}	X_{a6}	X_{a7}	X_{a8}	X_{a9}	X_{a10}

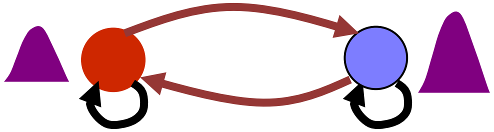
Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	X_{b1}	X_{b2}	X_{b3}	X_{b4}	X_{b5}	X_{b6}	X_{b7}	X_{b8}	X_{b9}

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	X_{c1}	X_{c2}	X_{c3}	X_{c4}	X_{c5}	X_{c6}	X_{c7}	X_{c8}

Example: Learning HMM Parameters



- **Transition probabilities:**
  - State S2 occurs 13 times in non-terminal locations
  - Of these, it is followed immediately by S1 5 times
  - It is followed immediately by S2 8 times
  - P(S1 | **S2**) = 5 / 13;   P(S2 | **S2**) = 8 / 13

Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	X_{a1}	X_{a2}	X_{a3}	X_{a4}	X_{a5}	X_{a6}	X_{a7}	X_{a8}	X_{a9}	X_{a10}

Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	X_{b1}	X_{b2}	X_{b3}	X_{b4}	X_{b5}	X_{b6}	X_{b7}	X_{b8}	X_{b9}

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	X_{c1}	X_{c2}	X_{c3}	X_{c4}	X_{c5}	X_{c6}	X_{c7}	X_{c8}

Parameters learnt so far

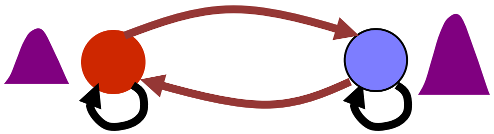
- State initial probabilities, often denoted as π
  - $\pi(S1) = 2/3 = 0.66$
  - $\pi(S2) = 1/3 = 0.33$
- State transition probabilities
  - $P(S1 \mid S1) = 6/11 = 0.545$; $P(S2 \mid S1) = 5/11 = 0.455$
  - $P(S1 \mid S2) = 5/13 = 0.385$; $P(S2 \mid S2) = 8/13 = 0.615$
  - Represented as a transition matrix

$$A = \begin{pmatrix} P(S1 \mid S1) & P(S2 \mid S1) \\ P(S1 \mid S2) & P(S2 \mid S2) \end{pmatrix} = \begin{pmatrix} 0.545 & 0.455 \\ 0.385 & 0.615 \end{pmatrix}$$

Each row of this matrix must sum to 1.0

Example: Learning HMM Parameters

- State output probability for S1
  - There are 13 observations in S1



Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	X_{a1}	X_{a2}	X_{a3}	X_{a4}	X_{a5}	X_{a6}	X_{a7}	X_{a8}	X_{a9}	X_{a10}

Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	X_{b1}	X_{b2}	X_{b3}	X_{b4}	X_{b5}	X_{b6}	X_{b7}	X_{b8}	X_{b9}

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	X_{c1}	X_{c2}	X_{c3}	X_{c4}	X_{c5}	X_{c6}	X_{c7}	X_{c8}

Example: Learning HMM Parameters

- State output probability for S1
  - There are 13 observations in S1
  - Segregate them out and count
    - Compute parameters (mean and variance) of Gaussian output density for state S1

Time	1	2	6	7	9	10
state	S1	S1	S1	S1	S1	S1
Obs	X_{a1}	X_{a2}	X_{a6}	X_{a7}	X_{a9}	X_{a10}

$$P(X \mid S_1) = \frac{1}{\sqrt{(2\pi)^d \, |\Theta_1|}} \exp\left(-0.5 (X-\mu_1)^T \Theta_1^{-1} (X-\mu_1)\right)$$

Time	3	4	9
state	S1	S1	S1
Obs	X_{b3}	X_{b4}	X_{b9}

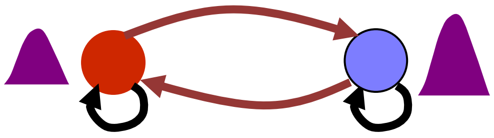
$$\mu_1 = \frac{1}{13}\left(\begin{array}{c} X_{a1} + X_{a2} + X_{a6} + X_{a7} + X_{a9} + X_{a10} + X_{b3} + \\ X_{b4} + X_{b9} + X_{c1} + X_{c2} + X_{c4} + X_{c5} \end{array}\right)$$

Time	1	3	4	5
state	S1	S1	S1	S1
Obs	X_{c1}	X_{c2}	X_{c4}	X_{c5}

$$\Theta_1 = \frac{1}{13}\left(\begin{array}{c} (X_{a1}-\mu_1)(X_{a1}-\mu_1)^T + (X_{a2}-\mu_1)(X_{a2}-\mu_1)^T + \ldots \\ (X_{b3}-\mu_1)(X_{b3}-\mu_1)^T + (X_{b4}-\mu_1)(X_{b4}-\mu_1)^T + \ldots \\ (X_{c1}-\mu_1)(X_{c1}-\mu_1)^T + (X_{c2}-\mu_1)(X_{c2}-\mu_1)^T + \ldots \end{array}\right)$$

Example: Learning HMM Parameters

- State output probability for S2
  - There are 14 observations in S2



Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	X_{a1}	X_{a2}	X_{a3}	X_{a4}	X_{a5}	X_{a6}	X_{a7}	X_{a8}	X_{a9}	X_{a10}

Observation 2

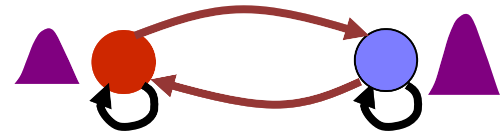
Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	X_{b1}	X_{b2}	X_{b3}	X_{b4}	X_{b5}	X_{b6}	X_{b7}	X_{b8}	X_{b9}

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	X_{c1}	X_{c2}	X_{c3}	X_{c4}	X_{c5}	X_{c6}	X_{c7}	X_{c8}

Example: Learning HMM Parameters

- State output probability for S2
  - There are 14 observations in S2
  - Segregate them out and count
    - Compute parameters (mean and variance) of Gaussian output density for state S2



Time	3	4	5	8
state	S2	S2	S2	S2
Obs	X_{a3}	X_{a4}	X_{a5}	X_{a8}

$$P(X \mid S_2) = \frac{1}{\sqrt{(2\pi)^d \,|\Theta_2|}} \exp\left(-0.5(X-\mu_2)^T \Theta_2^{-1}(X-\mu_2)\right)$$

Time	1	2	5	6	7	8
state	S2	S2	S2	S2	S2	S2
Obs	X_{b1}	X_{b2}	X_{b5}	X_{b6}	X_{b7}	X_{b8}

$$\mu_2 = \frac{1}{14}\begin{pmatrix} X_{a3} + X_{a4} + X_{a5} + X_{a8} + X_{b1} + X_{b2} + X_{b5} + \\ X_{b6} + X_{b7} + X_{b8} + X_{c2} + X_{c6} + X_{c7} + X_{c8} \end{pmatrix}$$

Time	2	6	7	8
state	S2	S2	S2	S2
Obs	X_{c2}	X_{c6}	X_{c7}	X_{c8}

$$\Theta_1 = \frac{1}{14}\left((X_{a3}-\mu_2)(X_{a3}-\mu_2)^T + \ldots\right)$$

We have learnt all the HMM parmeters

- State initial probabilities, often denoted as π
  - $\pi(S1) = 0.66$ $\pi(S2) = 1/3 = 0.33$
- State transition probabilities

$$A = \begin{pmatrix} 0.545 & 0.455 \\ 0.385 & 0.615 \end{pmatrix}$$

- State output probabilities

State output probability for S1

$$P(X \mid S_1) = \frac{1}{\sqrt{(2\pi)^d \mid \Theta_1 \mid}} \exp\left(-0.5(X-\mu_1)^T \Theta_1^{-1} (X-\mu_1)\right)$$

State output probability for S2

$$P(X \mid S_2) = \frac{1}{\sqrt{(2\pi)^d \mid \Theta_2 \mid}} \exp\left(-0.5(X-\mu_2)^T \Theta_2^{-1} (X-\mu_2)\right)$$

Update rules at each iteration

$$\pi(s_i) = \frac{\sum_{obs} \delta_{t=1}(s_i)}{N_{obs}}$$

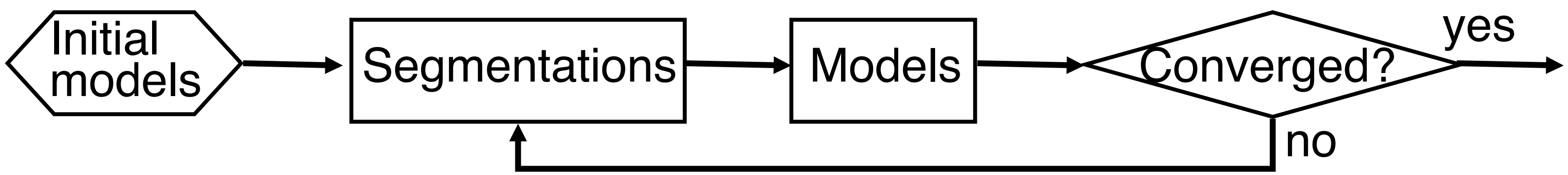
$$P(s_j|s_i) = \frac{\sum_{obs} \sum_{t=1}^{T-1} \delta_{t,t+1}(s_j|s_i)}{\sum_{obs} \sum_{t=1}^{T-1} \delta_t(s_i)}$$

$$\mu_i = \frac{\sum_{obs} \sum_{t=1}^{T} \delta_t(s_i)\, X_{obs}(t)}{\sum_{obs} \sum_{t=1}^{T} \delta_t(s_i)}$$

$$\Theta_i = \frac{\sum_{obs} \sum_{t=1}^{T} \delta_t(s_i)\, (X_{obs}(t) - \mu_i)(X_{obs}(t) - \mu_i)^T}{\sum_{obs} \sum_{t=1}^{T} \delta_t(s_i)}$$

- Assumes state output PDF = Gaussian
  - For GMMs, estimate GMM parameters from collection of observations at any state

Training by segmentation: Viterbi training



- Initialize all HMM parameters

- Segment all training observation sequences into states using the Viterbi algorithm with the current models

- Using estimated state sequences and training observation sequences, reestimate the HMM parameters

- This method is also called a “segmental k-means” learning procedure

Poll 1

Alternative to counting: SOFT counting

- Expectation maximization
- *Every* observation contributes to every state

Update rules at each iteration

$$\pi(s_i) = \frac{\sum_{obs} \delta_{t=1}(s_i)}{N_{obs}}$$

$$P(s_j|s_i) = \frac{\sum_{obs} \sum_{t=1}^{T-1} \delta_{t,t+1}(s_j|s_i)}{\sum_{obs} \sum_{t=1}^{T-1} \delta_t(s_i)}$$

$$\mu_i = \frac{\sum_{obs} \sum_{t=1}^{T} \delta_t(s_i)\, X_{obs}(t)}{\sum_{obs} \sum_{t=1}^{T} \delta_t(s_i)}$$

$$\Theta_i = \frac{\sum_{obs} \sum_{t=1}^{T} \delta_t(s_i)\, (X_{obs}(t) - \mu_i)(X_{obs}(t) - \mu_i)^T}{\sum_{obs} \sum_{t=1}^{T} \delta_t(s_i)}$$

- Assumes state output PDF = Gaussian
  - For GMMs, estimate GMM parameters from collection of observations at any state

Update rules at each iteration

$$\pi(s_i) = \frac{\sum_{Obs} P(state(t=1) = s_i \mid Obs)}{\text{Total no. of observation sequences}}$$

$$P(s_j \mid s_i) = \frac{\sum_{Obs}\sum_{t} P(state(t) = s_i, state(t+1) = s_j \mid Obs)}{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)}$$

$$\mu_i = \frac{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs) X_{Obs,t}}{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)}$$

$$\Theta_i = \frac{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)(X_{Obs,t} - \mu_i)(X_{Obs,t} - \mu_i)^T}{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)}$$

- Every observation contributes to every state

Poll 2

Update rules at each iteration

$$\pi(s_i) = \frac{\sum_{Obs} P(state(t=1) = s_i \mid Obs)}{\text{Total no. of observation sequences}}$$

$$P(s_j \mid s_i) = \frac{\sum_{Obs}\sum_{t} P(state(t) = s_i, state(t+1) = s_j \mid Obs)}{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)}$$

$$\mu_i = \frac{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs) X_{Obs,t}}{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)}$$

$$\Theta_i = \frac{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)(X_{Obs,t} - \mu_i)(X_{Obs,t} - \mu_i)^T}{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)}$$

- Where did these terms come from?

$$P(state(t) = s \mid Obs)$$

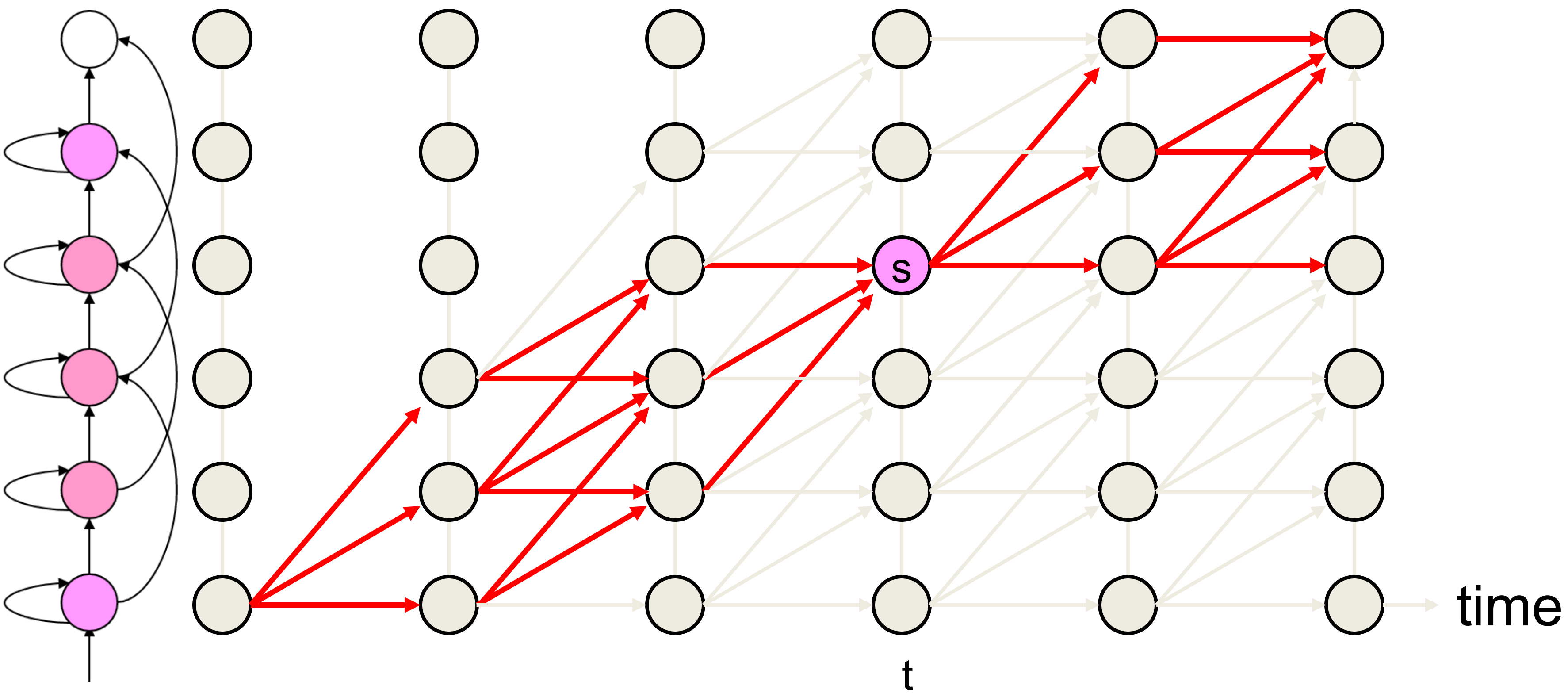
- The probability that the process was at *s* when it generated X_t given the entire observation
  - Dropping the "Obs" subscript for brevity

$$P(state(t) = s \mid X_1, X_2, \ldots, X_T) \propto P(state(t) = s, X_1, X_2, \ldots, X_T)$$

- We will compute $P(state(t) = s_i, x_1, x_2, \ldots, x_T)$ first
  – This is the probability that the process visited *s* at time *t* while producing the entire observation

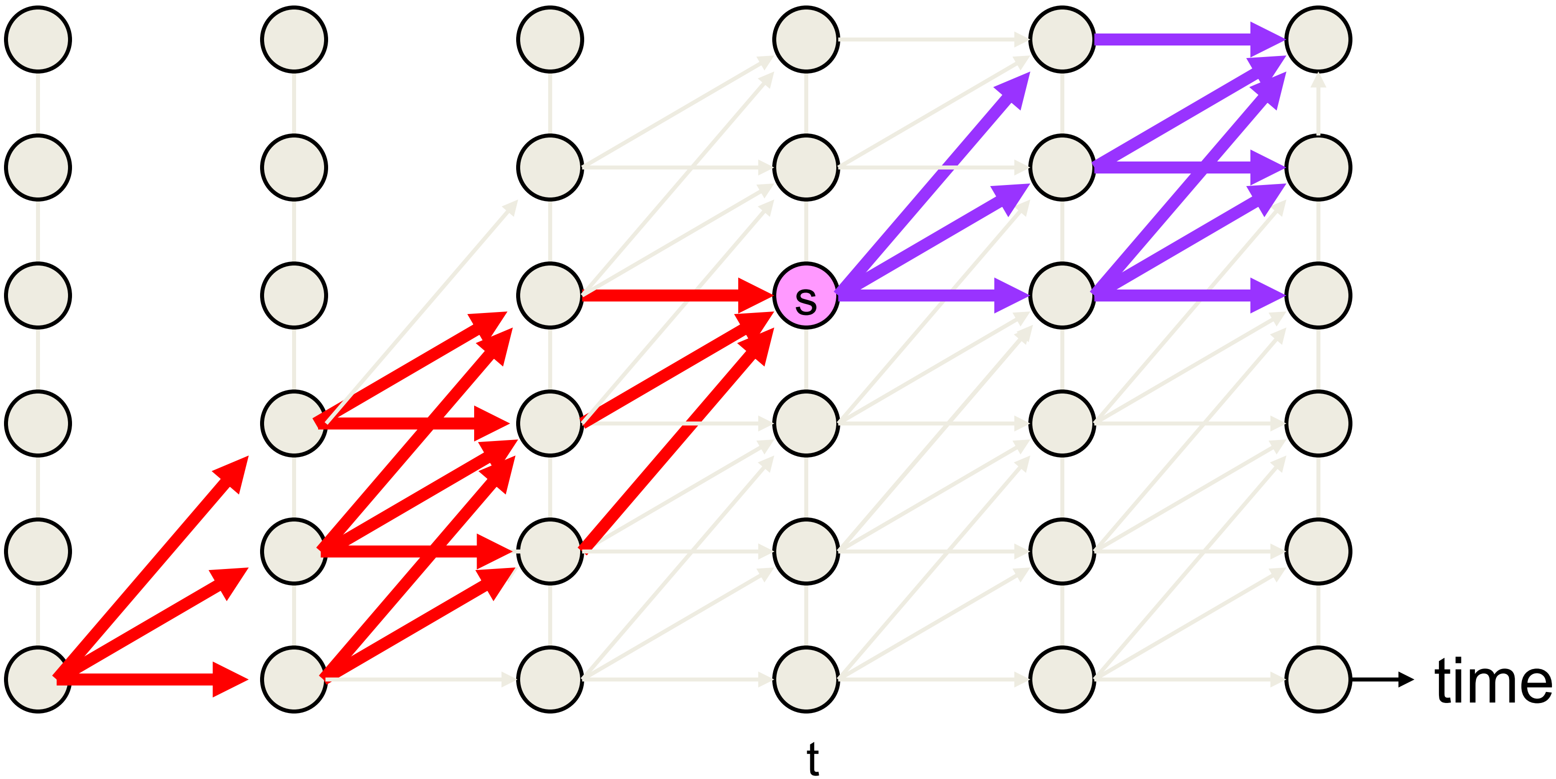
$$P(state(t) = s, x_1, x_2, \ldots, x_T)$$

- The probability that the HMM was in a particular state *s* when generating the observation sequence is the probability that it followed a state sequence that passed through *s* at time *t*



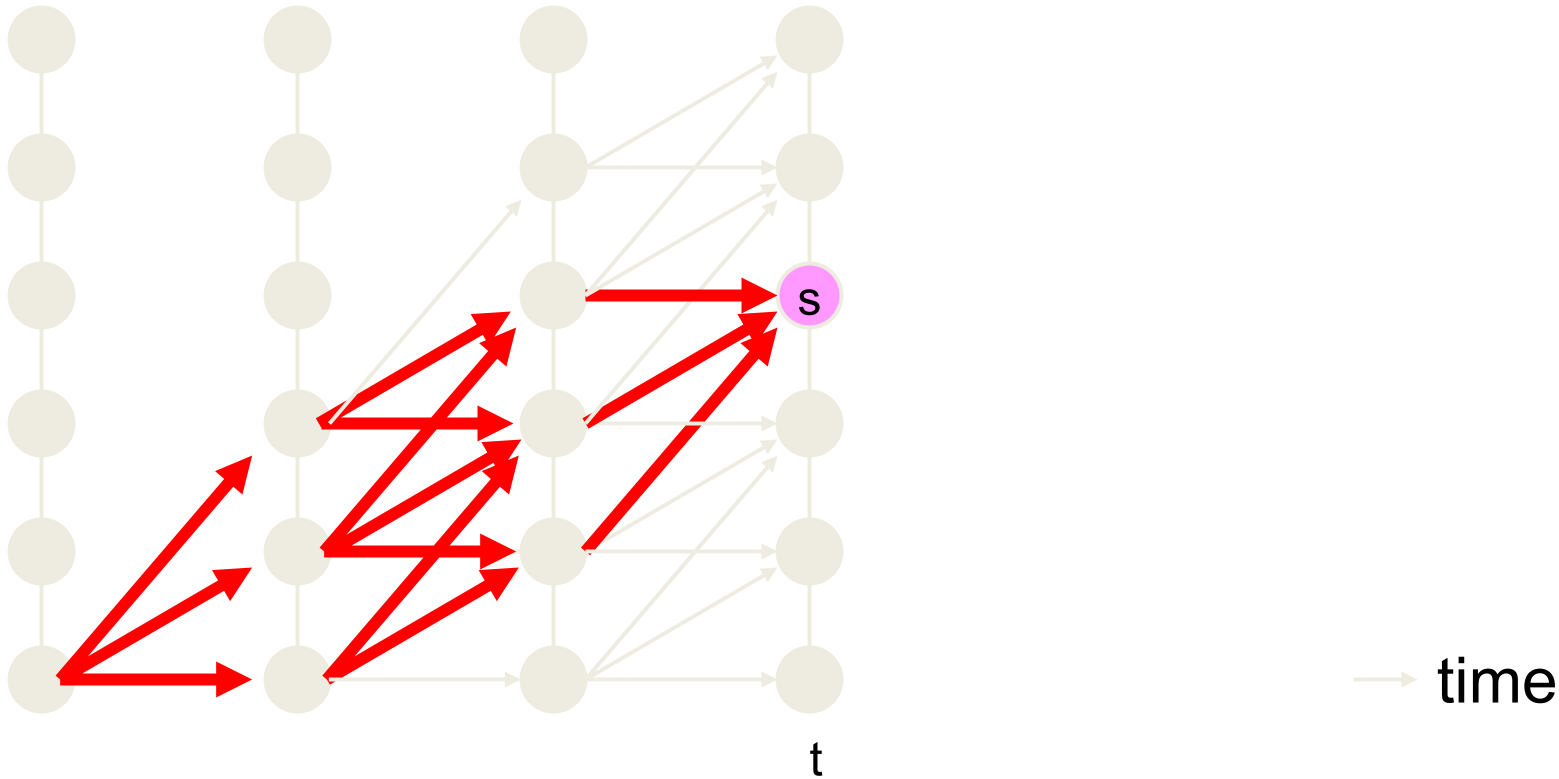
$$P(state(t) = s, x_1, x_2, \ldots, x_T)$$

- This can be decomposed into two multiplicative sections
  - The section of the lattice leading into state *s* at time t and the section leading out of it



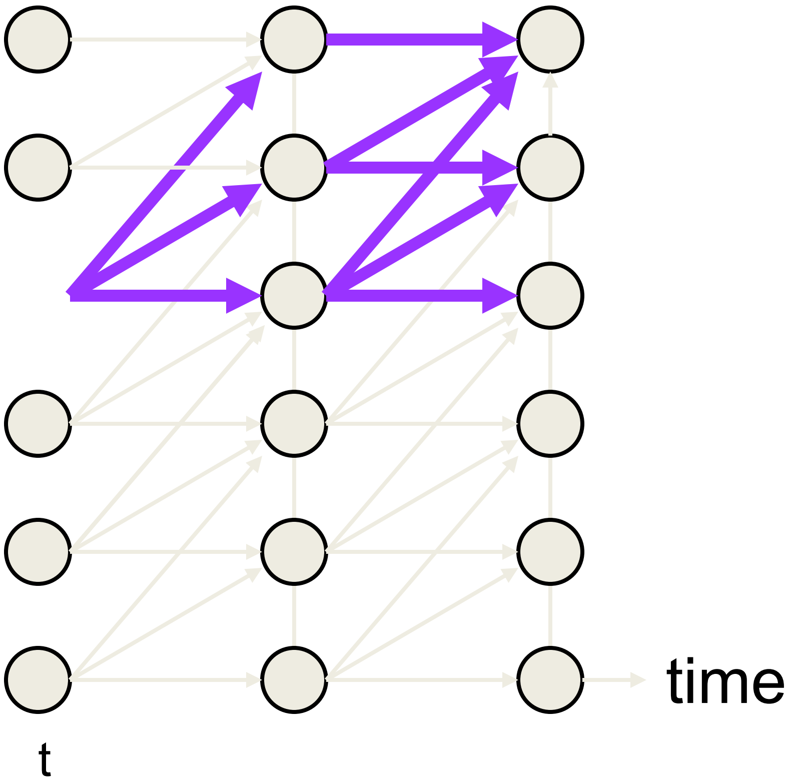
The Forward Paths

- The probability of the red section is the total probability of all state sequences ending at state *s* at time *t*
  - This is simply $\alpha(s,t)$
  - Can be computed using the forward algorithm



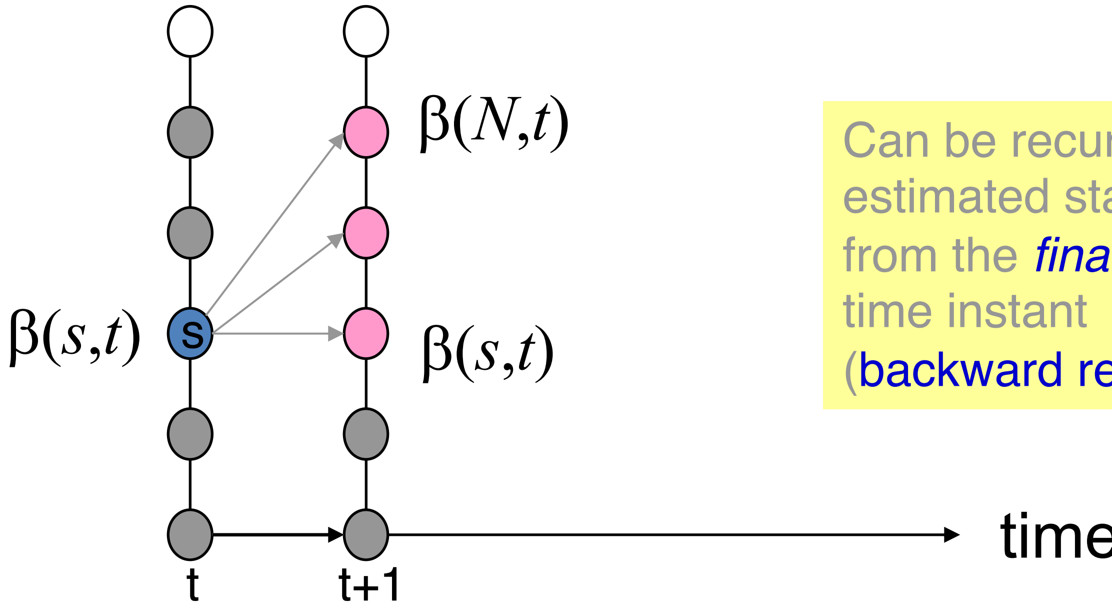
The Backward Paths

- The blue portion represents the probability of all state sequences that began at state *s* at time *t*
  - Like the red portion it can be computed using a *backward recursion*



The Backward Recursion

$$\beta(s,t) = P(x_{t+1}, x_{t+2}, \ldots, x_T \mid state(t) = s)$$



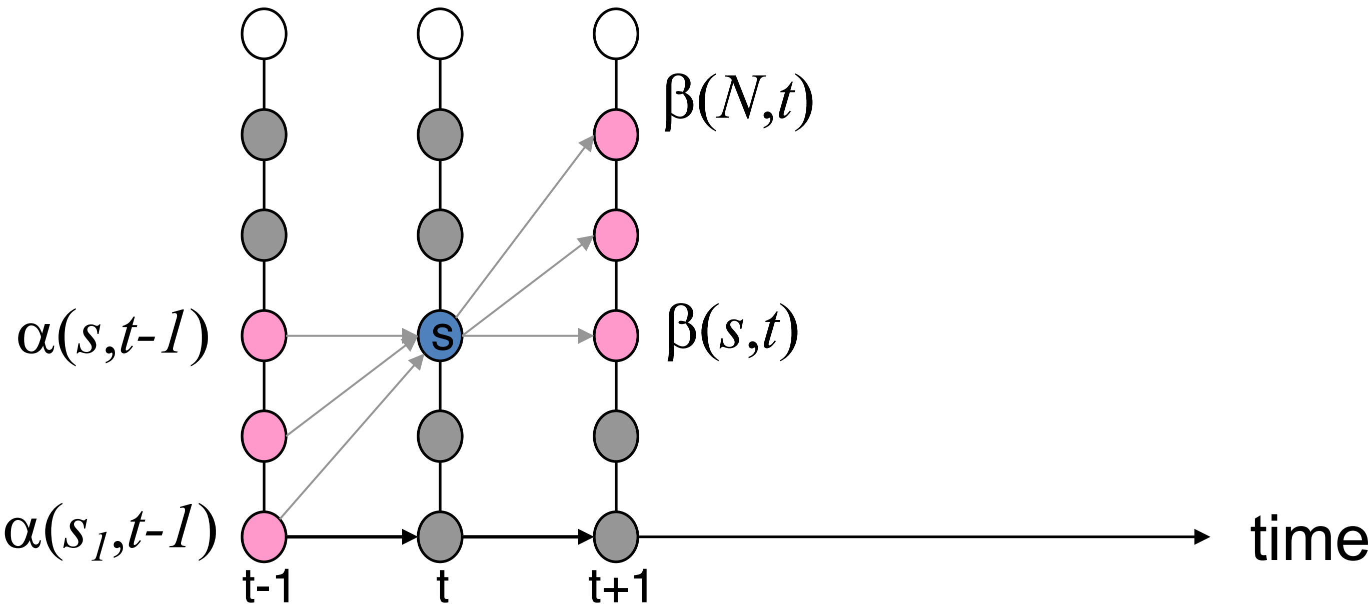
Can be recursively estimated starting from the *final* time time instant (backward recursion)

$$\beta(s,t) = \sum_{s'} \beta(s',t+1) P(s' \mid s) P(x_{t+1} \mid s')$$

- $\beta(s,t)$ is the total probability of ALL state sequences that depart from s at time t, and all observations after x_t
  - $\beta(s,T) = 1$ at the final time instant for all valid final states

The complete probability

$$\alpha(s,t)\beta(s,t) = P(x_{t+1}, x_{t+2}, \ldots, x_T, state(t) = s)$$



Poll 3

Posterior probability of a state

- The probability that the process was in state *s* at time *t*, given that we have observed the data is obtained by simple normalization

$$P(state(t)=s \mid Obs) = \frac{P(state(t)=s, x_1, x_2, \ldots, x_T)}{\sum_{s'} P(state(t)=s, x_1, x_2, \ldots, x_T)} = \frac{\alpha(s,t)\beta(s,t)}{\sum_{s'} \alpha(s',t)\beta(s',t)}$$

- This term is often referred to as the gamma term and denoted by $\gamma_{s,t}$

Update rules at each iteration

$$\pi(s_i) = \frac{\sum_{Obs} P(state(t=1) = s_i \mid Obs)}{\text{Total no. of observation sequences}}$$

$$P(s_j \mid s_i) = \frac{\sum_{Obs} \sum_{t} P(state(t) = s_i, state(t+1) = s_j \mid Obs)}{\sum_{Obs} \sum_{t} P(state(t) = s_i \mid Obs)}$$

$$\mu_i = \frac{\sum_{Obs} \sum_{t} P(state(t) = s_i \mid Obs) X_{Obs,t}}{\sum_{Obs} \sum_{t} P(state(t) = s_i \mid Obs)}$$

$$\Theta_i = \frac{\sum_{Obs} \sum_{t} P(state(t) = s_i \mid Obs)(X_{Obs,t} - \mu_i)(X_{Obs,t} - \mu_i)^T}{\sum_{Obs} \sum_{t} P(state(t) = s_i \mid Obs)}$$

- These have been found

Update rules at each iteration

$$\pi(s_i) = \frac{\sum_{Obs} P(state(t=1) = s_i \mid Obs)}{\text{Total no. of observation sequences}}$$

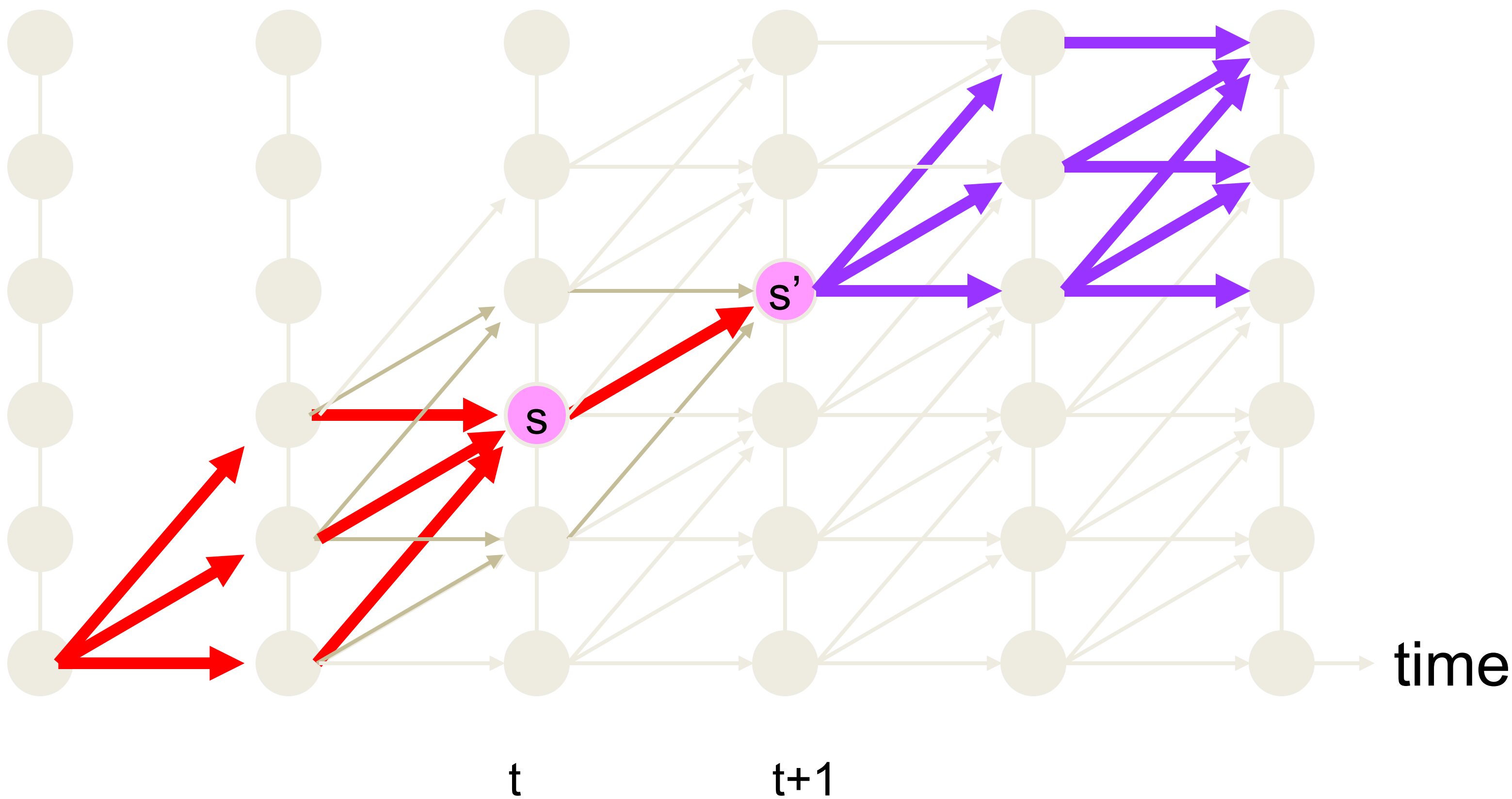
$$P(s_j \mid s_i) = \frac{\sum_{Obs}\sum_{t} P(state(t) = s_i, state(t+1) = s_j \mid Obs)}{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)}$$

$$\mu_i = \frac{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs) X_{Obs,t}}{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)}$$

$$\Theta_i = \frac{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)(X_{Obs,t} - \mu_i)(X_{Obs,t} - \mu_i)^T}{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)}$$

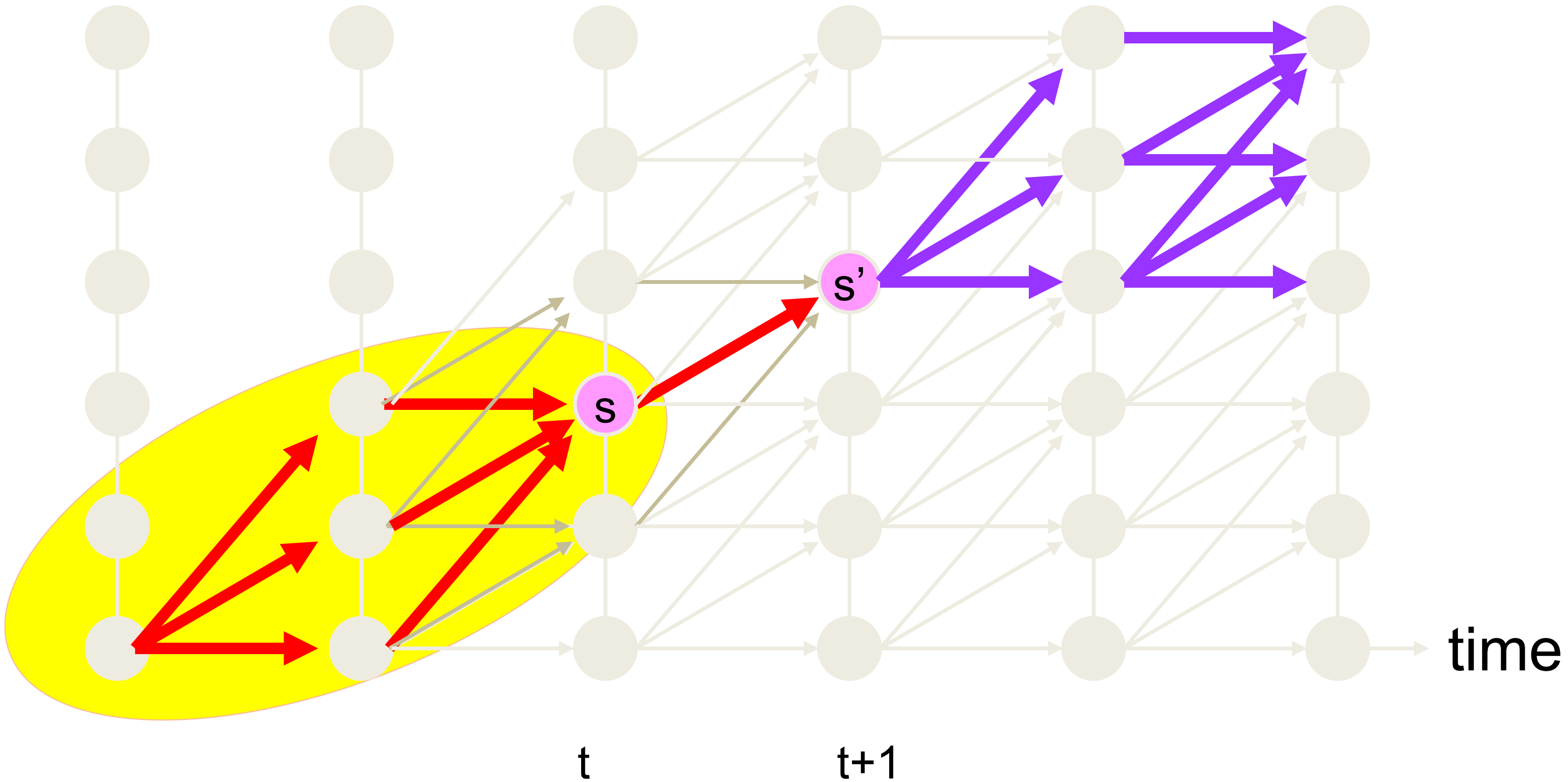
- Where did these terms come from?

$$P(state(t)=s, state(t+1)=s', x_1, x_2, \ldots, x_T)$$



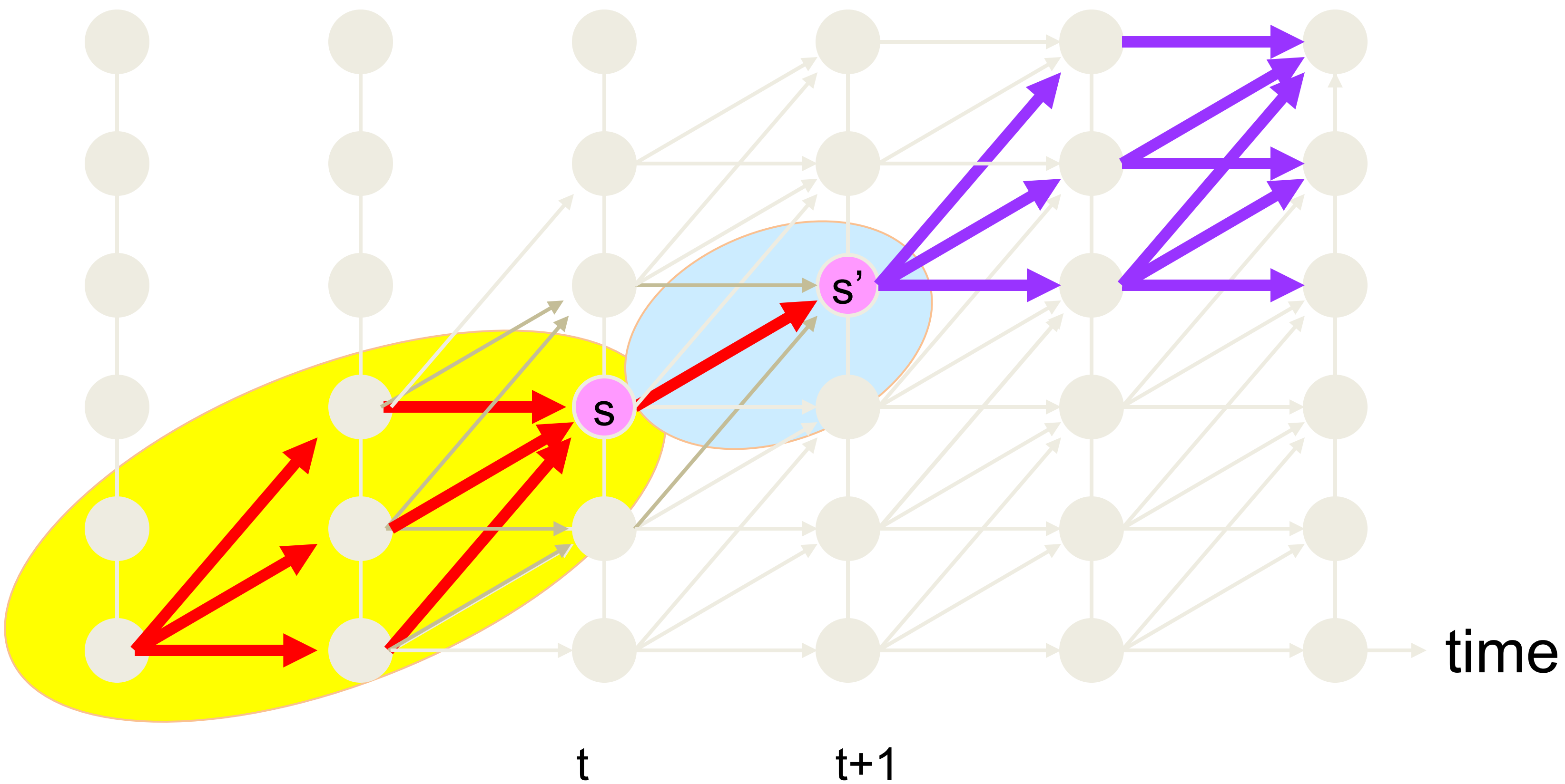
$$P(state(t)=s, state(t+1)=s', x_1, x_2, \ldots, x_T)$$

$$\alpha(s,t)$$



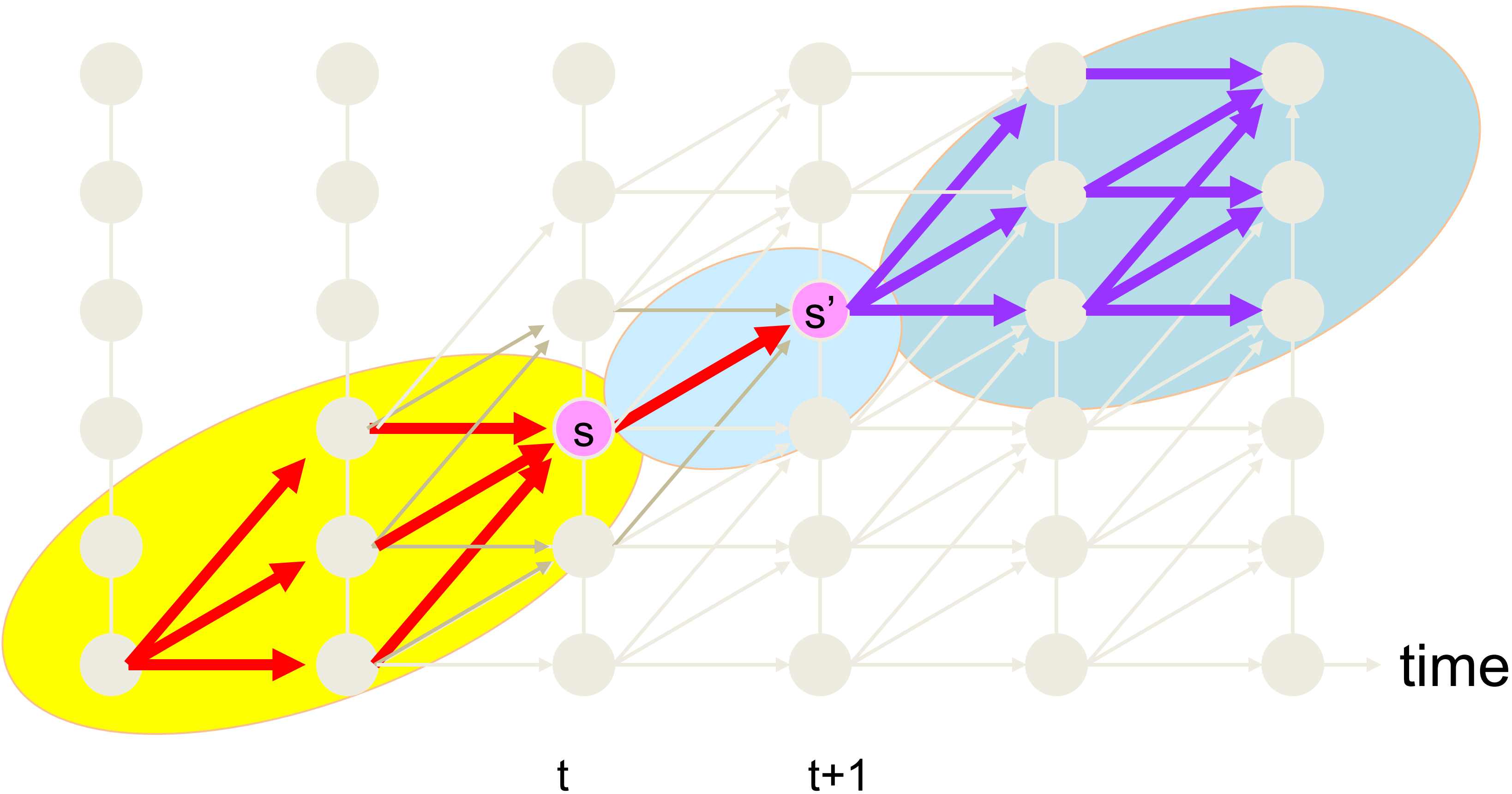
$$P(state(t)=s, state(t+1)=s', x_1, x_2, \ldots, x_T)$$

$$\alpha(s,t)\, P(s'|s) P(x_{t+1}|s')$$



$$P(state(t) = s, state(t+1) = s', x_1, x_2, \ldots, x_T)$$

$$\alpha(s,t) P(s'|s) P(x_{t+1}|s') \beta(s', t+1)$$



The a posteriori probability of transition

$$P(state(t)=s, state(t+1)=s' \mid Obs) = \frac{\alpha(s,t)P(s'\mid s)P(x_{t+1}\mid s')\beta(s',t+1)}{\sum_{s_1}\sum_{s_2}\alpha(s_1,t)P(s_2\mid s_1)P(x_{t+1}\mid s_2)\beta(s_2,t+1)}$$

- The a posteriori probability of a transition given an observation

Update rules at each iteration

$$\pi(s_i) = \frac{\sum_{Obs} P(state(t=1) = s_i \mid Obs)}{\text{Total no. of observation sequences}}$$

$$P(s_j \mid s_i) = \frac{\sum_{Obs}\sum_{t} P(state(t) = s_i, state(t+1) = s_j \mid Obs)}{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)}$$

$$\mu_i = \frac{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs) X_{Obs,t}}{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)}$$

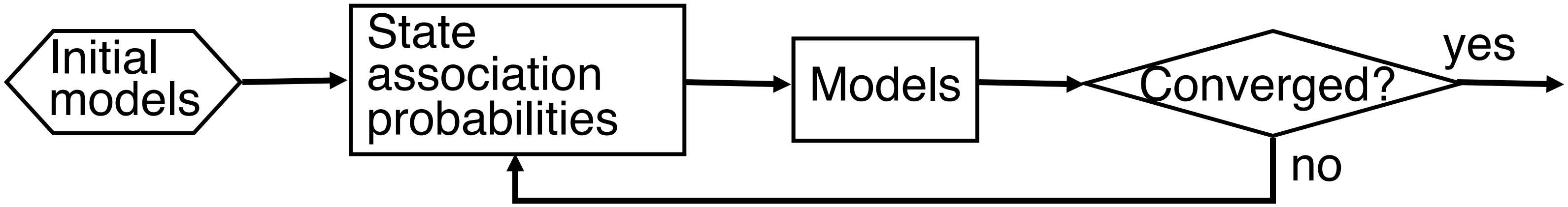
$$\Theta_i = \frac{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)(X_{Obs,t} - \mu_i)(X_{Obs,t} - \mu_i)^T}{\sum_{Obs}\sum_{t} P(state(t) = s_i \mid Obs)}$$

- These have been found

Poll 4

Training without explicit segmentation: Baum-Welch training

- Every feature vector associated with every state of every HMM with a probability



- Probabilities computed using the forward-backward algorithm
- Soft decisions taken at the level of HMM state
- In practice, the segmentation based Viterbi training is much easier to implement and is much faster
- The difference in performance between the two is small, especially if we have lots of training data

HMM Issues

- How to find the best state sequence: Covered
- How to learn HMM parameters: Covered
- How to compute the probability of an observation sequence: Covered

Magic numbers

- How many states:
  - No nice automatic technique to learn this
  - You choose
    - For speech, HMM topology is usually left to right (no backward transitions)
    - For other cyclic processes, topology must reflect nature of process
    - No. of states – 3 per phoneme in speech
    - For other processes, depends on estimated no. of distinct states in process

Applications of HMMs

- Classification:
  - Learn HMMs for the various classes of time series from training data
  - Compute probability of test time series using the HMMs for each class
  - Use in a Bayesian classifier
  - Speech recognition, vision, gene sequencing, character recognition, text mining…
- Prediction
- Tracking

Applications of HMMs

- Segmentation:
  - Given HMMs for various events, find event boundaries
    - Simply find the best state sequence and the locations where state identities change

- Automatic speech segmentation, text segmentation by topic, geneome segmentation, …