Markov Models

&

Hidden Markov Models

What is an Markov Chain Model?

- A stochastic model that describe the probabilities of transition among the states of a system.
- It is a random process that undergoes transitions from one state to another on a state space.
- Change of states depends probabilistically only on the current state of the system.
- It is required to possess a property that is usually characterized as "memoryless": the probability distribution of the next state depends only on the current state and not on the sequence of events that preceded it.

Markov Assumptions

- The probabilities of moving from a state to all others sum to one.
- The probabilities apply to all system participants.
- The probabilities are constant over time.

Configuration of the Markov-Chain Model

- Markov systems deal with stochastic environments in which possible "outcomes occur at the end of a well-defined, usually first period".

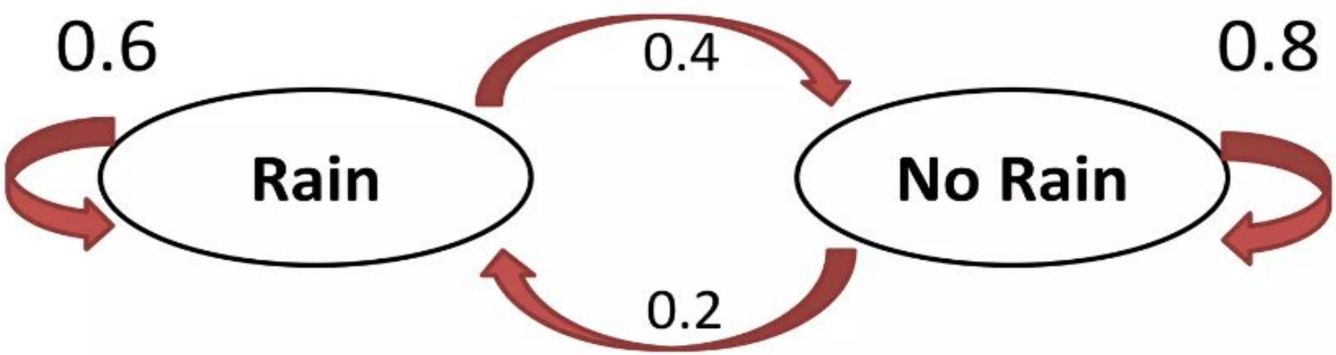
- This situation further involves a multi-period time frame, during which the occurring consumer's transient behavior, for example, affects the stability of the firm's performance.
- This transient behavior, whose future outcome is unknown but needs to be predicated, creates inter-period transitional probabilities. - Such a stochastic process, known as the Markov process, contains a special case, where the transitional probabilities from one time period to another remains stationary, in which case the process is referred to as the Markov-Chain.

Lets try to understand Markov chain from very simple example

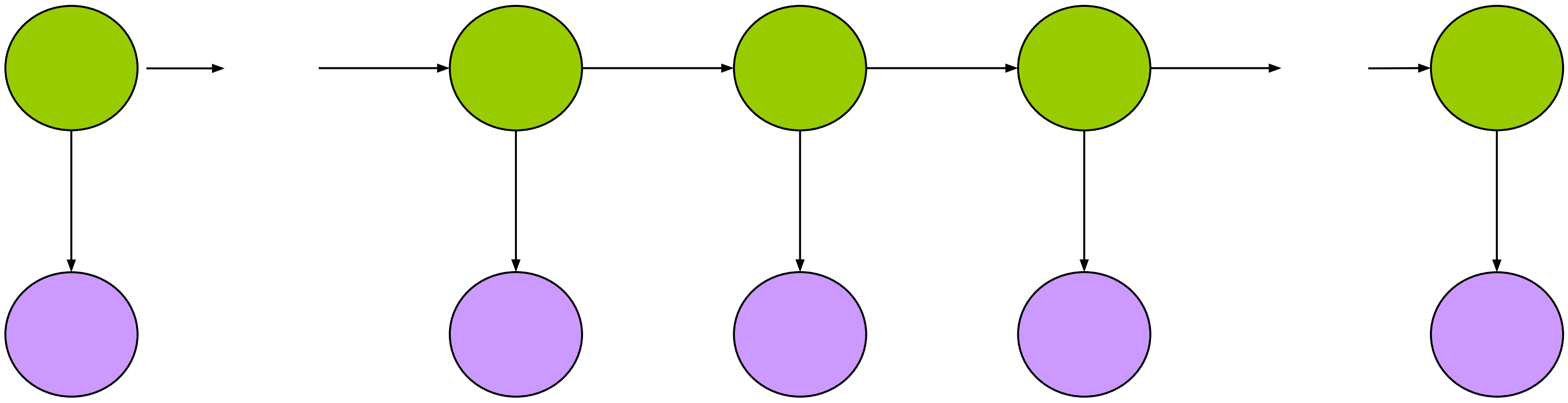
Weather:

- raining today → 60% rain tomorrow
  40% no rain tomorrow
- not raining today → 20% rain tomorrow
  80% no rain tomorrow

Stochastic Finite State Machine:

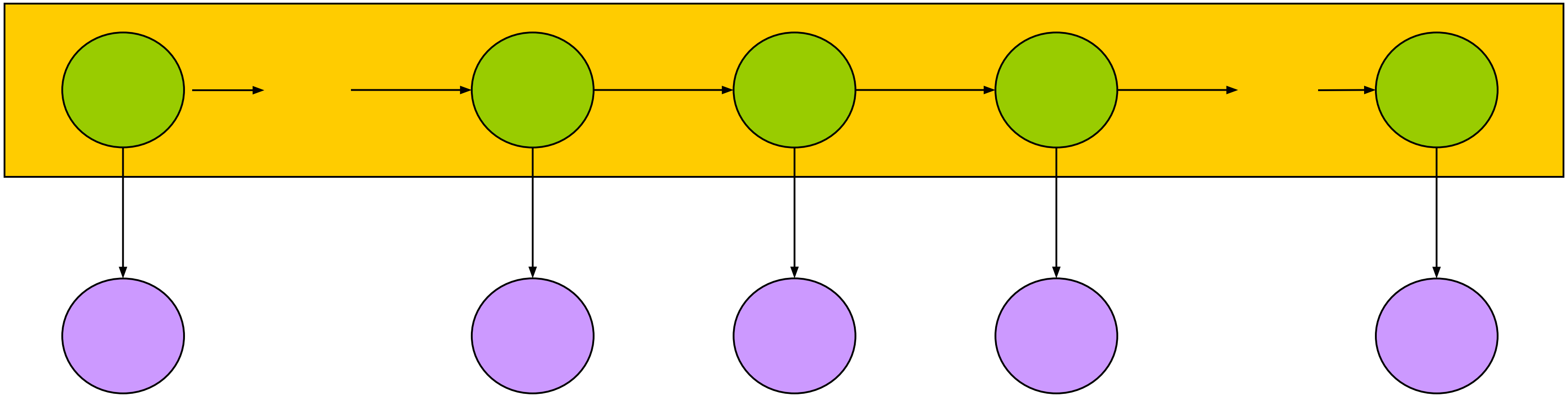


What is an HMM?



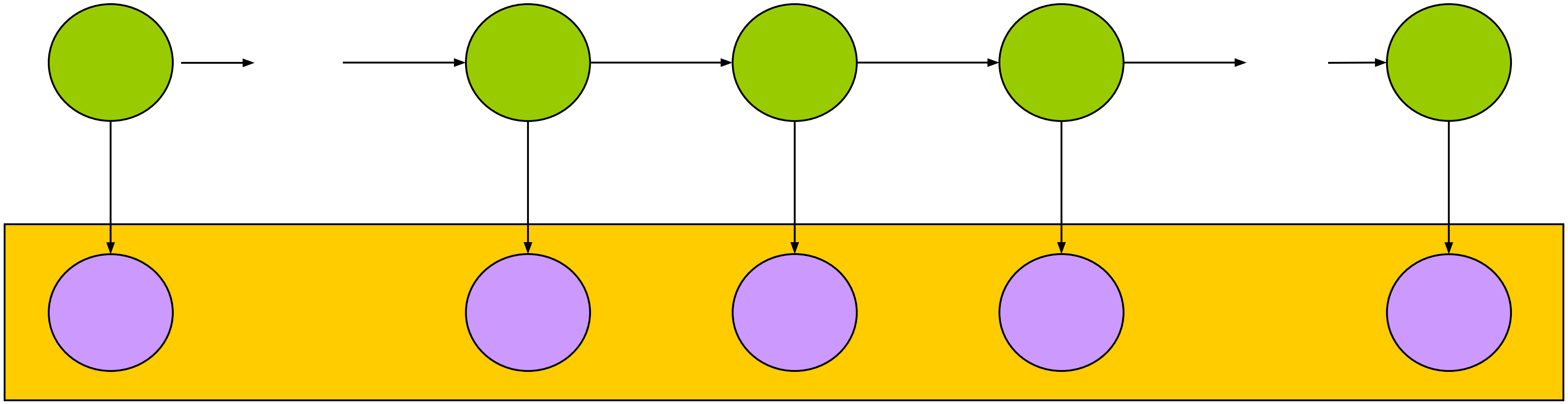
- Graphical Model
- Circles indicate states
- Arrows indicate probabilistic dependencies between states

What is an HMM?



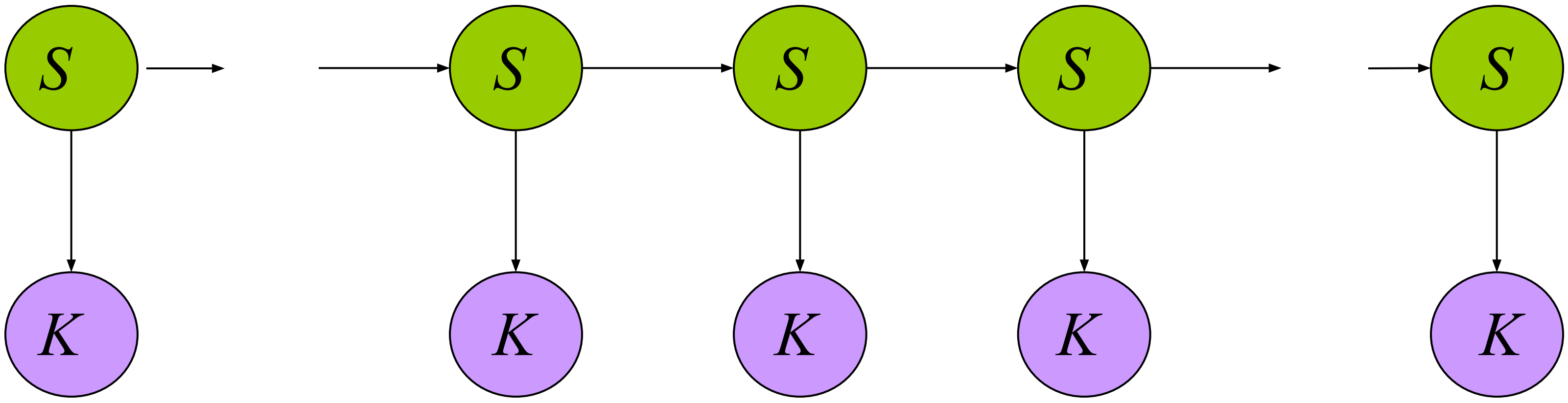
- Green circles are ***hidden states***
- Dependent only on the previous state
- "The past is independent of the future given the present."

What is an HMM?



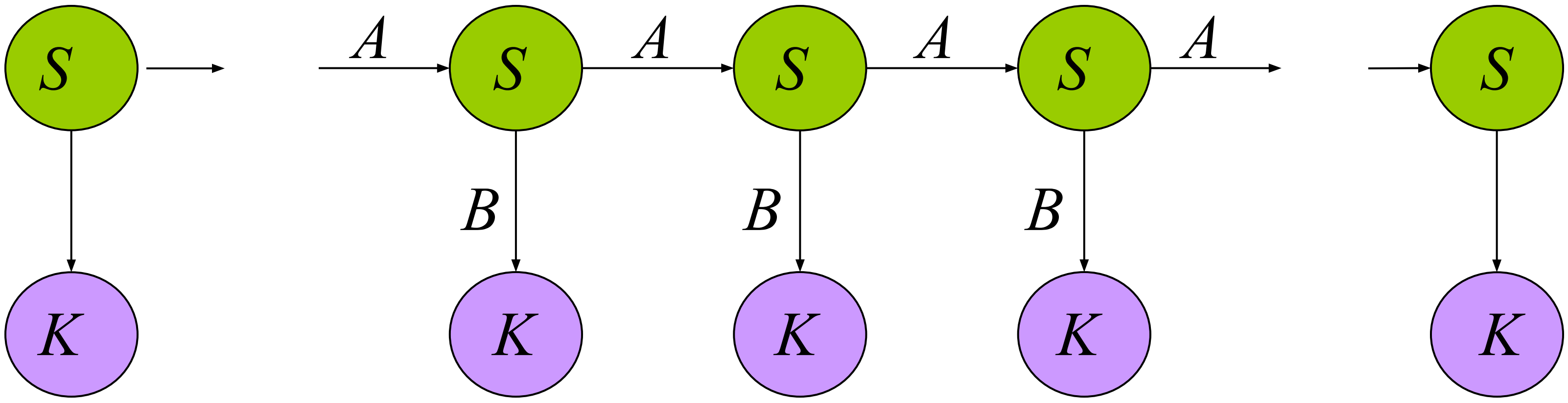
- Purple nodes are ***observed states***
- Dependent only on their corresponding hidden state

HMM Formalism



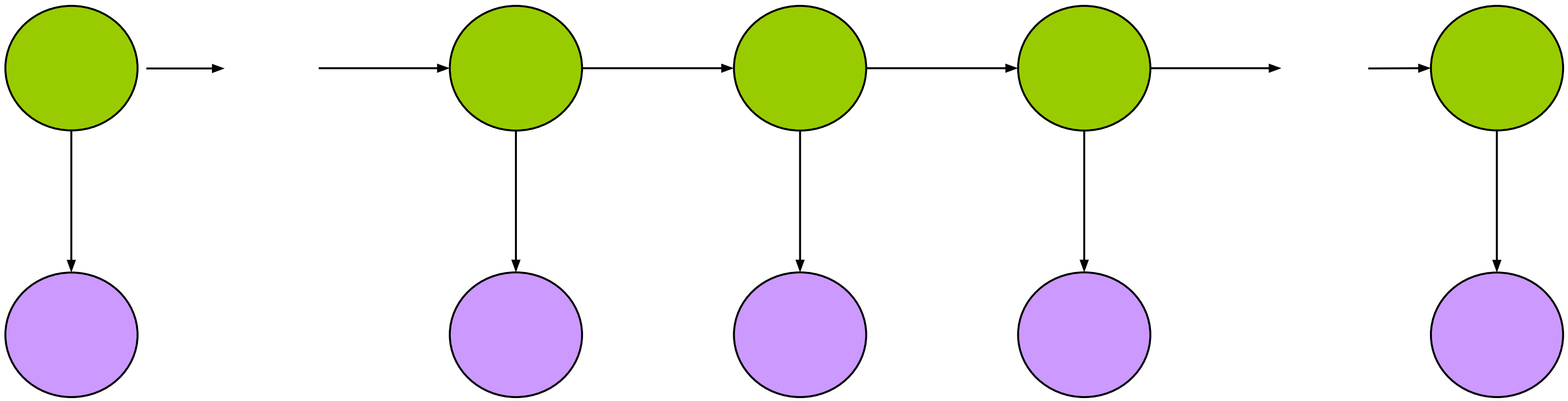
- $\{S, K, \Pi, A, B\}$
- S : $\{s_1 \ldots s_N\}$ are the values for the hidden states
- K : $\{k_1 \ldots k_M\}$ are the values for the observations

HMM Formalism



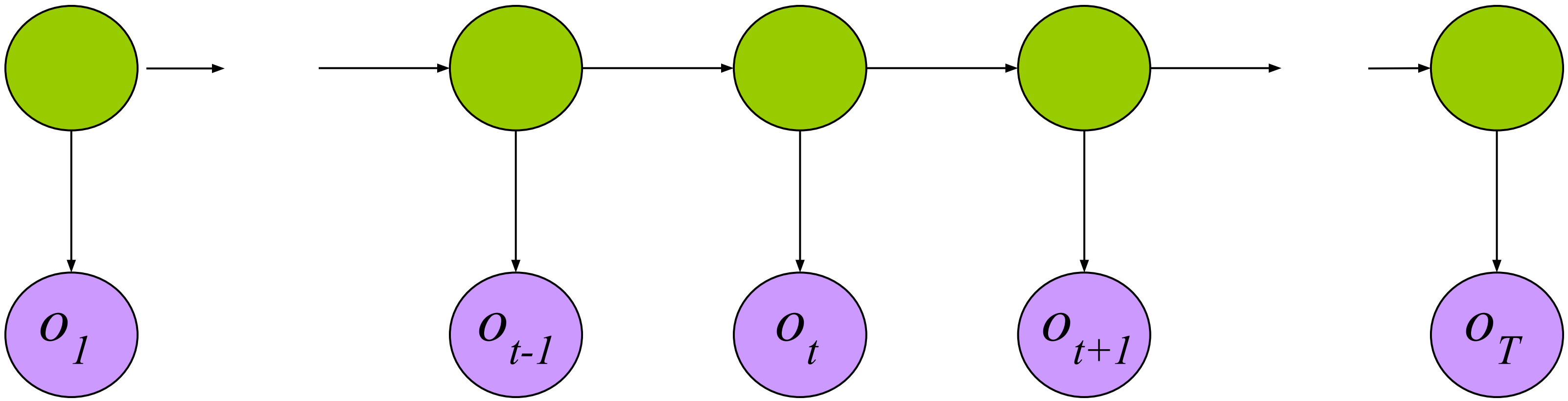
- $\{S, K, \Pi, A, B\}$
- $\Pi = \{\pi_\iota\}$ are the initial state probabilities
- $A = \{a_{ij}\}$ are the state transition probabilities
- $B = \{b_{ik}\}$ are the observation state probabilities

Inference in an HMM



- Compute the probability of a given observation sequence
- Given an observation sequence, compute the most likely hidden state sequence
- Given an observation sequence and set of possible models, which model most closely fits the data?

Decoding

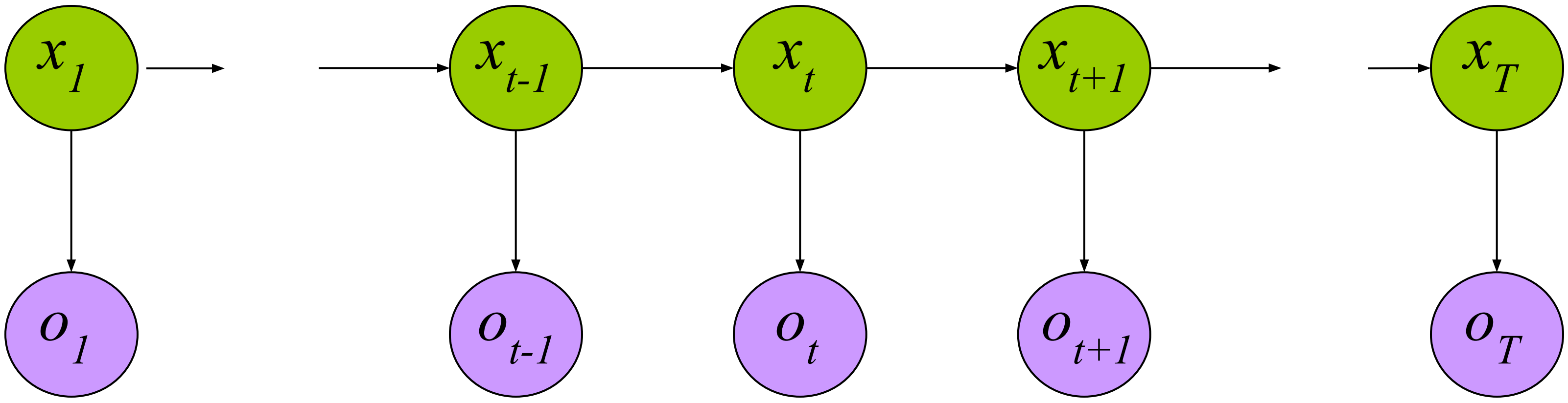


Given an observation sequence and a model, compute the probability of the observation sequence

$$O = (o_1 \ldots o_T),\ \mu = (A, B, \Pi)$$

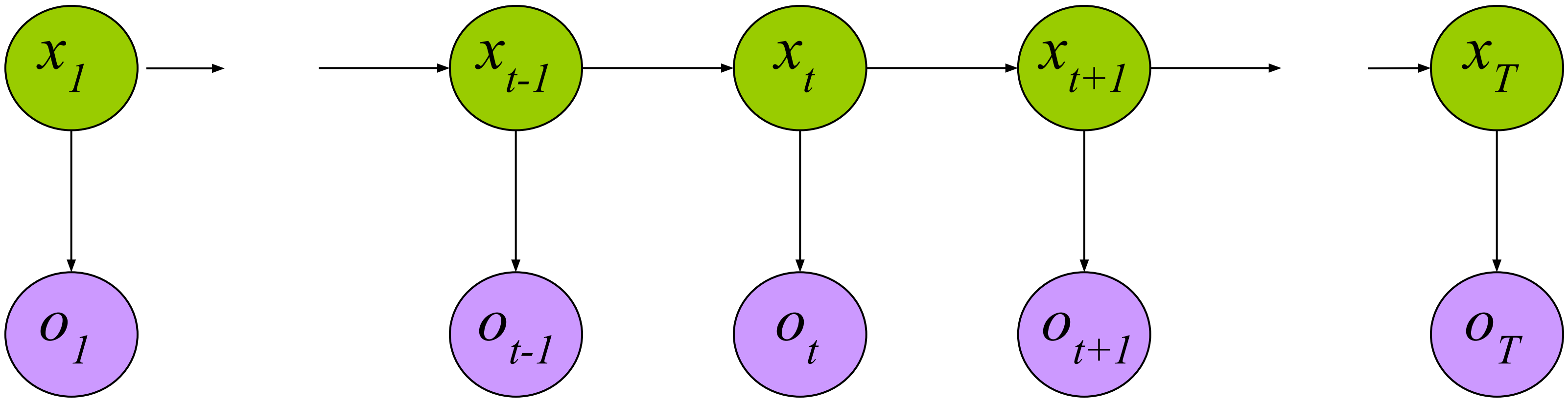
Compute $P(O \mid \mu)$

Decoding



$$P(O \mid X, \mu) = b_{x_1 o_1} b_{x_2 o_2} \dots b_{x_T o_T}$$

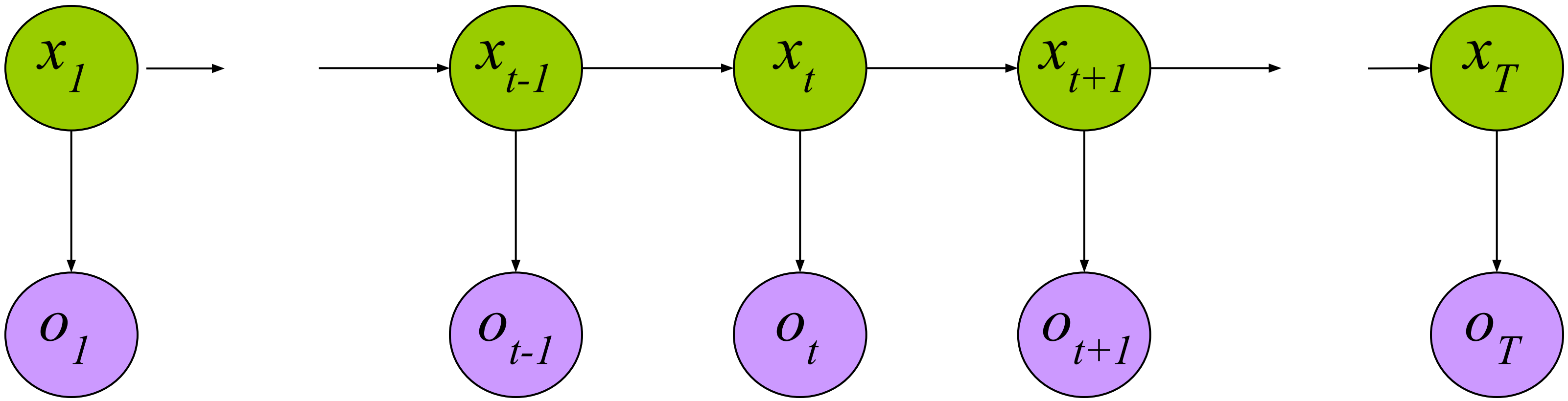
Decoding



$$P(O \mid X, \mu) = b_{x_1 o_1} b_{x_2 o_2} \ldots b_{x_T o_T}$$

$$\boxed{P(X \mid \mu) = \pi_{x_1} a_{x_1 x_2} a_{x_2 x_3} \ldots a_{x_{T-1} x_T}}$$

Decoding

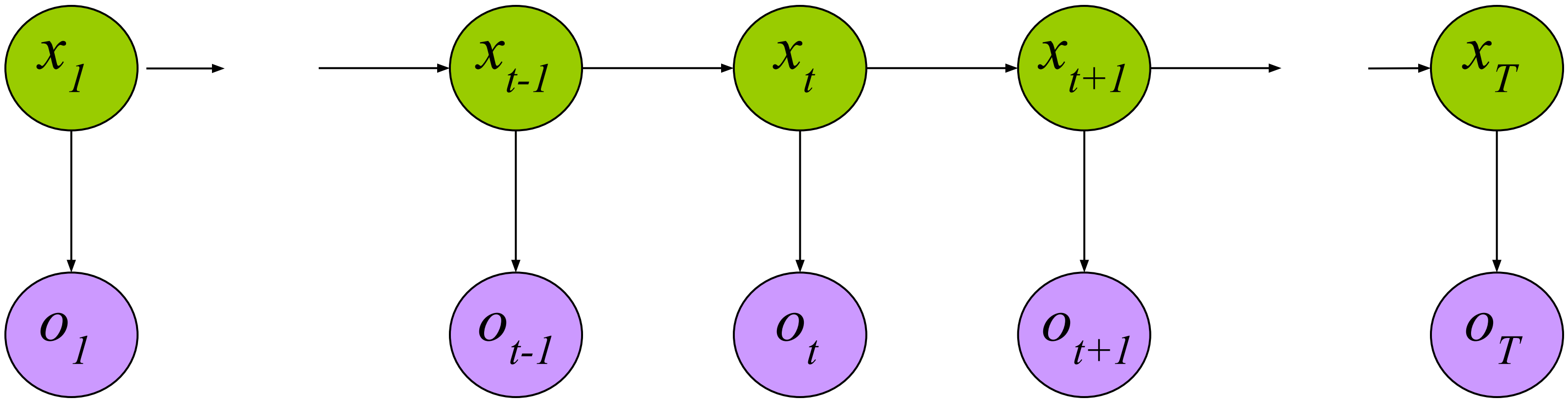


$$P(O \mid X, \mu) = b_{x_1 o_1} b_{x_2 o_2} \ldots b_{x_T o_T}$$

$$P(X \mid \mu) = \pi_{x_1} a_{x_1 x_2} a_{x_2 x_3} \ldots a_{x_{T-1} x_T}$$

$$\boxed{P(O, X \mid \mu) = P(O \mid X, \mu) P(X \mid \mu)}$$

Decoding



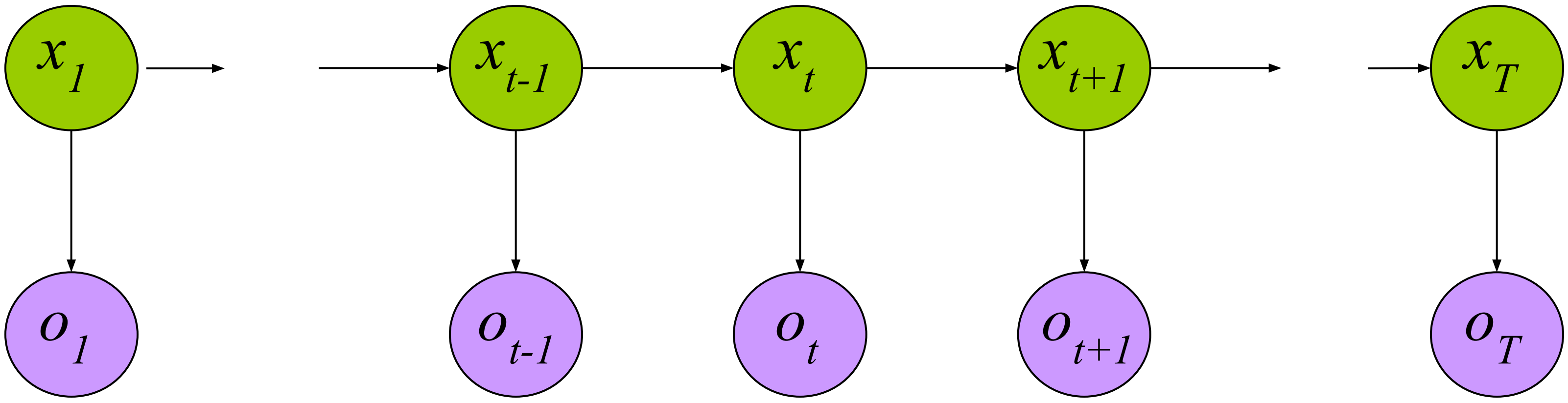
$$P(O \mid X, \mu) = b_{x_1 o_1} b_{x_2 o_2} \ldots b_{x_T o_T}$$

$$P(X \mid \mu) = \pi_{x_1} a_{x_1 x_2} a_{x_2 x_3} \ldots a_{x_{T-1} x_T}$$

$$P(O, X \mid \mu) = P(O \mid X, \mu) P(X \mid \mu)$$

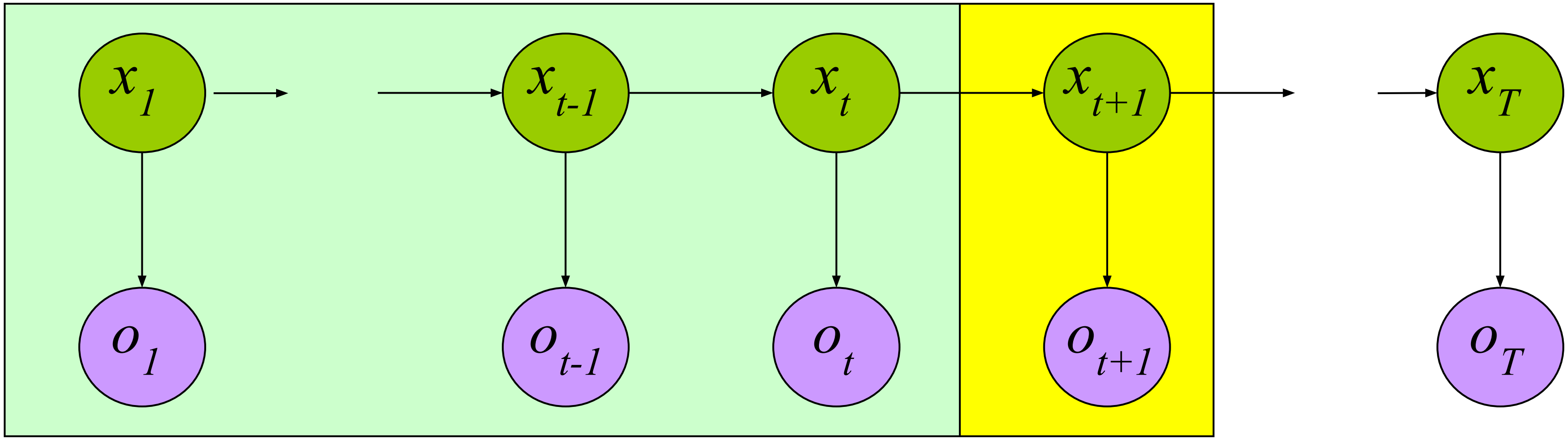
$$\boxed{P(O \mid \mu) = \sum_X P(O \mid X, \mu) P(X \mid \mu)}$$

Decoding



$$P(O \mid \mu) = \sum_{\{x_1 \dots x_T\}} \pi_{x_1} b_{x_1 o_1} \prod_{t=1}^{T-1} a_{x_t x_{t+1}} b_{x_{t+1} o_{t+1}}$$

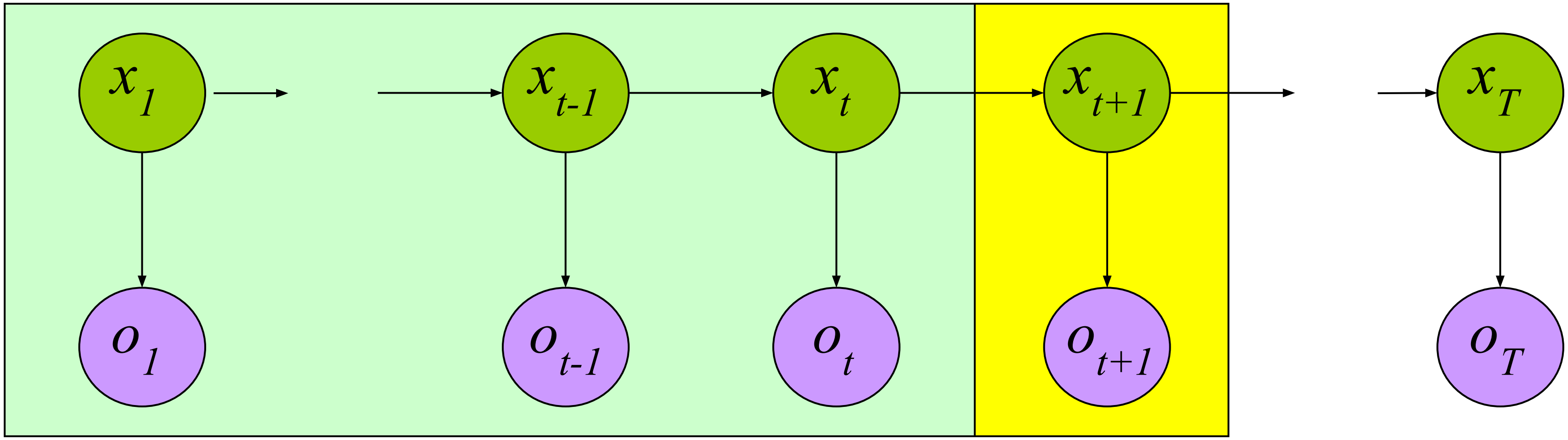
Forward Procedure



- Special structure gives us an efficient solution using *dynamic programming*.
- **Intuition**: Probability of the first *t* observations is the same for all possible *t*+1 length state sequences.
- **Define:**

$$\alpha_i(t) = P(o_1 \ldots o_t, x_t = i \mid \mu)$$

Forward Procedure



$$\alpha_j(t+1)$$

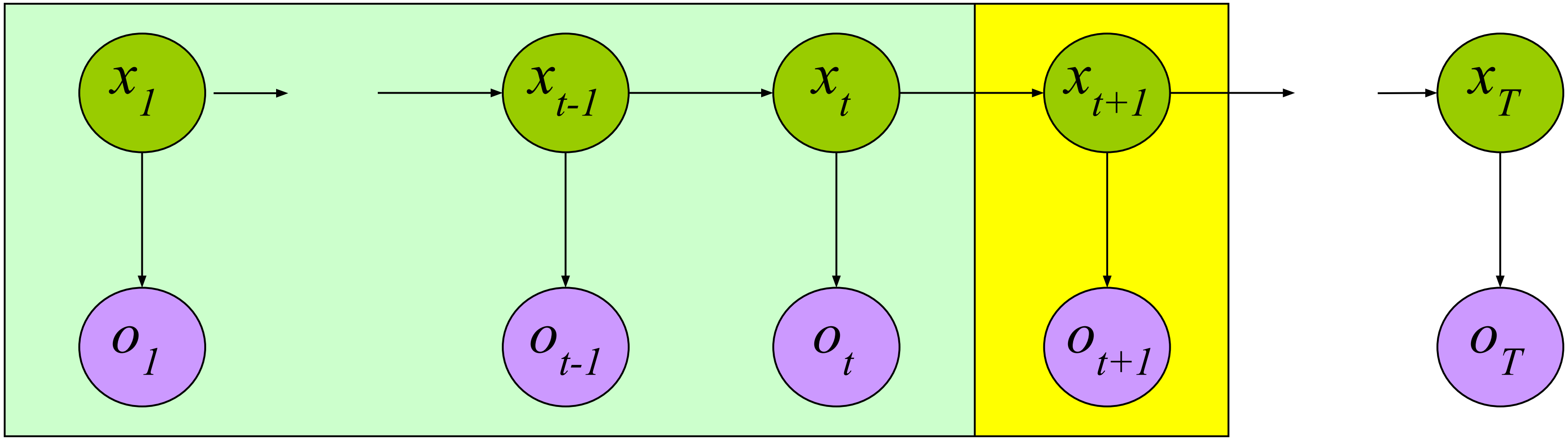
$$= P(o_1 \ldots o_{t+1}, x_{t+1} = j)$$

$$= P(o_1 \ldots o_{t+1} \mid x_{t+1} = j) P(x_{t+1} = j)$$

$$= P(o_1 \ldots o_t \mid x_{t+1} = j) P(o_{t+1} \mid x_{t+1} = j) P(x_{t+1} = j)$$

$$= P(o_1 \ldots o_t, x_{t+1} = j) P(o_{t+1} \mid x_{t+1} = j)$$

Forward Procedure



$$\alpha_j(t+1)$$

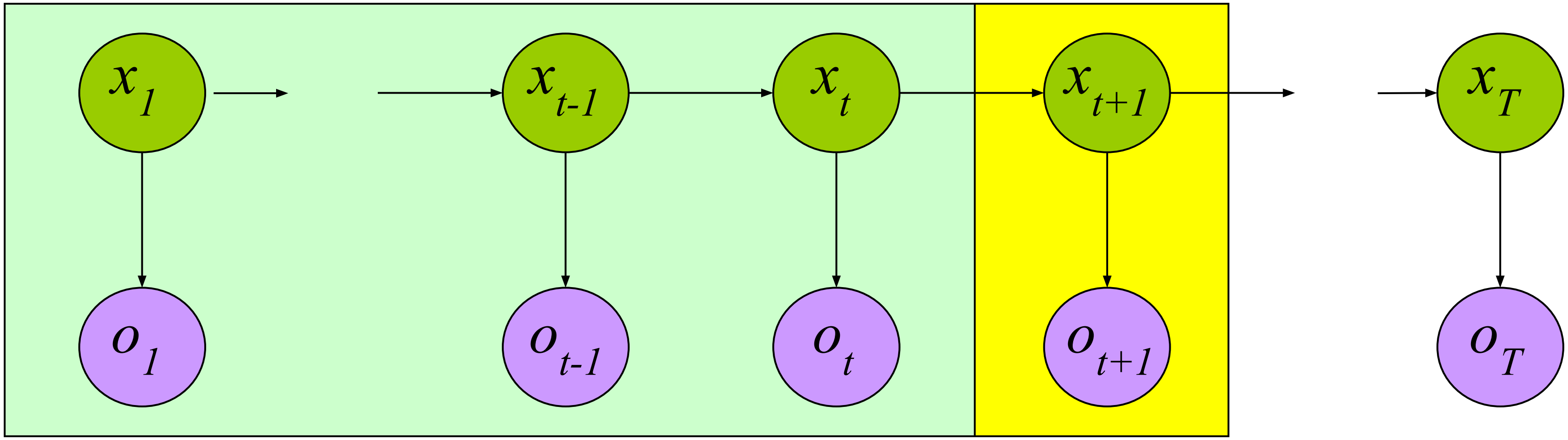
$$= P(o_1 \ldots o_{t+1}, x_{t+1} = j)$$

$$= P(o_1 \ldots o_{t+1} \mid x_{t+1} = j) P(x_{t+1} = j)$$

$$= P(o_1 \ldots o_t \mid x_{t+1} = j) P(o_{t+1} \mid x_{t+1} = j) P(x_{t+1} = j)$$

$$= P(o_1 \ldots o_t, x_{t+1} = j) P(o_{t+1} \mid x_{t+1} = j)$$

Forward Procedure



$$\alpha_j(t+1)$$

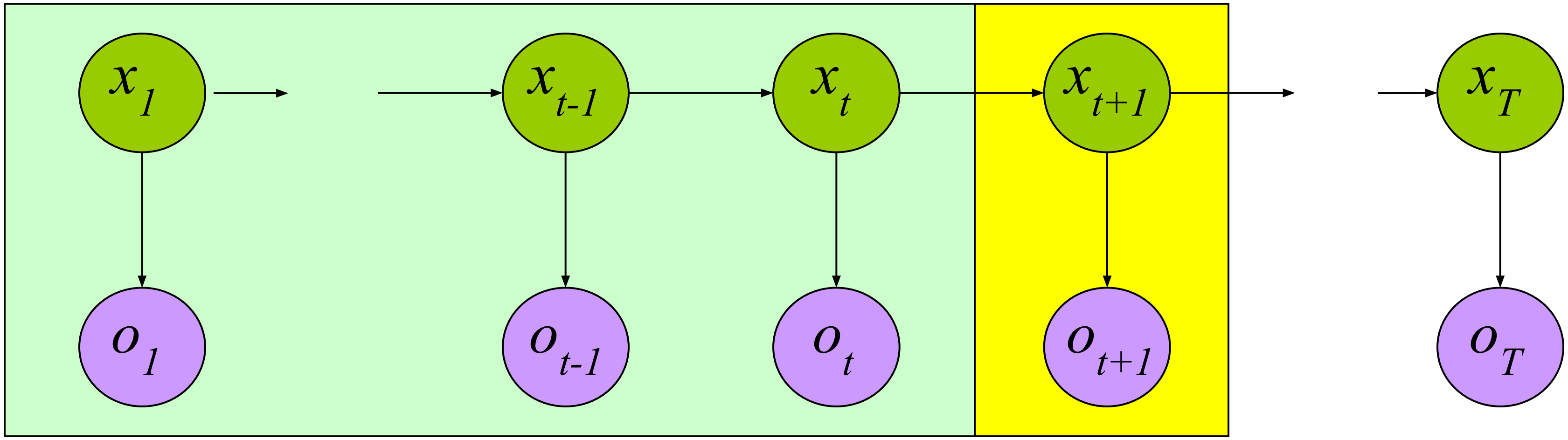
$$= P(o_1 \ldots o_{t+1}, x_{t+1} = j)$$

$$= P(o_1 \ldots o_{t+1} \mid x_{t+1} = j) P(x_{t+1} = j)$$

$$\boxed{= P(o_1 \ldots o_t \mid x_{t+1} = j) P(o_{t+1} \mid x_{t+1} = j) P(x_{t+1} = j)}$$

$$= P(o_1 \ldots o_t, x_{t+1} = j) P(o_{t+1} \mid x_{t+1} = j)$$

Forward Procedure



$$\alpha_j(t+1)$$

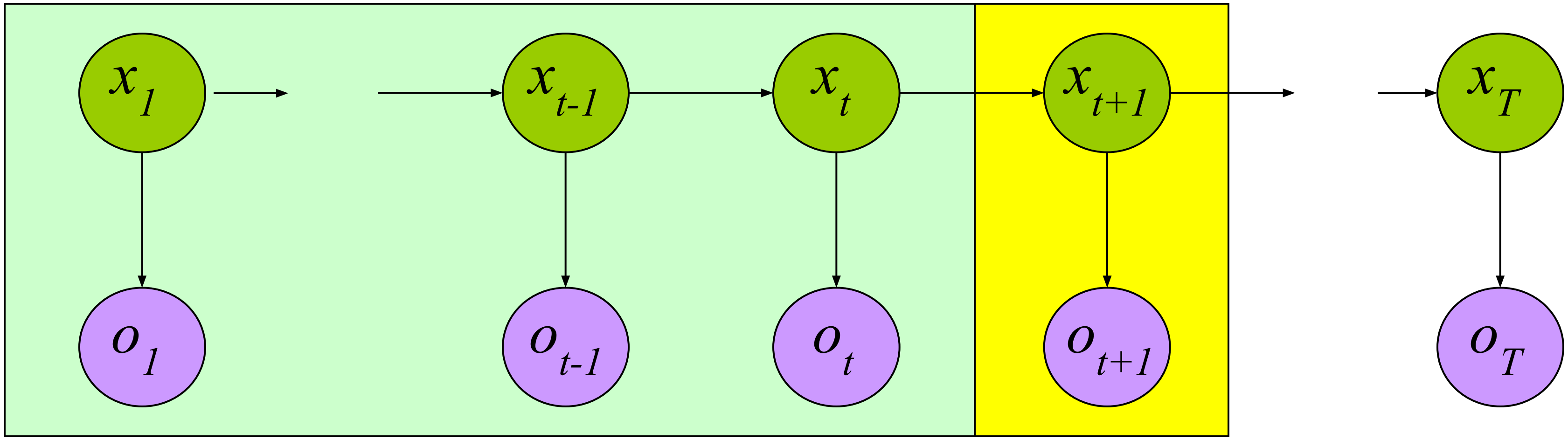
$$= P(o_1 \dots o_{t+1}, x_{t+1} = j)$$

$$= P(o_1 \dots o_{t+1} \mid x_{t+1} = j) P(x_{t+1} = j)$$

$$= P(o_1 \dots o_t \mid x_{t+1} = j) P(o_{t+1} \mid x_{t+1} = j) P(x_{t+1} = j)$$

$$= P(o_1 \dots o_t, x_{t+1} = j) P(o_{t+1} \mid x_{t+1} = j)$$

Forward Procedure



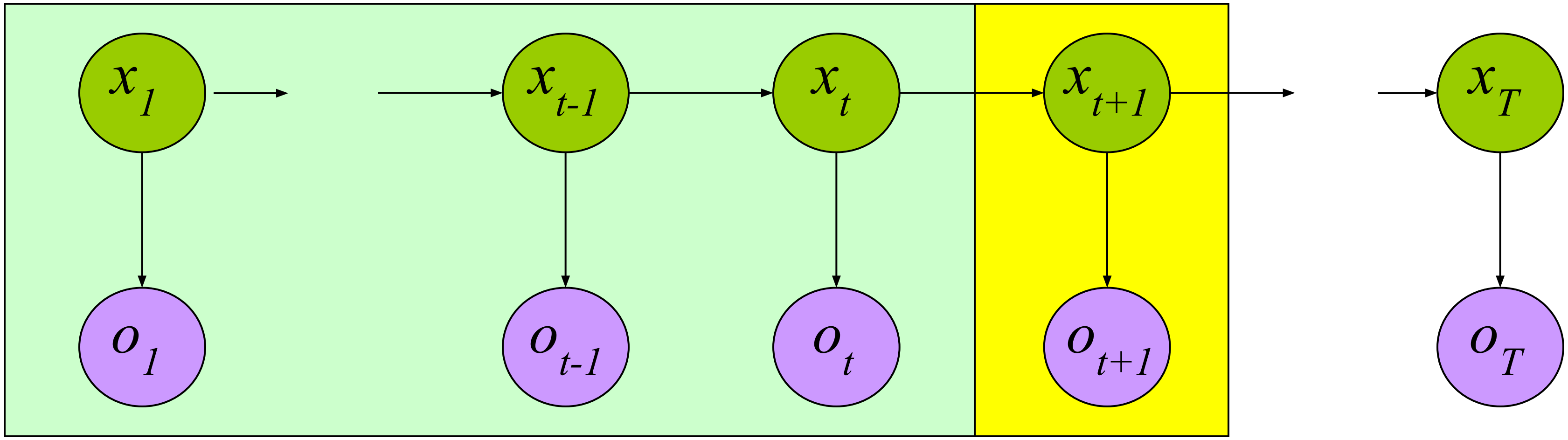
$$= \sum_{i=1...N} P(o_1...o_t, x_t = i, x_{t+1} = j) P(o_{t+1} | x_{t+1} = j)$$

$$= \sum_{i=1...N} P(o_1...o_t, x_{t+1} = j | x_t = i) P(x_t = i) P(o_{t+1} | x_{t+1} = j)$$

$$= \sum_{i=1...N} P(o_1...o_t, x_t = i) P(x_{t+1} = j | x_t = i) P(o_{t+1} | x_{t+1} = j)$$

$$= \sum_{i=1...N} \alpha_i(t) a_{ij} b_{jo_{t+1}}$$

Forward Procedure



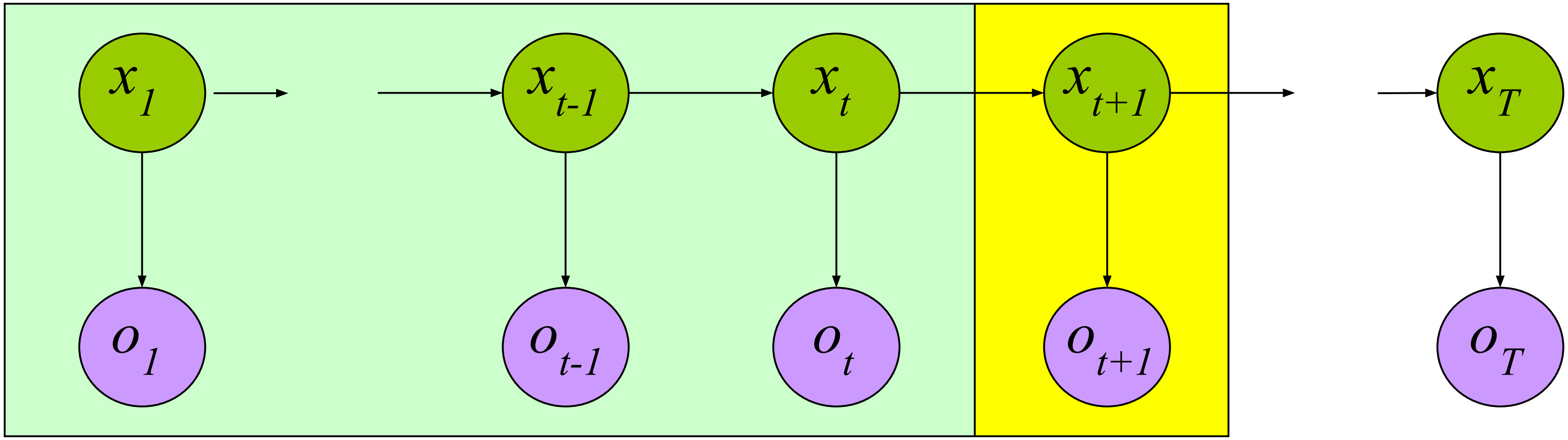
$$= \sum_{i=1\ldots N} P(o_1 \ldots o_t, x_t = i, x_{t+1} = j) P(o_{t+1} \mid x_{t+1} = j)$$

$$= \sum_{i=1\ldots N} P(o_1 \ldots o_t, x_{t+1} = j \mid x_t = i) P(x_t = i) P(o_{t+1} \mid x_{t+1} = j)$$

$$= \sum_{i=1\ldots N} P(o_1 \ldots o_t, x_t = i) P(x_{t+1} = j \mid x_t = i) P(o_{t+1} \mid x_{t+1} = j)$$

$$= \sum_{i=1\ldots N} \alpha_i(t) a_{ij} b_{j o_{t+1}}$$

Forward Procedure



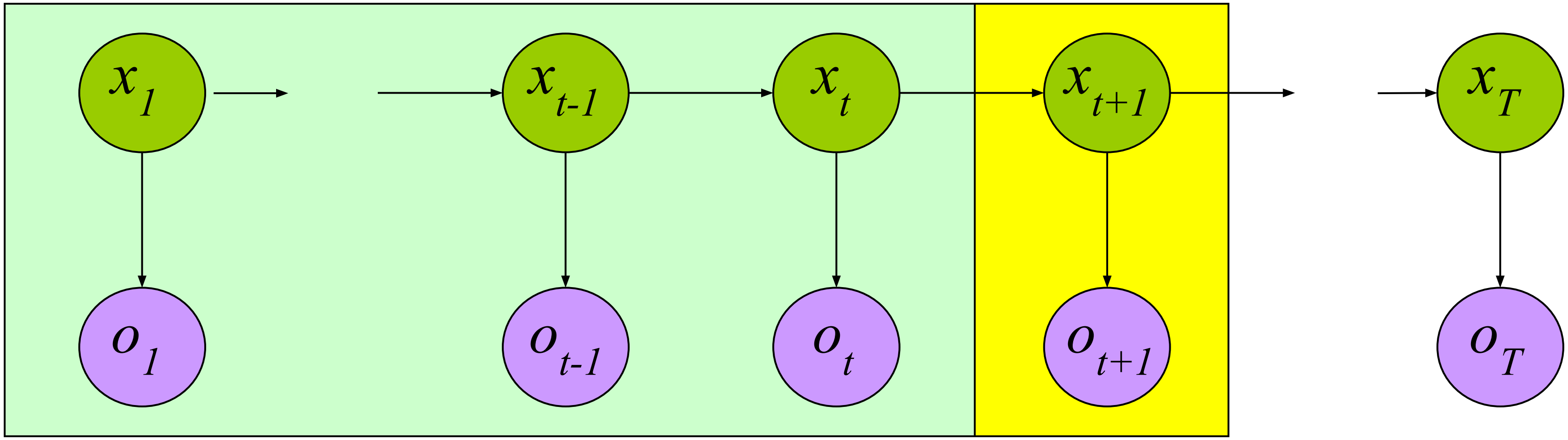
$$= \sum_{i=1\ldots N} P(o_1 \ldots o_t, x_t = i, x_{t+1} = j) P(o_{t+1} \mid x_{t+1} = j)$$

$$= \sum_{i=1\ldots N} P(o_1 \ldots o_t, x_{t+1} = j \mid x_t = i) P(x_t = i) P(o_{t+1} \mid x_{t+1} = j)$$

$$\boxed{= \sum_{i=1\ldots N} P(o_1 \ldots o_t, x_t = i) P(x_{t+1} = j \mid x_t = i) P(o_{t+1} \mid x_{t+1} = j)}$$

$$= \sum_{i=1\ldots N} \alpha_i(t) a_{ij} b_{j o_{t+1}}$$

Forward Procedure



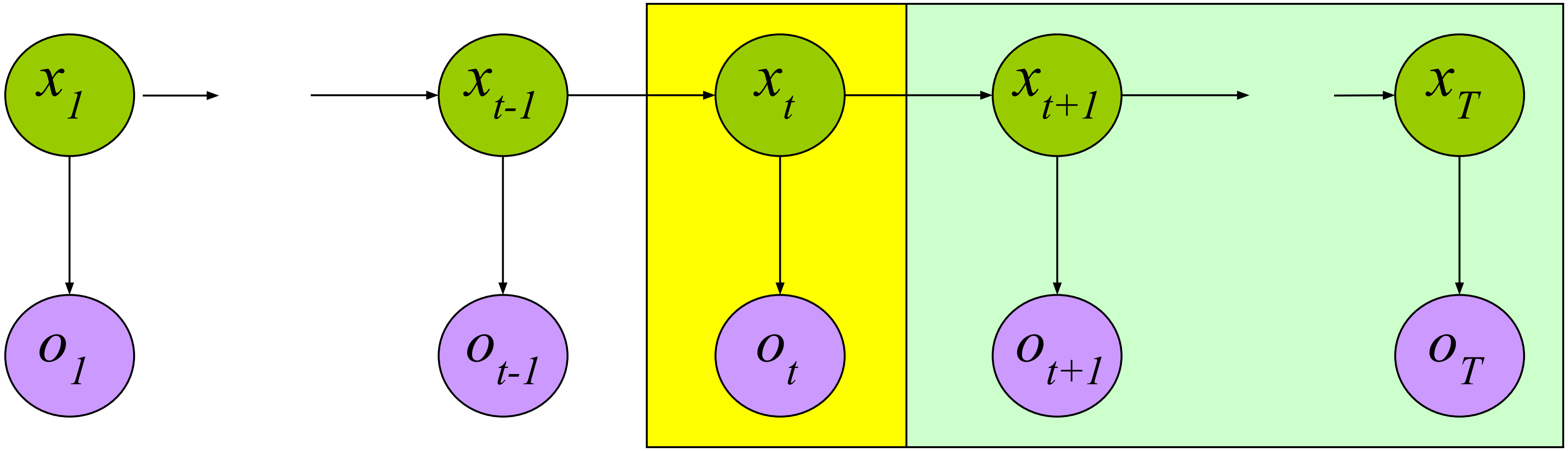
$$= \sum_{i=1\ldots N} P(o_1 \ldots o_t, x_t = i, x_{t+1} = j) P(o_{t+1} \mid x_{t+1} = j)$$

$$= \sum_{i=1\ldots N} P(o_1 \ldots o_t, x_{t+1} = j \mid x_t = i) P(x_t = i) P(o_{t+1} \mid x_{t+1} = j)$$

$$= \sum_{i=1\ldots N} P(o_1 \ldots o_t, x_t = i) P(x_{t+1} = j \mid x_t = i) P(o_{t+1} \mid x_{t+1} = j)$$

$$= \sum_{i=1\ldots N} \alpha_i(t) a_{ij} b_{jo_{t+1}}$$

Backward Procedure



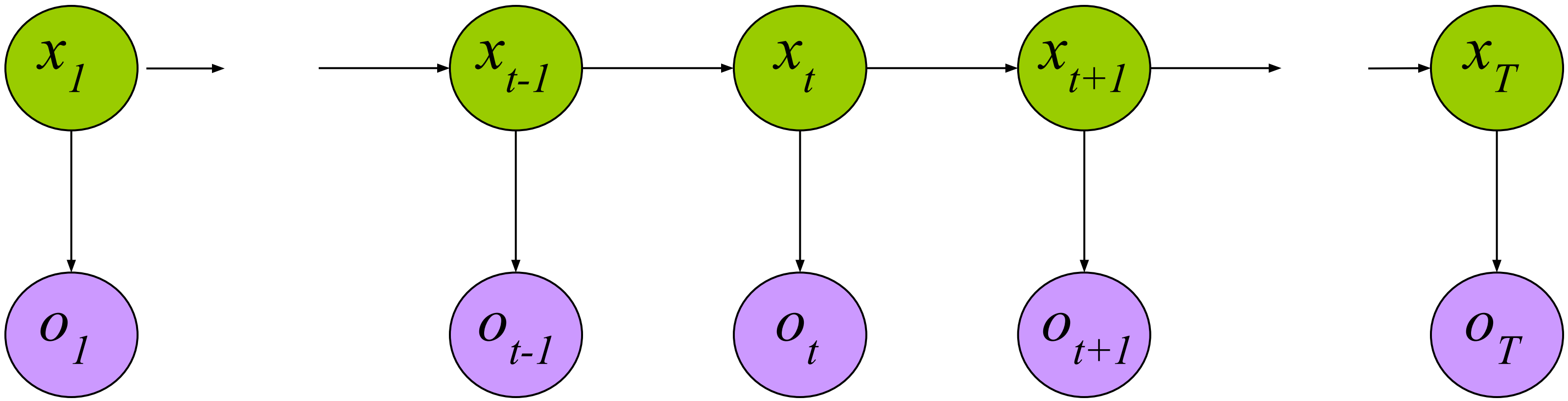
$$\beta_i(T+1) = 1$$

$$\beta_i(t) = P(o_t \dots o_T \mid x_t = i)$$

$$\beta_i(t) = \sum_{j=1\dots N} a_{ij} b_{io_t} \beta_j(t+1)$$

Probability of the rest of the states given the first state

Decoding Solution

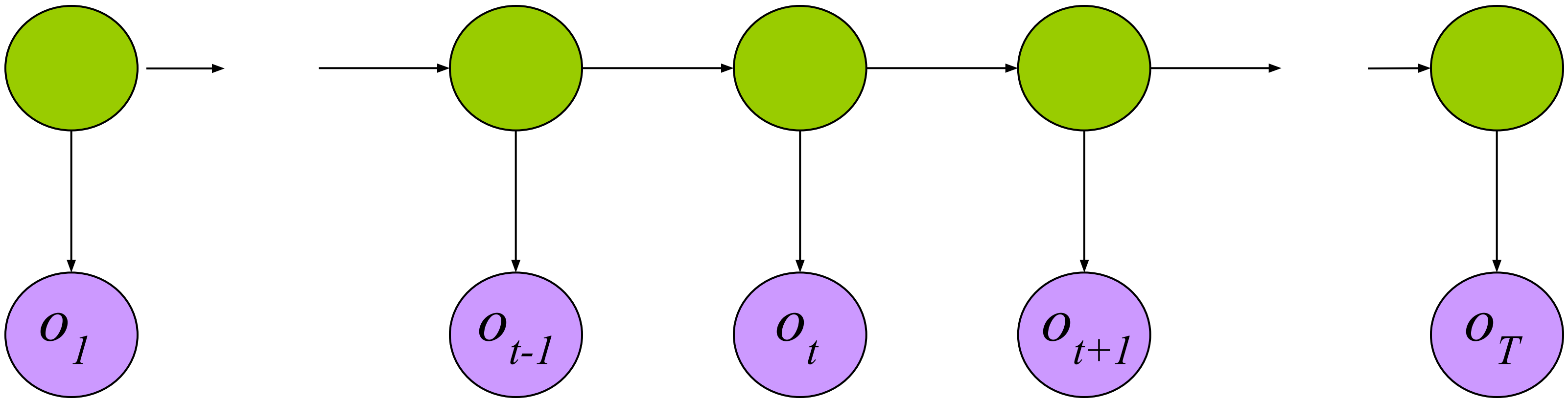


$$P(O \mid \mu) = \sum_{i=1}^{N} \alpha_i(T)$$ **Forward Procedure**

$$P(O \mid \mu) = \sum_{i=1}^{N} \pi_i \beta_i(1)$$ **Backward Procedure**

$$P(O \mid \mu) = \sum_{i=1}^{N} \alpha_i(t) \beta_i(t)$$ **Combination**

Best State Sequence

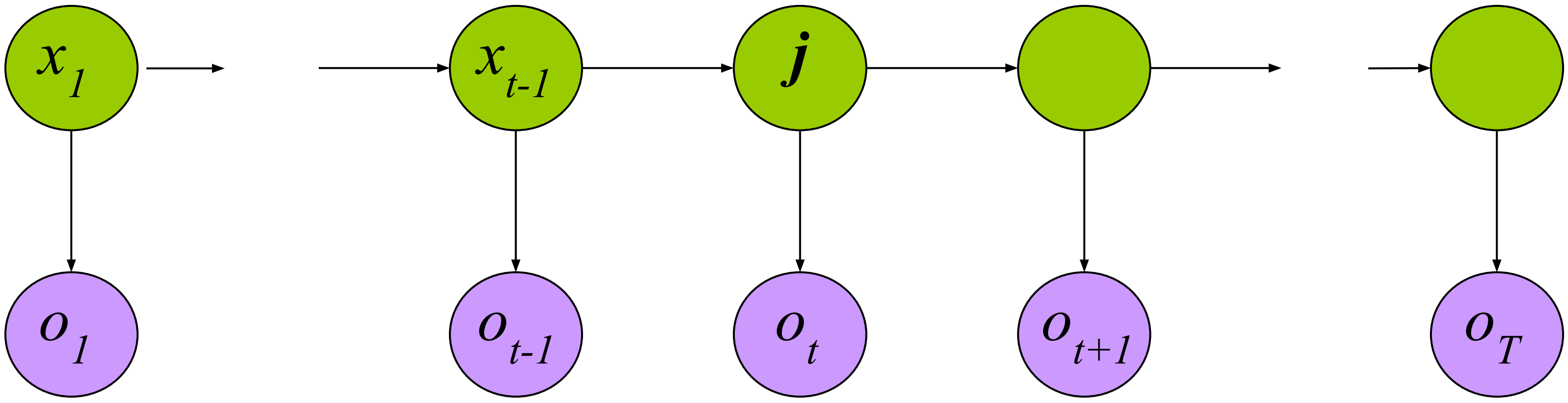


- Find the state sequence that best explains the observations

- **Viterbi** algorithm

- $\arg\max_X P(X \mid O)$

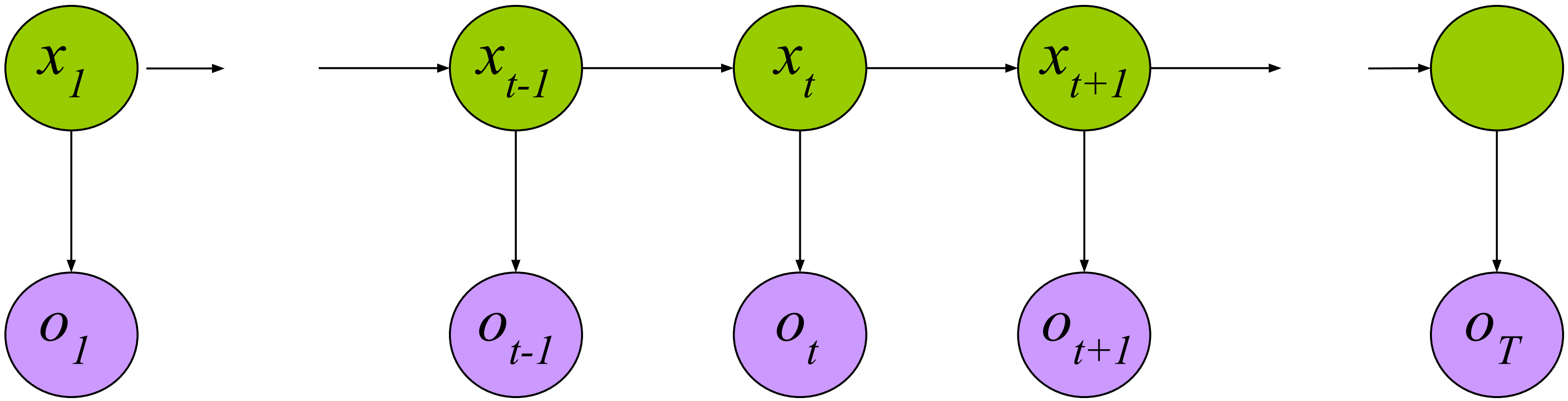
Viterbi Algorithm



$$\delta_j(t) = \max_{x_1 \ldots x_{t-1}} P(x_1 \ldots x_{t-1}, o_1 \ldots o_{t-1}, x_t = j, o_t)$$

The state sequence which maximizes the probability of seeing the observations to time t-1, landing in state j, and seeing the observation at time t

Viterbi Algorithm



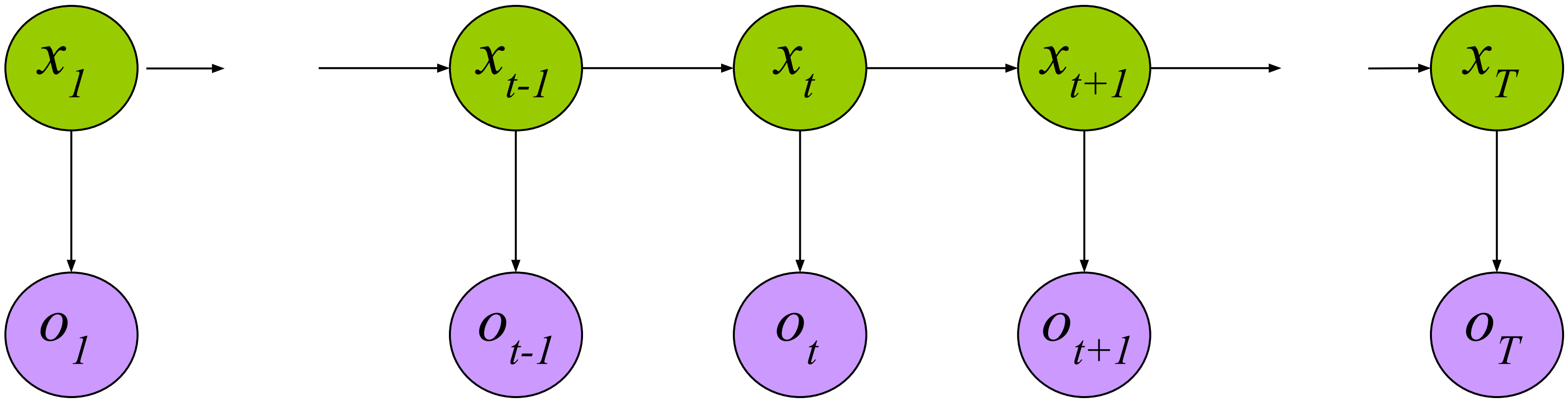
$$\delta_j(t) = \max_{x_1 \ldots x_{t-1}} P(x_1 \ldots x_{t-1}, o_1 \ldots o_{t-1}, x_t = j, o_t)$$

$$\delta_j(t+1) = \max_i \delta_i(t) a_{ij} b_{jo_{t+1}}$$

$$\psi_j(t+1) = \arg\max_i \delta_i(t) a_{ij} b_{jo_{t+1}}$$

Recursive Computation

Viterbi Algorithm



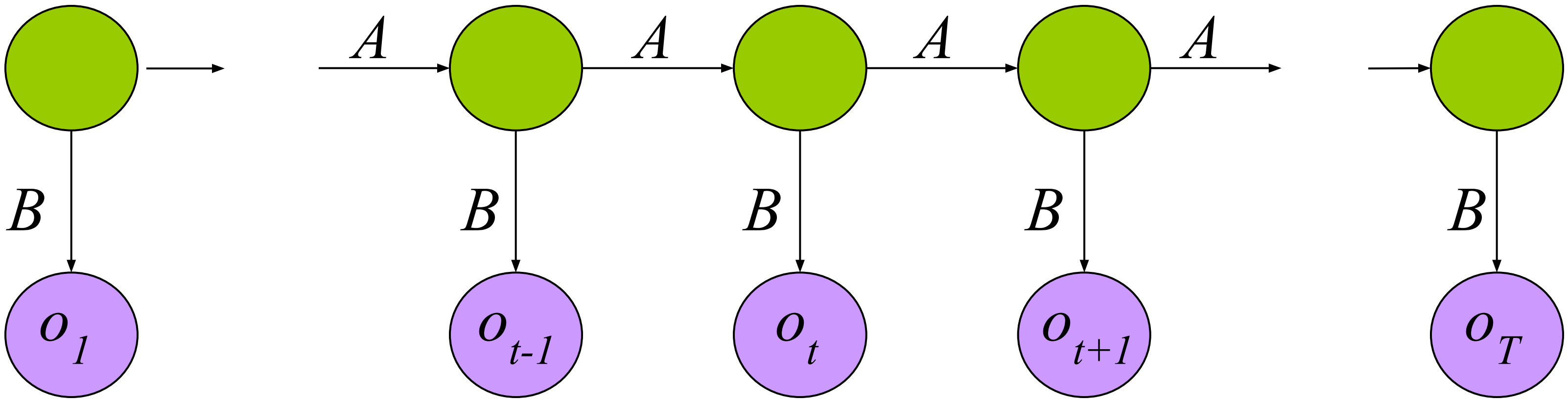
$$\hat{X}_T = \arg\max_i \delta_i(T)$$

$$\hat{X}_t = \psi_{\hat{X}_{t+1}}(t+1)$$

$$P(\hat{X}) = \arg\max_i \delta_i(T)$$

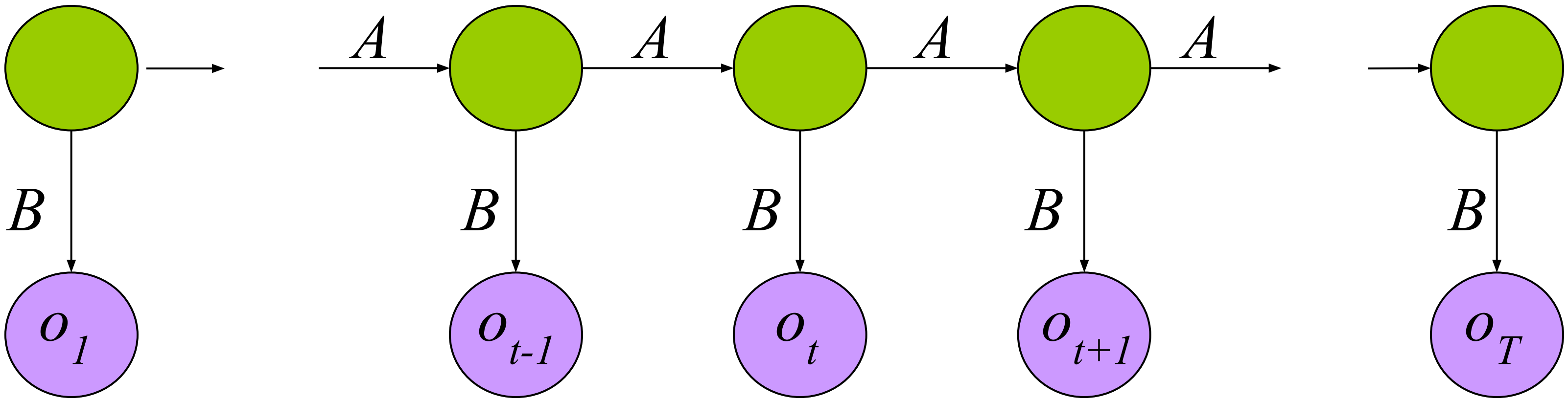
Compute the most likely state sequence by working backwards

Parameter Estimation



- Given an observation sequence, find the model that is most likely to produce that sequence.
- No analytic method
- Given a model and observation sequence, update the model parameters to better fit the observations.

Parameter Estimation



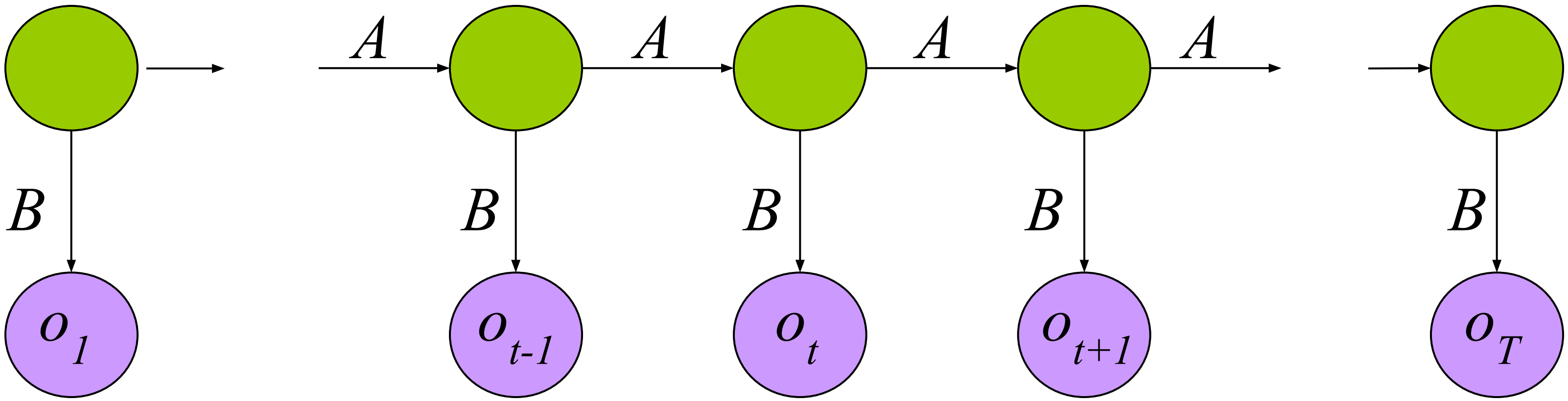
$$p_t(i,j) = \frac{\alpha_i(t) a_{ij} b_{jo_{t+1}} \beta_j(t+1)}{\sum_{m=1\ldots N} \alpha_m(t)\beta_m(t)}$$

Probability of traversing an arc

$$\gamma_i(t) = \sum_{j=1\ldots N} p_t(i,j)$$

Probability of being in state *i*

Parameter Estimation



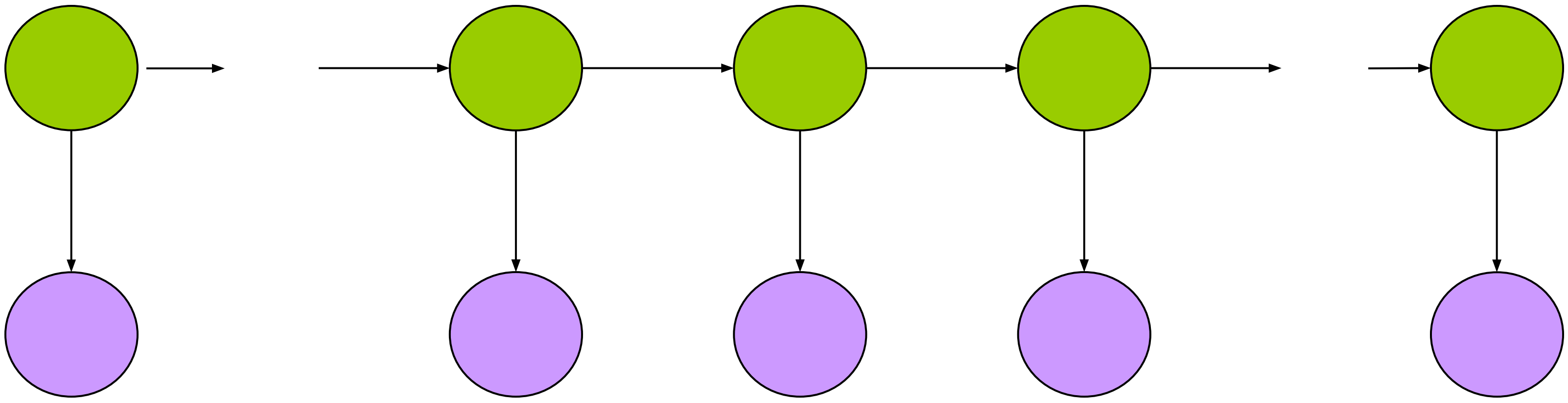
$$\hat{\pi}_i = \gamma_{\mathrm{i}}(1)$$

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T} p_t(i,j)}{\sum_{t=1}^{T} \gamma_i(t)}$$

$$\hat{b}_{ik} = \frac{\sum_{\{t:o_t=k\}} \gamma_t(i)}{\sum_{t=1}^{T} \gamma_i(t)}$$

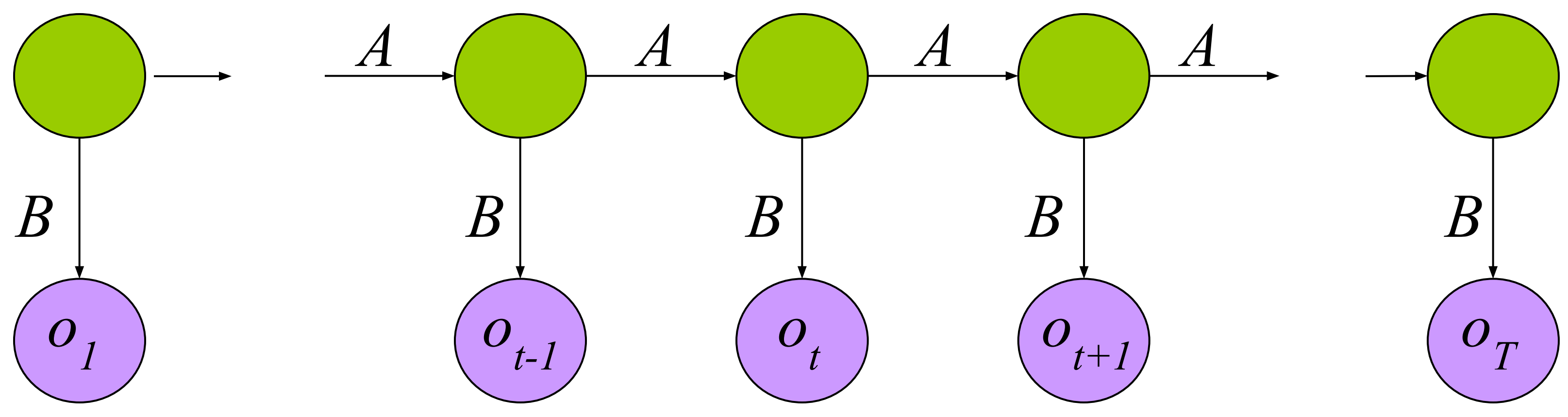
Now we can compute the new estimates of the model parameters.

HMM Applications



- Generating parameters for n-gram models
- Tagging speech
- Speech recognition

The Most Important Thing



We can use the special structure of this model to do a lot of neat math and solve problems that are otherwise not solvable.

References

- Foundations of Statistical Natural Language Processing · Christopher D. Manning and Hinrich Schütze · Chapter 9: Markov Models.

- R.O. Duda, P.E. Hart, and D.G. Stork, Pattern Classification, New York: John Wiley & Sons, 2001.