Hidden Markov Models (HMM)

ECE/CS 498 DS U/G

Lecture 17: Introduction to Hidden Markov Models

Ravi K. Iyer

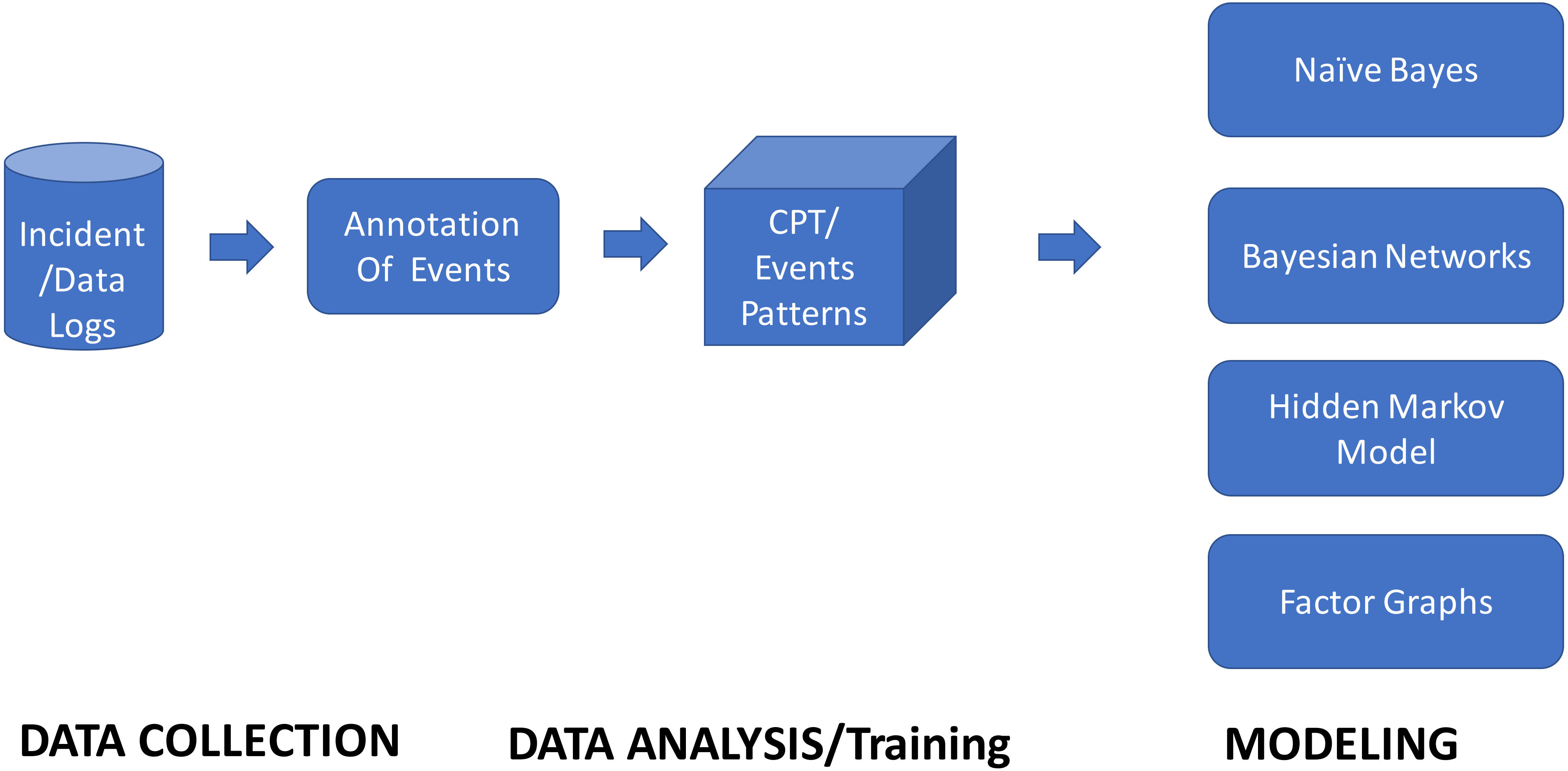
Dept. of Electrical and Computer Engineering

University of Illinois at Urbana Champaign

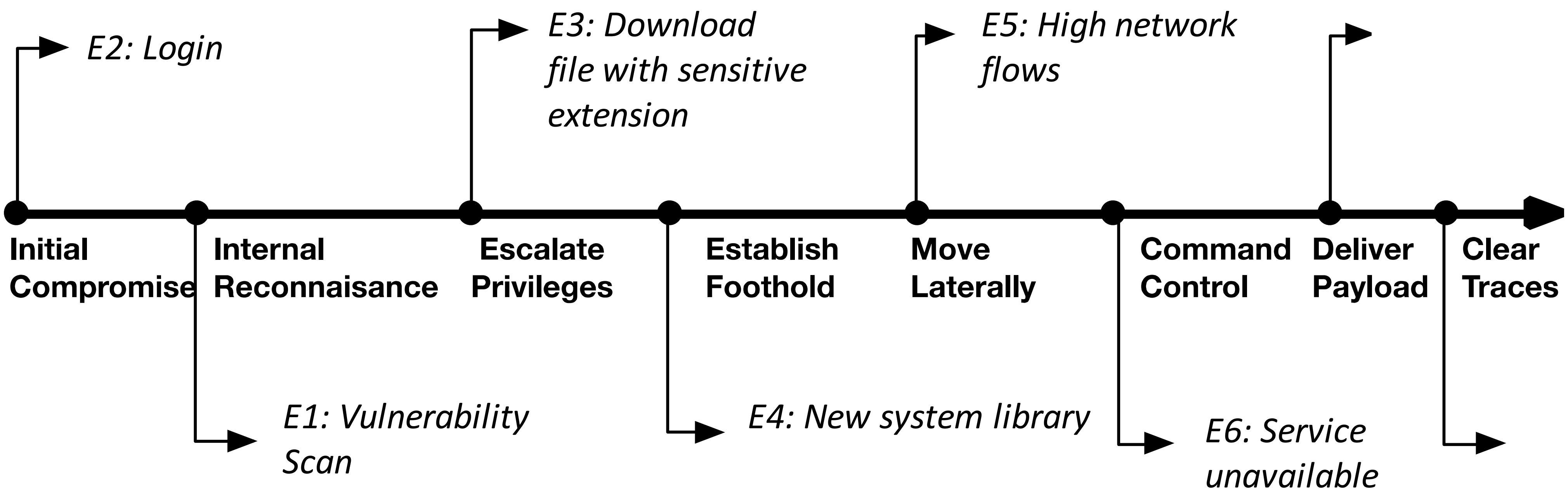
Announcements

- Course Timeline
  - Today 3/25: Introduction to Hidden Markov Models (HMMs)
  - Mon 3/30: HMMs Continued, **ICA 4**
    - Will try to use Zoom breakout rooms to coordinate these
- MP 2 Timeline
  - Checkpoint 1.5 due **tonight March 25 @ 11:59 PM**
    - Submit via https://forms.gle/88Wk6QtxvaWsFChX6
  - Final Checkpoint due on **Monday March 30 @ 11:59 PM on Compass2G**
- Final Project
  - Make sure to review feedback from proposals
  - Progress report 1 due **Friday March 27 @ 11:59 PM** on Compass2G
    - We are expecting reasonable progress from the time of the project proposals…

Overview of PGM Data Analytics/Modeling Process



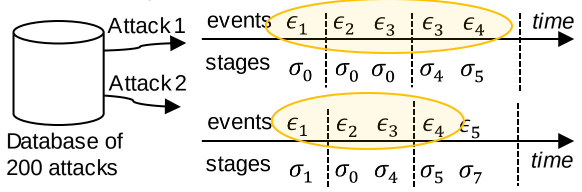
An Application in Security Data Analytics
Individual components of an attack as attack progresses



Attack stages for the credential stealing attack

Annotation and extracting patterns in past attacks

a) Annotated events and attack stages in a pair of attacks



annotations

b) Event-stage annotation table for the attack pair (Attack 1 and Attack 2)

Event	**Attack stage**
$\{\epsilon_1\}$	$\{\sigma_0 \vert \sigma_1\}$
$\{\epsilon_2\}$	$\{\sigma_0\}$
$\{\epsilon_3\}$	$\{\sigma_4\}$
$\{\epsilon_4\}$	$\{\sigma_5\}$
$\{\epsilon_5\}$	$\{\sigma_7\}$

c) Example patterns, stages, probabilities, and significance learned from the attack pair

Pattern	Attack stages	Probability in past attacks	Significance (p-value)
$[\epsilon_1, \epsilon_3, \epsilon_4]$	$[\sigma_1, \sigma_4, \sigma_5]$	q_a	p_a
$[\epsilon_1]$	$[\sigma_0 \vert \sigma_1]$	q_b	p_b

…

Naïve Bayes

Bayesian Network

Dynamic Bayesian Network

Hidden Markov Model

Factor Graphs

OFFLINE ANNOTATION ON PAST ATTACKS

OFFLINE LEARNING OF PATTERNS

PROBABILISTIC GRAPHICAL MODELS

Observed Security events

Unknown attack stages

Factor function

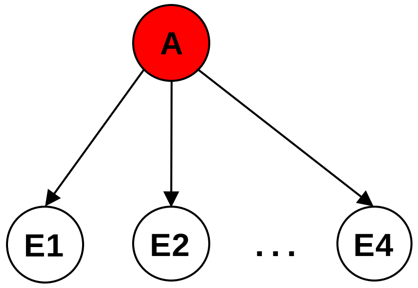
* Attack detected and stopped before the system misuse

ϵ_1 vulnerability scan
ϵ_2 login
ϵ_3 sensitive_uri
ϵ_4 new_library

σ_0 benign
σ_1 discovery
σ_4 privilege escalation
σ_5 persistence

Note: ϵ_i *is the corresponding value of an event* E_t

Modeling the credential stealing attack using Naïve Bayes vs. Bayesian Network

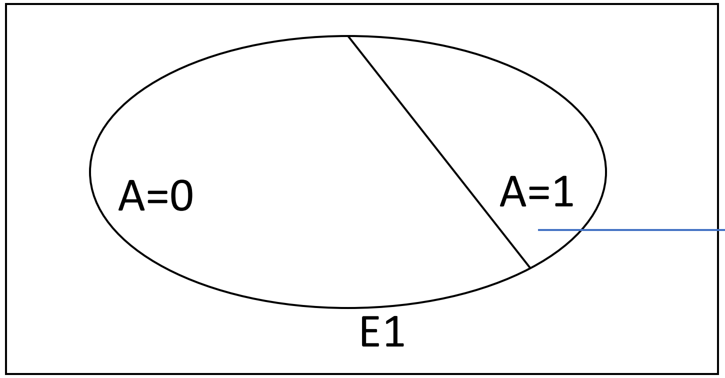


Naïve Bayes

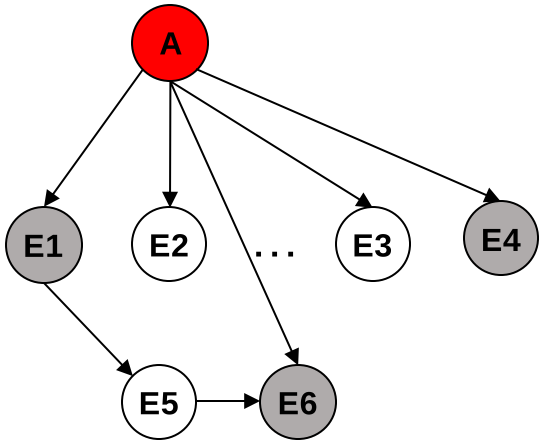
$$P(A, E_1, E_2, \dots, E_4) = P(A) \prod_i P(E_i|A)$$

Is (E1, E2, ..., E4) represents Benign activity?

$$[P(\mathrm{E}_1|\mathrm{A} = \mathrm{Benign}) \dots . P(E_4|\mathrm{A} = \mathrm{Benign})]P(\mathrm{A} = \mathrm{Benign}) > [P(E_1|A = Attack) \dots P(E_4|\mathrm{A} = \mathrm{Attack})]P(\mathrm{A} = \mathrm{Attack})$$



$P(E_1|A = 1)$



Bayesian Network

Joint Distribution: $P(E_1, E_2, \dots, E_n, A) = P(A) \prod_{i=1}^{n} P(E_i|parents(E_i))$

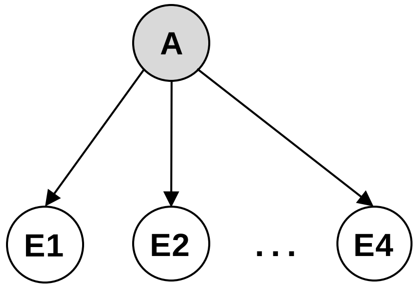
Hypothesis:

$$P(A = attack|E_1, E_4\ E_6) = ?$$

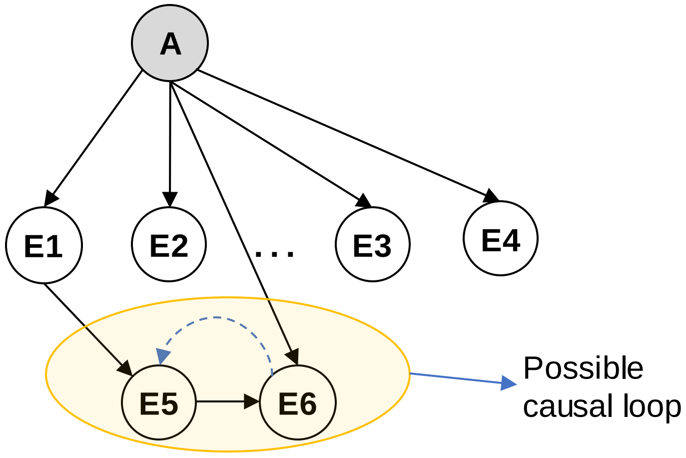
$$P(A = benign|E_1, E_4\ E_6) = ?$$

ID	Description
A	Attack
E1	Vulnerability scan
E2	Login
E3	Download file with sensitive extension
E4	New system library
E5	High network flows
E6	Service unavailable

Modeling the credential stealing attack using Naïve Bayes vs. Bayesian Network



Naïve Bayes



Bayesian Network

ID	Description
A	Attack
E1	Vulnerability scan
E2	Login
E3	Download file with sensitive extension
E4	New system library
E5	High network flows
E6	Service unavailable

Model assumptions

1. All events share the same parent variable
2. All events are conditionally independent

Advantage:

Simplify calculation of posterior probability on A

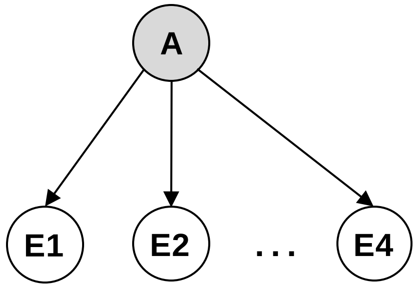
Model assumptions

1. An event can be preceded (causal) by another event
2. There is no cycle in the network

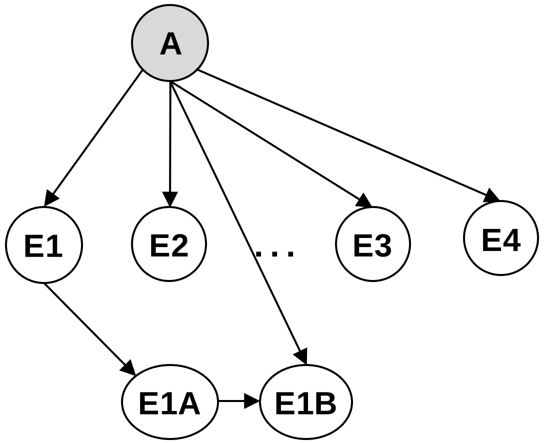
Disadvantage

Explicitly assume causal relationships
(Causality may not be clear from the data)
For complicated attacks, causal loops may form and render the BN invalid

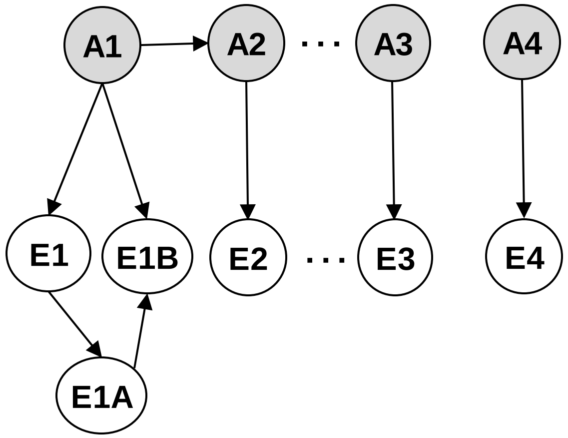
Modeling the credential stealing attack using Naïve Bayes vs. Bayesian Network



Naïve Bayes

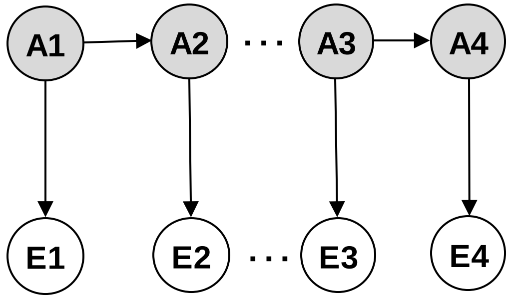


Bayesian Network



Dynamic Bayesian Network

- When we consider the time evolution of the BN each variable in each timestep together, e.g., *t* and *t+1,* we have a Dynamic Bayesian Network that captures the first-order dependency --> referred to as the Markov Property
- This concept can be extended to higher order dependencies e.g on , t-2, t-3, … and is called a higher-order Markov property, e.g., 2nd or 3rd Markov property.

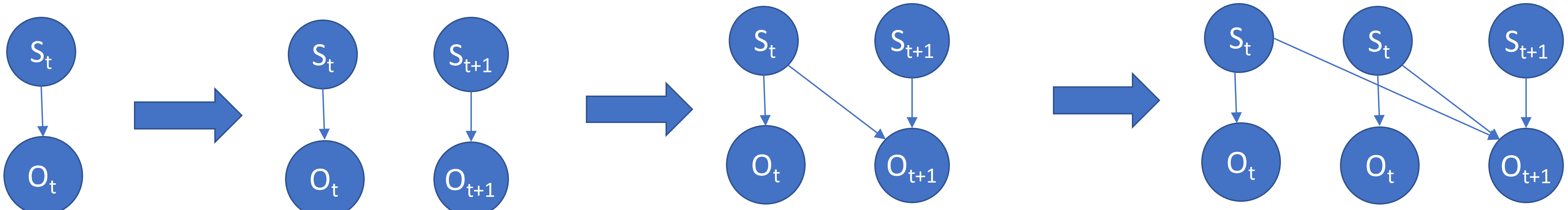


$$P(A_1, E_1, \dots, A_n, E_n) = P(A_1)P(E_1|A_1) \dots P(E_{t+1}|A_{t+1})P(A_{t+1}|A_t)$$

Hidden Markov Model

Dynamic Bayesian Networks

- We have considered BNs with a static set of random variables, e.g., two variables: only one measurement variable and one state variable of the system.
- In reality, data is often time series in which each time step *t* has one measurement variable O_t and one state variable S_t . Thus, the number of random variables is proportional with the number of timesteps.
- Without correlating the random variables in each timestep, we have T disconnected BNs
- When we correlate each variable in each timestep together, e.g., *t* and *t+1,* we have a Dynamic Bayesian Network that captures the first-order Markov property.
- This concept can be extended for t, t+1, t+2, … and is called a higher-order Markov property, e.g., 2^{nd} or 3^{rd}



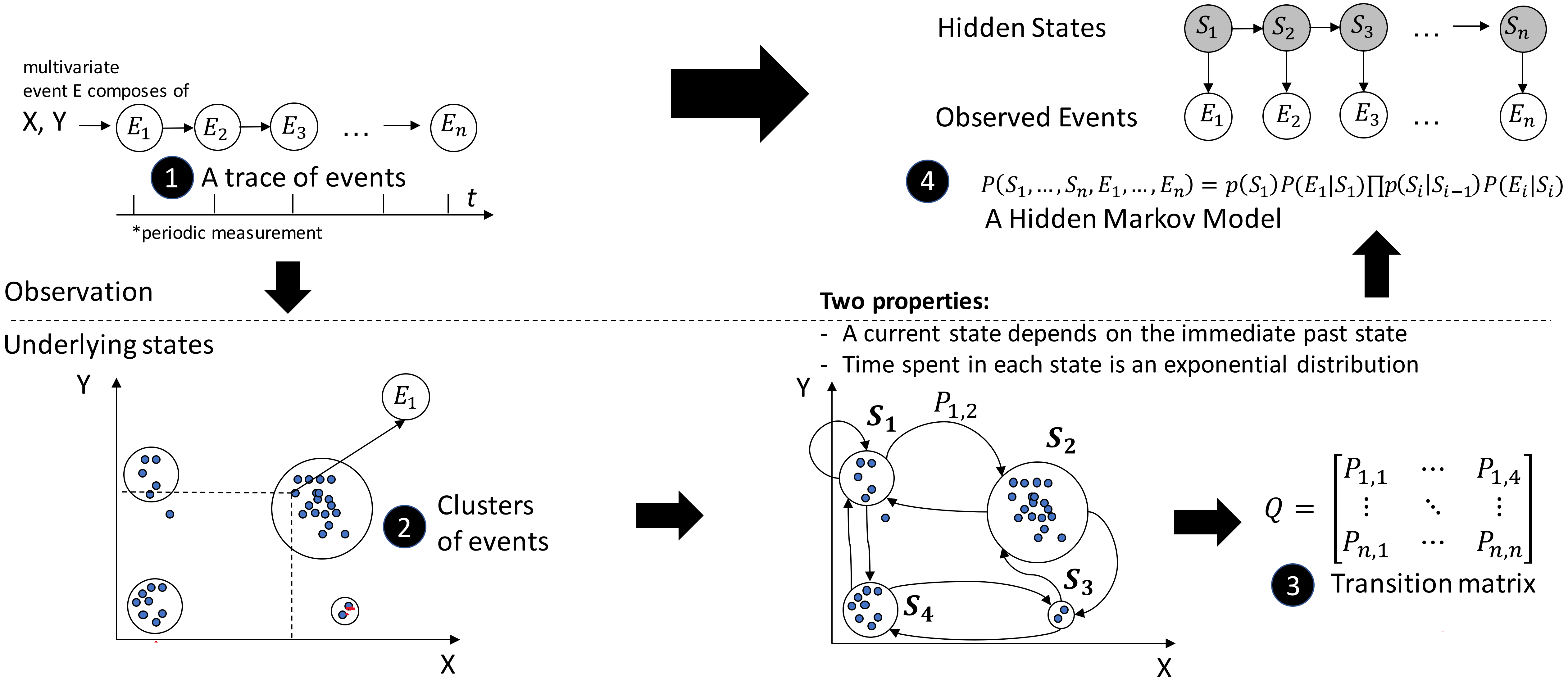
$P(S_t, O_t) = P(S_t)P(O_t|S_t)$

$P(S_t, O_t) = P(S_t)P(O_t|S_t)$

$P(S_{t+1}, O_{t+1}) = P(S_{t+1})P(O_{t+1}|S_{t+1})$

$P(S_t, S_{t+1}, O_t, O_{t+1}) = P(S_t)P(O_t|S_t)P(O_{t+1}|S_t, S_{t+1})P(S_{t+1})$

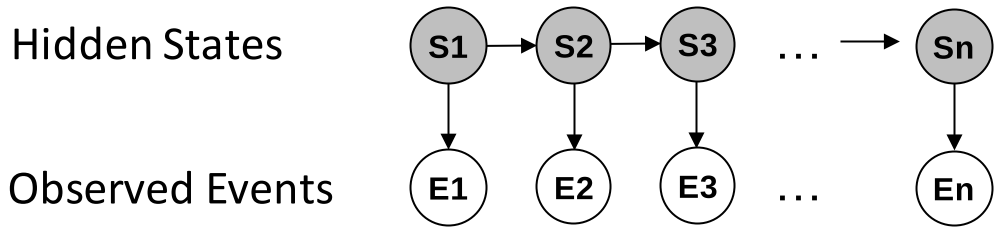
From a trace of events to a Hidden Markov Model



Hidden Markov Models

Model assumptions
An observation depends on its hidden state
A state variable only depends on the immediate previous state (Markov assumption)
The future observations and the past observations are conditionally independent given the current hidden state

Advantages:
HMM can model sequential nature of input data (future depends on the past)
HMM has a linear-chain structure that clearly separates system state and observed events.



$$P(S_1, \dots, S_n, E_1, \dots, E_n) = p(S_1)P(E_1|S_1)\prod p(S_i|S_{i-1})P(E_i|S_i)$$

A Hidden Markov model on observed events and system states

Markov Model

- Consider a system which can occupy one of *N* discrete *states* or *categories*

$$x_t \in \{1, 2, \dots, N\} \longrightarrow \text{state at time } t$$

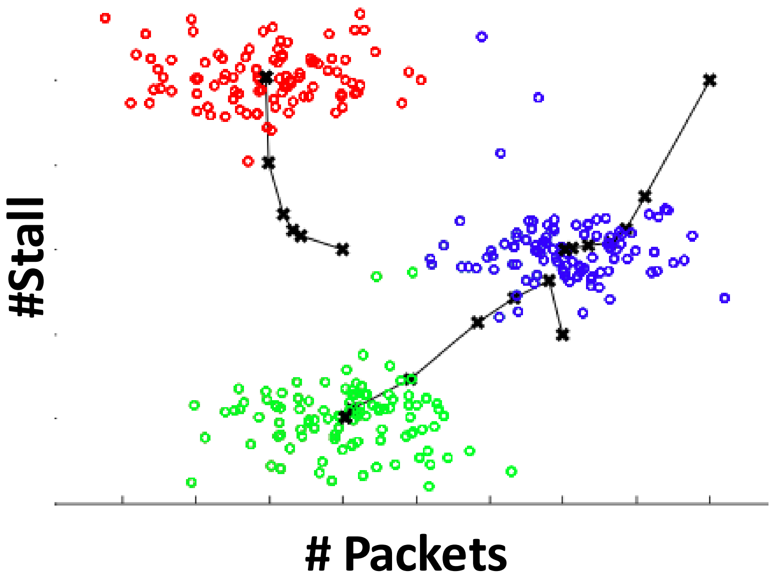
- We are interested in *stochastic* systems, in which state evolution is random
- Any *joint* distribution can be factored into a series of *conditional* distributions:

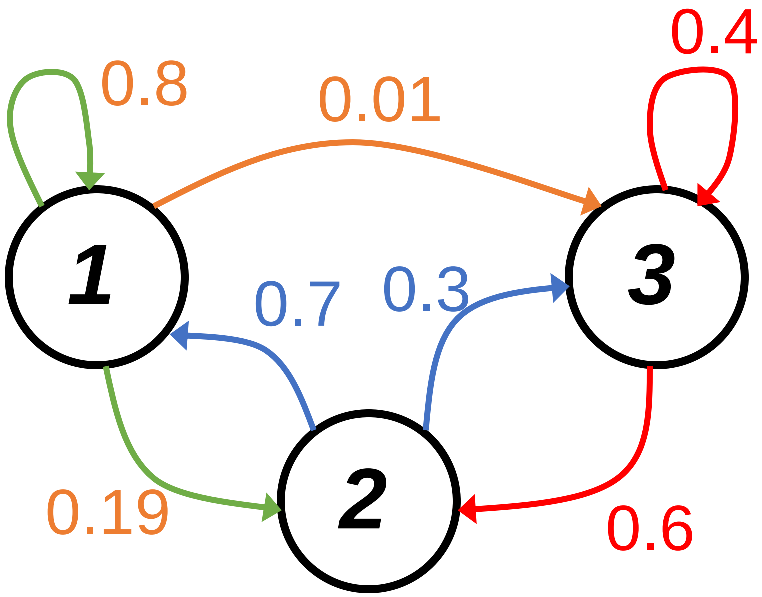
$$p(x_0, x_1, \dots, x_T) = p(x_0) \prod_{t=1}^{T} p(x_t \mid x_0, \dots, x_{t-1})$$

- For a *Markov* process, the next state depends only on the current state:

$$p(x_{t+1} \mid x_0, \dots, x_t) = p(x_{t+1} \mid x_t)$$

State Transition Diagrams (Packet Stalls in a Network)



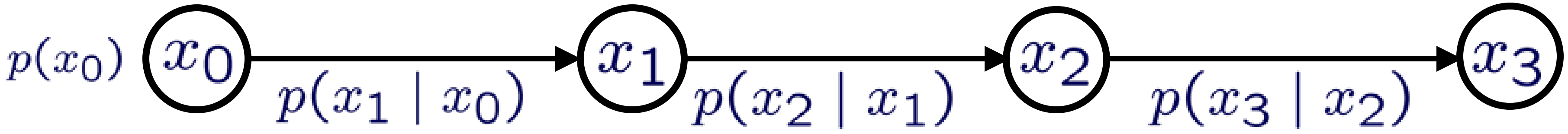


$$q_{ij} \triangleq p(x_{t+1} = i \mid x_t = j)$$

- Think of a particle randomly following an arrow at each discrete time step
- Most useful when *N* (# of States)small, and *Q* (TransProb Matrix)*sparse*

Markov Chains: Graphical Models

$$p(x_0, x_1, \ldots, x_T) = p(x_0) \prod_{t=1}^{T} p(x_t \mid x_{t-1})$$



$$Q = \begin{bmatrix} 0.80 & 0.7 & 0.0 \\ 0.19 & 0.0 & 0.6 \\ 0.01 & 0.3 & 0.4 \end{bmatrix}$$

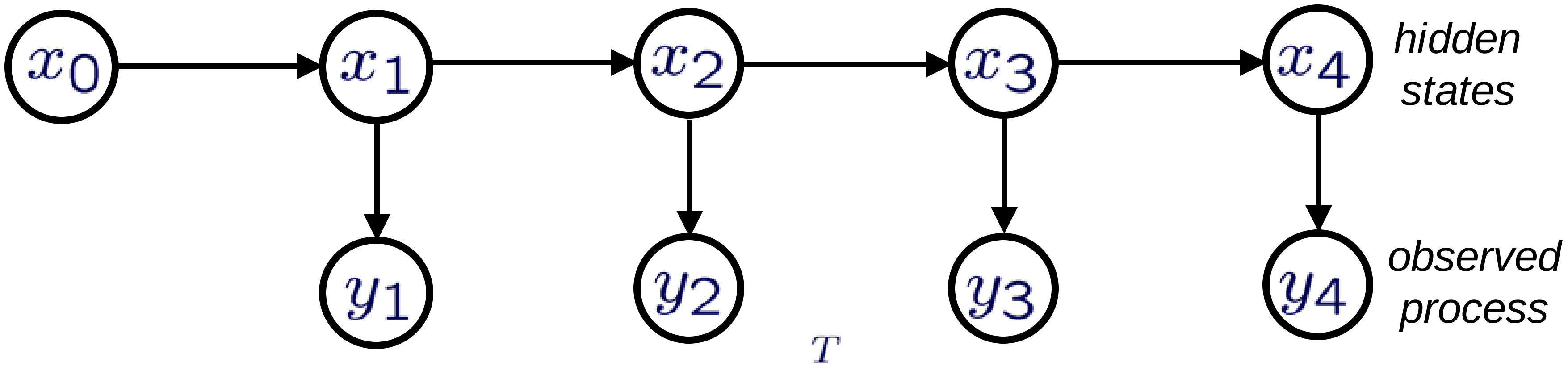
Constraints on valid transition matrices:

$q_{ij} \geq 0$, $\sum_{i=1}^{N} q_{ij} = 1$ for all j

$$q_{ij} \triangleq p(x_{t+1} = i \mid x_t = j)$$

Hidden Markov Models (Packet Stall Example Cont'd)

- Stall exists due to congestion
- Not directly measurable at runtime (hidden)
- Motivates *hidden Markov models (HMM):*



$$p(x_0, x_1, \ldots, x_T) = p(x_0) \prod_{t=1}^{T} p(x_t \mid x_{t-1}) p(y_t \mid x_t)$$

Given x_t, previous observations impact future observations

$$p(y_t, y_{t+1}, \cdots \mid x_t, y_{t-1}, y_{t-2}, \ldots) = p(y_t, y_{t+1}, \cdots \mid x_t)$$

State Transition Matrices

- A *stationary* Markov chain with *N* states is described by an *NxN* *transition matrix:*

$$Q = \begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{21} & q_{22} & q_{23} \\ q_{31} & q_{32} & q_{33} \end{bmatrix}$$

$$q_{ij} \triangleq p(x_{t+1} = i \mid x_t = j)$$

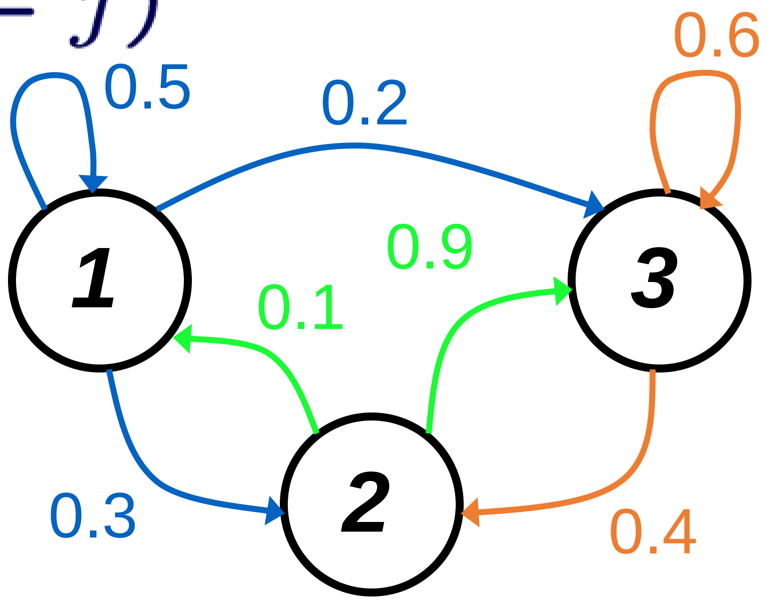
- Constraints on valid transition matrices:

$$q_{ij} \geq 0 \qquad\qquad \sum_{i=1}^{N} q_{ij} = 1 \quad \text{for all } j$$

State Transition Diagrams(Another Example)

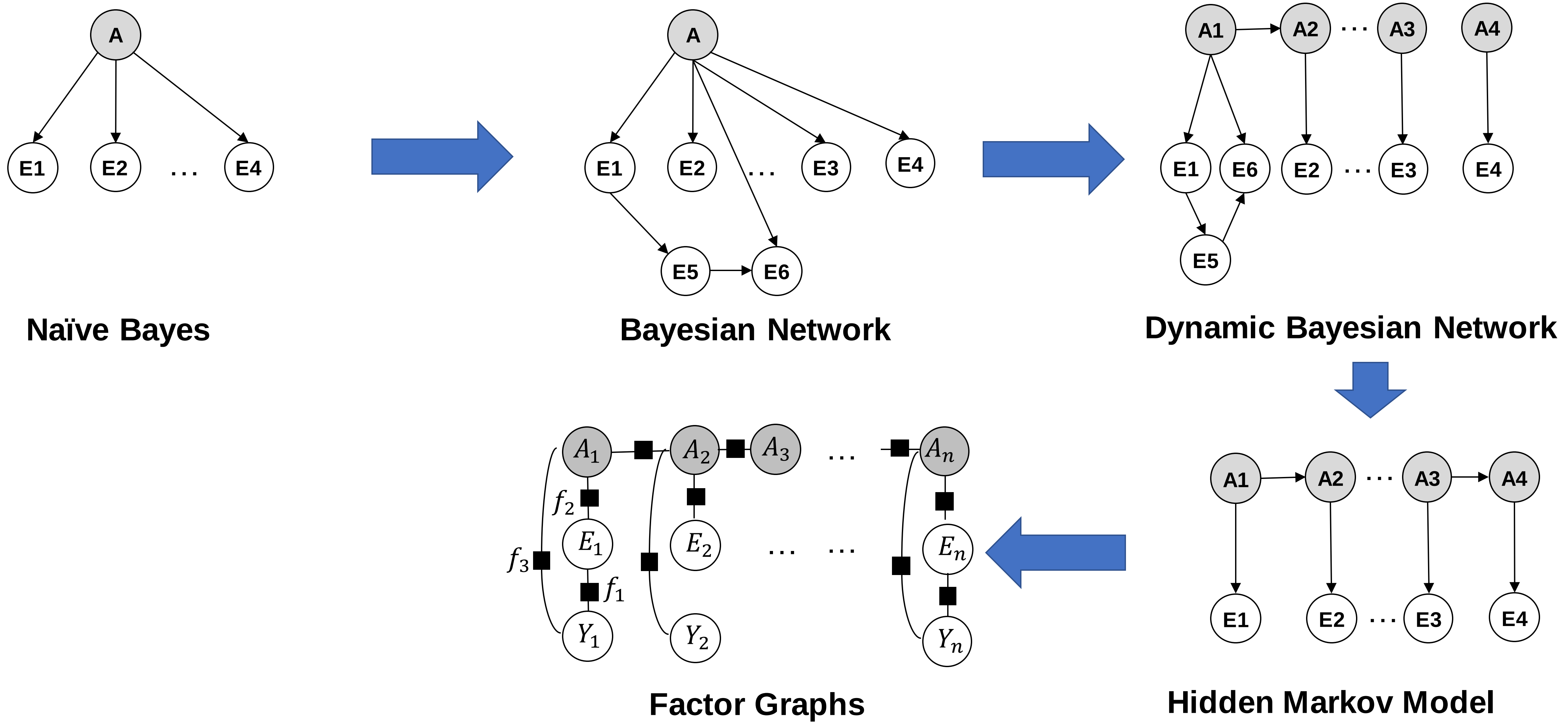
$$q_{ij} \triangleq p(x_{t+1} = i \mid x_t = j)$$

$$Q = \begin{bmatrix} 0.5 & 0.1 & 0.0 \\ 0.3 & 0.0 & 0.4 \\ 0.2 & 0.9 & 0.6 \end{bmatrix}$$



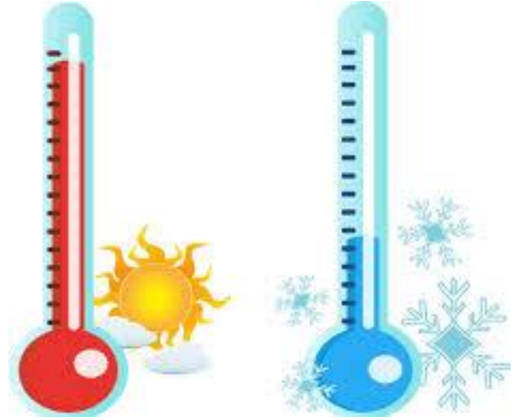
- Think of a particle randomly following an arrow at each discrete time step
- Most interesting when *Q* *sparse*

Modeling the credential stealing attack using Naïve Bayes vs. Bayesian Network



HMM Motivating Example: Paleontological Temperature Model

- Want to determine the average temperature at a particular place on earth over a sequence of years in the distant past
- Only annual average temperatures -- hot **(H)** and cold (**C**)
  - Probability of a hot year followed by another hot year is 0.7, and the probability of a cold year followed by another cold year is 0.6, independent of the temperature in prior years
- Correlation between the size of tree growth rings and temperature
  - Three different ring sizes, small (**T**), medium (**D**), and large (**L**)
- Assume that probability values from current period held in paleontological period too
- Determine the most likely temperature state in past years
  - Can't directly observe the temperature in the past
  - We can observe the size of tree rings – can this information be used?



H C

T D L

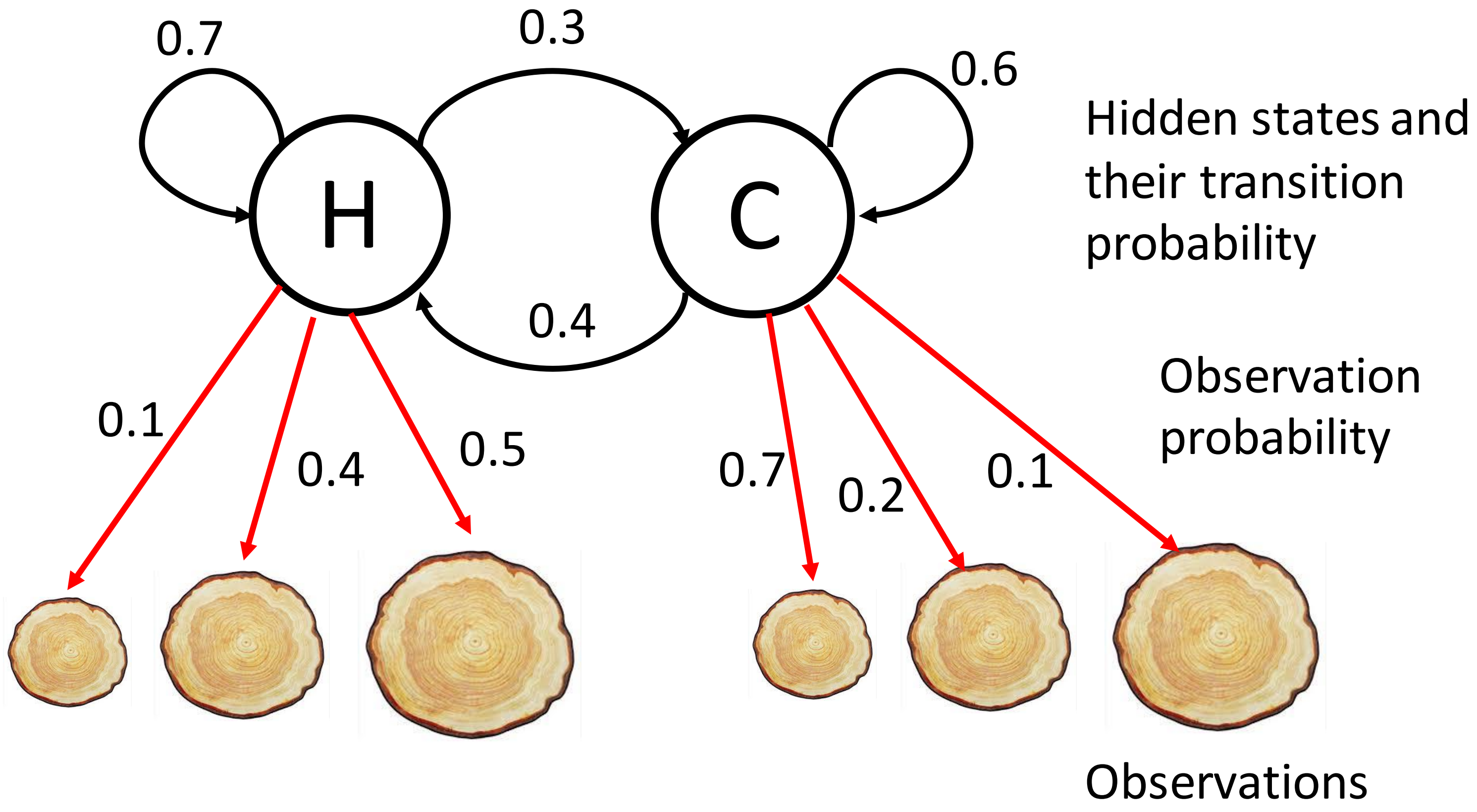
Tree ring size

Source: http://www.math.unl.edu/~sdunbar1/ProbabilityTheory/Lessons/HiddenMarkovModels/Examples/examples.html

Paleontological Temperature Model

- State space of hidden states: $S = \{H, C\}$
- State space of observations: $E = \{T, D, L\}$
- Transition probability matrix: A
- Observation Matrix: B
- Initial distribution for the hidden states: π

Given by an oracle



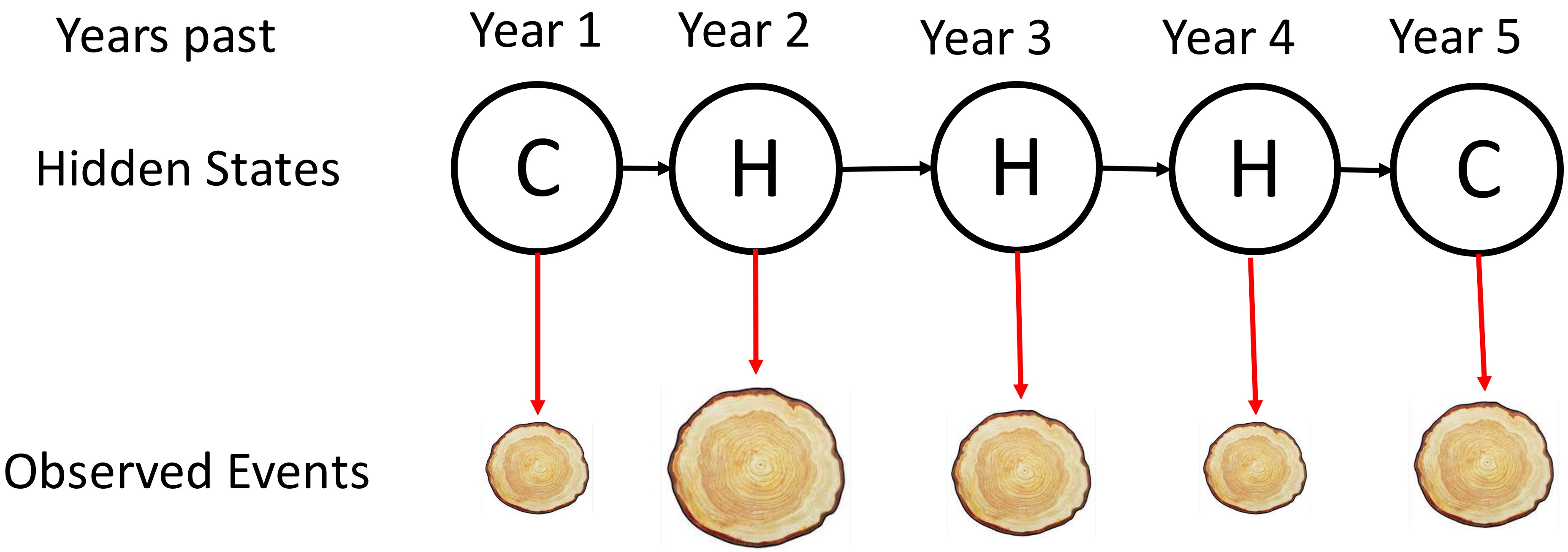
$$A = \begin{array}{c c} & \begin{array}{cc} H & C \end{array} \\ \begin{array}{c} H \\ C \end{array} & \begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix} \end{array}$$

$$B = \begin{array}{c c} & \begin{array}{ccc} T & D & L \end{array} \\ \begin{array}{c} H \\ C \end{array} & \begin{bmatrix} 0.1 & 0.4 & 0.5 \\ 0.7 & 0.2 & 0.1 \end{bmatrix} \end{array}$$

$$\pi = \begin{array}{c} \begin{array}{cc} H & C \end{array} \\ \begin{bmatrix} 0.5 & 0.5 \end{bmatrix} \end{array}$$

Paleontological Temperature Model

Example sequence with 5 observations



Determine the sequence of hidden states

Hidden Markov Models

Model

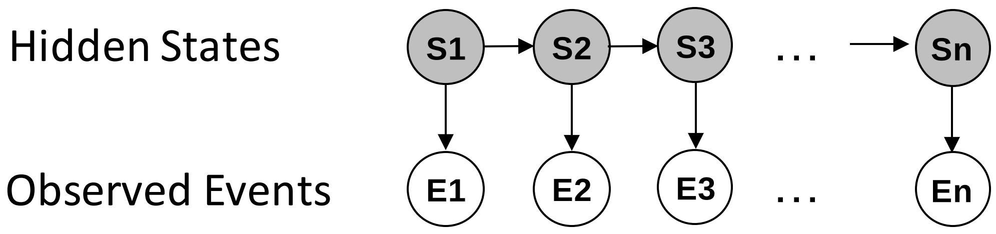
- Set of hidden states $\boldsymbol{S} = \{\boldsymbol{\sigma_1}, \dots, \boldsymbol{\sigma_N}\}$
- Set of observable events $\boldsymbol{E} = \{\boldsymbol{\epsilon_1}, \dots, \boldsymbol{\epsilon_M}\}$
- Transition probability matrix $\boldsymbol{A}$
- Observation matrix $\boldsymbol{B}$
- Initial distribution of hidden states $\boldsymbol{\pi}$

Model assumptions

- An observation depends on its hidden state
- A state variable only depends on the immediate previous state (Markov assumption)
- The future observations and the past observations are conditionally independent given the current hidden state

Advantages:

- HMM can model sequential nature of input data (future depends on the past)
- HMM has a linear-chain structure that clearly separates system state and observed events.



A Hidden Markov model on observed events and system states

$$P(S_1, \dots, S_n, E_1, \dots, E_n) = P(S_1)P(E_1|S_1)\prod_{i=2}^{n} P(S_i|S_{i-1})P(E_i|S_i)$$

Inference question – Paleontological Temperature

Given the sequence of 5 observations T, L, D, T, D and the model (A, B, π), how do we choose a corresponding state sequence $S_1, S_2, \ldots, S_n$ which is optimal in some meaningful sense (i.e., best explains the observations) where $S_t \in \{H, C\}$?

A simpler question: Given the sequence of 5 observations T, L, D, T, D and the model (A, B, π), which of the two is more probable eg., $S_3 = H$ or $S_3 = C$?

HMM Security Example

- Suppose you are a security expert monitoring the NCSA system
- By monitoring the system events, you want to say whether the system is safe or not
  - System's safety is a hidden state
  - Events are observed
  - Events are related to the safety of the system
- Is the system safe?
  - **HMM** to the rescue!

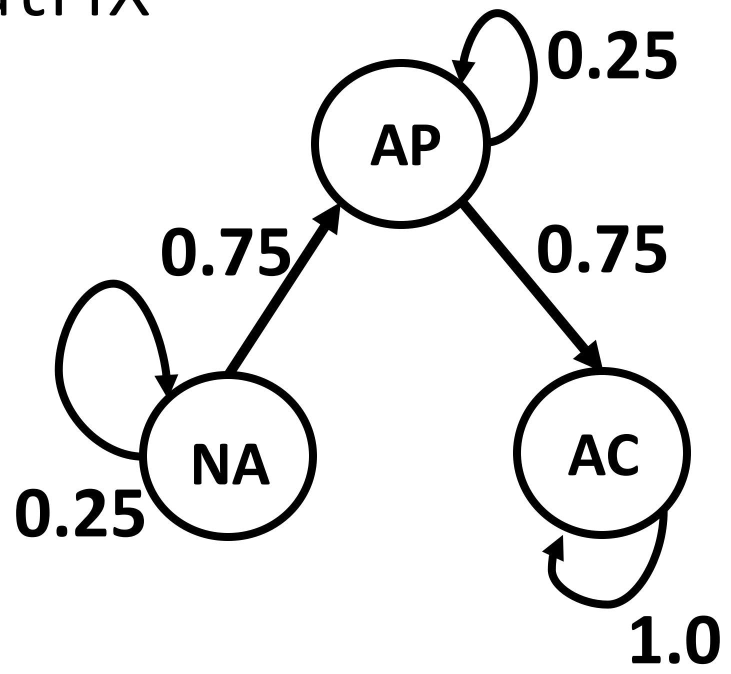
Security Example: Transition Matrix

Transition matrix (A)

The system has three distinct security states –

(a) No Attack **(NA)**,
(b) Attack in Progress **(AP),** and
(c) Attack Complete **(AC).**

- Every hour, the system is being attacked by attackers coordinating together around the world and trying to compromise the system.
- The system states always transition from **NA to AP** and **AP to AC.**
- An attacker is successful in changing the state of the system with probability of 0.75 and fails with a probability of 0.25.
- If the attack fails, the system stays in its current state.
- If the system state reaches **AC** the attack is complete, and the system stays in that state.



$$A = \begin{array}{c} \\ \textbf{NA} \\ \textbf{AP} \\ \textbf{AC} \end{array} \begin{array}{c} \begin{array}{ccc} \textbf{NA} & \textbf{AP} & \textbf{AC} \end{array} \\ \begin{pmatrix} 0.25 & 0.75 & 0 \\ 0 & 0.25 & 0.75 \\ 0 & 0 & 1 \end{pmatrix} \end{array}$$

Transition Probability Matrix

Security Example: Emission matrix and initial distribution

Observation matrix (B)

- Your monitoring system reports two types of events
  - Port Scan **(PS)**
  - Software Installation **(SI)**
- Monitors are always accurate and works. Attackers cannot compromise the monitors. Every hour, we get information from the monitors if the attackers are trying to do **PS or SI.**

$$\mathbf{B} = \begin{array}{c c} & \begin{array}{cc} \textbf{PS} & \textbf{PI} \end{array} \\ \begin{array}{c} \textbf{NA} \\ \textbf{AP} \\ \textbf{AC} \end{array} & \begin{pmatrix} P_{PS|NA} & P_{PI|NA} \\ P_{PS|AP} & P_{PI|AP} \\ P_{PS|AC} & P_{PI|AC} \end{pmatrix} \end{array}$$

Observation Matrix

Initial distribution (π)

- We have no idea about the initial state of the system.

$$\pi_0 = \begin{array}{c} \textbf{NA} \\ \textbf{AP} \\ \textbf{AC} \end{array} \begin{pmatrix} 1/3 \\ 1/3 \\ 1/3 \end{pmatrix}$$

Initial state distribution/prior

General Inference question

Given the sequence of n observations $E_1, E_2, \ldots, E_n$, and the model (A, B, π), how do we choose a corresponding state sequence $S_1, S_2, \ldots, S_n$ which is optimal in some meaningful sense (i.e., best explains the observations)?

A simpler question: Given the sequence of n observations $E_1, E_2, \ldots, E_n$, and the model (A, B, π), what is the most probable state S_t at $t \in \{1, \ldots, n\}$?

$$\underset{j\in\{1,\ldots,N\}}{\operatorname{argmax}} P(S_t = \sigma_j | E_1, E_2, \ldots, E_n)$$

$$S = \{\sigma_1, \ldots, \sigma_N\}$$

Breaking down the inference question

$$P(S_t|E_1,E_2,\dots,E_n) = \frac{P(S_t,E_1,\dots,E_n)}{P(E_1,\dots,E_n)} = \frac{P(S_t,E_1,\dots,E_t,E_{t+1},\dots,E_n)}{P(E_1,\dots,E_n)}$$

Bayes rule

$$= \frac{P(E_{t+1},\dots,E_n\,|S_t,E_1,\dots,E_t)\,P(S_t,E_1,\dots,E_t)}{P(E_1,\dots,E_n)}$$

Bayes rule

$$= P(E_{t+1},\dots,E_n\,|S_t,E_1,\dots,E_t)\,P(S_t|E_1,\dots,E_t)\frac{P(E_1,\dots,E_t)}{P(E_1,\dots,E_n)}$$

Markov property

$$= \frac{P(E_{t+1},\dots,E_n\,|S_t)\,P(S_t|E_1,\dots,E_t)}{P(E_{t+1},\dots,E_n|E_1,\dots,E_t)}$$

Breaking down the inference question

$$P(S_t|E_1, E_2, \dots, E_n) = \frac{P(E_{t+1}, \dots, E_n\ |S_t)\ P(S_t|E_1, \dots, E_t)}{P(E_{t+1}, \dots, E_n|E_1, \dots, E_t)}$$

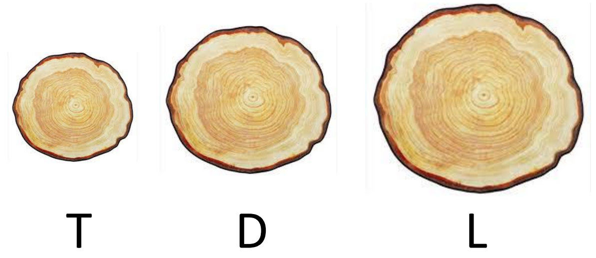
$P(S_t|E_1, \dots, E_t)$:
Probability of hidden state at time t given observation up to time t (**Forwards algorithm**)

$P(E_{t+1}, \dots, E_n\ |S_t)$:
Probability of the future observed sequence given the hidden state at time t (**Backwards algorithm**)

$P(E_{t+1}, \dots, E_n|E_1, \dots, E_t)$:
Does not depend on the hidden state (will not affect the maximization because it is just a scaling factor)

Forwards algorithm: Paleontological Temperature

Want to calculate $P(S_t|E_1, \ldots, E_t)$



- Let us calculate it for $t = 2$
- In the example, $E_1 = T, E_2 = L$
- Find $P(S_2 = H|E_1 = T, E_2 = L)$?

$$P(S_2 = H|E_1 = T, E_2 = L) = \frac{P(S_2 = H, E_1 = T, E_2 = L)}{P(E_1 = T, E_2 = L)}$$

Adding hidden state S_1

$$= \frac{\sum_{s\in\{H,C\}} P(S_2 = H, E_1 = T, E_2 = L, S_1 = s)}{P(E_1 = T, E_2 = L)}$$

Forwards algorithm: Paleontological Temperature

$$\frac{\sum_{s\in\{H,C\}} P(S_2 = H, E_1 = T, E_2 = L, S_1 = s)}{P(E_1 = T, E_2 = L)}$$

Bayes rule

$$= \frac{\sum_{s\in\{H,C\}} P(E_2 = L | S_2 = H, E_1 = T, S_1 = s) P(S_2 = H, E_1 = T, S_1 = s)}{P(E_1 = T, E_2 = L)}$$

Markov property

Bayes rule

$$= \frac{\sum_{s\in\{H,C\}} P(E_2 = L | S_2 = H) P(S_2 = H | E_1 = T, S_1 = s) P(S_1 = s | E_1 = T) P(E_1 = T)}{P(E_1 = T, E_2 = L)}$$

Markov property

Bayes rule

$$= \frac{\sum_{s\in\{H,C\}} P(E_2 = L | S_2 = H) P(S_2 = H | S_1 = s) P(S_1 = s | E_1 = T)}{P(E_2 = L | E_1 = T)}$$

Forwards algorithm: Paleontological Temperature

Hidden state given all observations up to that point

Observation probability

Transition probability

Hidden state given all observations up to that point

$$P(S_2 = H | E_1 = T, E_2 = L) = \frac{P(E_2 = L | S_2 = H) \sum_{s \in \{H,C\}} P(S_2 = H | S_1 = s) P(S_1 = s | E_1 = T)}{P(E_2 = L | E_1 = T)}$$

Define: $\alpha_t(i) = P(S_t = \sigma_i | E_1, E_2, \ldots, E_t)$ and $Z_t = P(E_t | E_1, \ldots E_{t-1})$

Above equation can be written as,

$$\boxed{\alpha_2(H)} = \frac{1}{Z_2} P(E_2 = L | S_2 = H) \sum_{s \in \{H,C\}} P(S_2 = H | S_1 = s) \boxed{\alpha_1(s)}$$

Where, $Z_2 = P(E_2 | E_1)$

Recursion

Forwards algorithm: General Expression

Define: $\alpha_t(j) = P(S_t = \sigma_j | E_1, E_2, \dots, E_t)$ and $Z_t = P(E_t | E_1, \dots E_{t-1})$

In general,

$$\alpha_t(j) = \frac{1}{Z_t} P\left(E_t \middle| S_t = \sigma_j\right) \sum_{i=1}^{N} P(S_t = \sigma_j \mid S_{t-1} = \sigma_i)\, \alpha_{t-1}(i) \qquad Z_t = \sum_{j=1}^{N} b_t \odot (A^T \alpha_{t-1})$$

Transition probability a_{ij}

Above equation can be written as a matrix for all j,

$$\begin{bmatrix} \alpha_t(1) \\ \vdots \\ \alpha_t(j) \\ \vdots \\ \alpha_t(N) \end{bmatrix} \propto \begin{bmatrix} P(E_t|S_t = \sigma_1) \\ \vdots \\ P(E_t|S_t = \sigma_j) \\ \vdots \\ P(E_t|S_t = \sigma_N) \end{bmatrix} \odot \begin{bmatrix} a_{11} & \dots & \dots & \dots & a_{N1} \\ \vdots & \ddots & \dots & \dots & \vdots \\ a_{1j} & \dots & a_{ij} & \dots & a_{Nj} \\ \vdots & \dots & \dots & \ddots & \dots \\ a_{1N} & \dots & \dots & \dots & a_{NN} \end{bmatrix} \begin{bmatrix} \alpha_{t-1}(1) \\ \vdots \\ \alpha_{t-1}(i) \\ \vdots \\ \alpha_{t-1}(N) \end{bmatrix}$$

$\odot$ Represents elementwise product (Hadamard product)

$$\alpha_t \propto b_t \odot (A^T \alpha_{t-1})$$

b_t is the column of the observation matrix B corresponding to E_t

Forwards Algorithm: Paleontological Temperature

For observations T, L, D, T, L

$$\begin{array}{ccc} & H & C \\ \begin{array}{c} H \\ C \end{array} & \multicolumn{2}{c}{\begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix}} \end{array}$$

Transition probability matrix

$P(S_2|E_1 = T, E_2 = L)$ is,

$$\begin{bmatrix} \alpha_2(H) \\ \alpha_2(C) \end{bmatrix} \propto \begin{bmatrix} 0.5 \\ 0.1 \end{bmatrix} \odot \left(\begin{bmatrix} 0.7 & 0.4 \\ 0.3 & 0.6 \end{bmatrix} \begin{bmatrix} \alpha_1(H) \\ \alpha_1(C) \end{bmatrix} \right)$$

$$\begin{array}{cccc} & T & D & L \\ \begin{array}{c} H \\ C \end{array} & \multicolumn{3}{c}{\begin{bmatrix} 0.1 & \boxed{0.4} & \boxed{0.5} \\ 0.7 & \boxed{0.2} & \boxed{0.1} \end{bmatrix}} \end{array}$$

Observation matrix

Similarly, $P(S_3|E_1 = T, E_2 = L,\ E_3 = D)$ is,

$$\begin{bmatrix} \alpha_3(H) \\ \alpha_3(C) \end{bmatrix} \propto \begin{bmatrix} 0.4 \\ 0.2 \end{bmatrix} \odot \left(\begin{bmatrix} 0.7 & 0.4 \\ 0.3 & 0.6 \end{bmatrix} \begin{bmatrix} \alpha_2(H) \\ \alpha_2(C) \end{bmatrix} \right)$$

Forwards Algorithm

1. Input: (A, B, π) and observed sequence $E_1, \dots, E_n$
2. $[\alpha_1, Z_1]$ = normalize($b_1 \odot \pi$)
3. **for** $t = 2{:}\, n$ **do**
   $[\alpha_t, Z_t]$ = normalize($b_t \odot (A^T \alpha_{t-1})$)
4. return $\alpha_1, \dots, \alpha_n$ and $\log\big(P(E_1, \dots, E_n)\big) = \sum_t \log(Z_t)$

Note:

Subroutine: [v, Z] = normalize(u): $Z = \sum_j u_j$; $v_j = u_j / Z$;

Breaking down the inference question

$$P(S_t|E_1, E_2, \dots, E_n) = \frac{P(E_{t+1}, \dots, E_n\ |S_t)\ P(S_t|E_1, \dots, E_t)}{P(E_{t+1}, \dots, E_n|E_1, \dots, E_t)}$$

$P(S_t|E_1, \dots, E_t)$:
Probability of hidden state at time t given observation up to time t (**Forwards algorithm**)

$P(E_{t+1}, \dots, E_n\ |S_t)$:
Probability of the future observed sequence given the hidden state at time t (**Backwards algorithm**)

$P(E_{t+1}, \dots, E_n|E_1, \dots, E_t)$:
Does not depend on the hidden state (will not affect the maximization because it is just a scaling factor)

Backwards Algorithm (similar to Forwards Algo.)

Calculate $P(E_{t+1}, \ldots, E_n \mid S_t)$

Define: $\beta_t(j) = P(E_{t+1}, \ldots, E_n \mid S_t = \sigma_j)$

Include S_t to use information from the one-step future

$$\boxed{\beta_{t-1}(j)} = P(E_t, \ldots, E_n \mid S_{t-1} = \sigma_j) = \sum_{i=1}^{N} P(S_t = \sigma_i, E_t, \ldots, E_n \mid S_{t-1} = \sigma_j)$$

Chain rule

$$= \sum_{i=1}^{N} P(E_{t+1}, \ldots, E_n \mid S_{t-1} = \sigma_j, S_t = \sigma_i, E_t)\, P(E_t \mid S_{t-1} = \sigma_j, S_t = \sigma_i) P(S_t = \sigma_i \mid S_{t-1} = \sigma_j)$$

Markov property

$$= \sum_{i=1}^{N} P(E_{t+1}, \ldots, E_n \mid S_t = \sigma_i)\, P(E_t \mid S_t = \sigma_i) P(S_t = \sigma_i \mid S_{t-1} = \sigma_j)$$

By definition of $\beta_t(j)$

$$= \sum_{i=1}^{N} \boxed{\beta_t(i)}\, P(E_t \mid S_t = \sigma_i) P(S_t = \sigma_i \mid S_{t-1} = \sigma_j)$$

Emission probability

Transition probability

In matrix form, we get,

$$\beta_{t-1} = A(b_t \odot \beta_t)$$

$$\beta_t = \begin{bmatrix} \beta_t(1) \\ \vdots \\ \beta_t(N) \end{bmatrix}$$

Backwards Algorithm

1. Input: (A, B, π) and observed sequence $E_1, \ldots, E_n$
2. *$\beta_n = 1$; // initialize $\beta_n(j)$ to 1 for all states σ_j*
3. **for** $t = n - 1 : 1$ **do**
   $\beta_{t-1} = A(b_t \odot \beta_t)$
4. return $\beta_1, \ldots, \beta_n$

Breaking down the inference question

$$P(S_t|E_1, E_2, \dots, E_n) = \frac{P(E_{t+1}, \dots, E_n \ |S_t)\ P(S_t|E_1, \dots, E_t)}{P(E_{t+1}, \dots, E_n|E_1, \dots, E_t)}$$

$P(S_t|E_1, \dots, E_t)$:
Probability of hidden state at time t given observation up to time t (**Forwards algorithm**)

$P(E_{t+1}, \dots, E_n \ |S_t)$:
Probability of the future observed sequence given the hidden state at time t (**Backwards algorithm**)

$P(E_{t+1}, \dots, E_n|E_1, \dots, E_t)$:
Does not depend on the hidden state (will not affect the maximization because it is just a scaling factor)

Inference – using Forwards-Backwards expressions

$$P(S_t|E_1, E_2, \dots, E_n) = \frac{P(E_{t+1}, \dots, E_n\,|S_t)\,P(S_t|E_1, \dots, E_t)}{P(E_{t+1}, \dots, E_n|E_1, \dots, E_t)}$$

For $S_t = \sigma_j$ and $\gamma_t(j) = P\left(S_t = \sigma_j \middle| E_1, E_2, \dots, E_n\right)$, the above equation is:

$$P\left(S_t = \sigma_j \middle| E_1, E_2, \dots, E_n\right) = \frac{P(E_{t+1}, \dots, E_n\,|S_t = \sigma_j)\,P(S_t = \sigma_j|E_1, \dots, E_t)}{P(E_{t+1}, \dots, E_n|E_1, \dots, E_t)}$$

$$\gamma_t(j) = \frac{\beta_t(j)\alpha_t(j)}{P(E_{t+1}, \dots, E_n|E_1, \dots, E_t)} = \frac{\beta_t(j)\alpha_t(j)}{\sum_{i=1}^{N} \beta_t(j)\alpha_t(j)}$$

Theorem of total probability

$$\boxed{\gamma_t(j) \propto \beta_t(j)\alpha_t(j)}$$

Inference: Most likely state

- Forwards-backwards algorithm gives $P(S_t = \sigma_j | E_1, \dots, E_n)$ for all j
- Find the individually most likely state at time t given all observations

$$S_t^* = \underset{j \in \{1,..,N\}}{\operatorname{argmax}} \gamma_t(j)$$

Optimality of inference

- In the inference problem we attempt to uncover the hidden part of HMM, i.e., find the "correct" state sequence
- It is impossible to find the "correct" state sequence (solution)
- Use optimality criterion to find the "best" possible solution
- Several reasonable criteria exist and is a strong function of the intended application
  - Most likely state given observations
    - Application in finding average statistics, expected number of correct states
    - Solved using Forwards-Backwards algorithm
  - Single best sequence that maximises probability of observed events
    - Application in continuous speech recognition
    - Solved using Viterbi algorithm

Resources

Rabiner’s (excellent) paper:
https://www.ece.ucsb.edu/Faculty/Rabiner/ece259/Reprints/tutorial%20on%20hmm%20and%20applications.pdf