Dynamic Bayesian networks

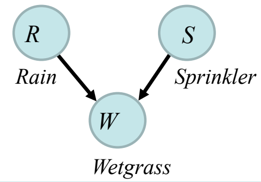
Lecture 10, DD2431, Machine Learning
KTH

A. Maki
October 2013

Background: a Bayesian network

- A probabilistic method for modeling dependencies between random variables through a **graph structure:**
  - random variables by nodes
  - conditional dependencies by edges

  (directed acyclic graph)



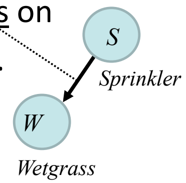
→ Probabilistic inference system

$P(R,S,W) = P(W|R,S)P(S)P(R)$

- Synonyms:
  = probabilistic network / causal network / knowledge map …

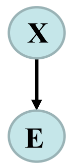
Probabilistic inference

- Intuitively, what we can observe depends on the true state of random query variables.



- The purpose of probabilistic inference in general:
  - to compute posterior probability distribution: $P(\mathbf{X} \mid \mathbf{e})$
  - for a set of query variables: $\mathbf{X}$
  - given a set of observed evidence variables: $\mathbf{E}$

    (assignment of values to them: $\mathbf{e}$)



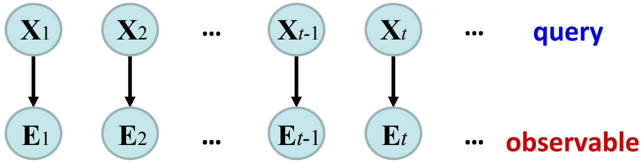
Notations

X = a random variable
(uppercase / begins with an uppercase, e.g. R or *"Rain"*)
$\mathbf{X}$ = a set of variables / vector random variables
$P(X)$ = a probability distribution for X
$P(\mathbf{X})$ = a probability distribution for $\mathbf{X}$
(probabilities of all the possible set of values of $\mathbf{X}$)

with a subscript
$\mathbf{X}_t$ = a set of variables at time t
$\mathbf{X}_{a:b} = \mathbf{X}_a, \mathbf{X}_{a+1}, \ldots, \mathbf{X}_{b-1}, \mathbf{X}_b$

A dynamic Bayesian network

- A Bayesian network that represents **sequences of variables**, e.g.
  - time-series generated by a dynamic system
  - sequences of symbols, e.g. a protein sequence

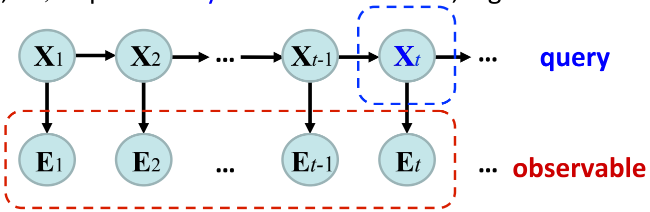


A **series of models** for the length of the sequence

- Again, two types of **random variables** in the graph:
  - A set of **query** variables $\mathbf{X}_t$ that are unknown (or hidden)
  - The other random variable $\mathbf{E}_t$ represents the **observations**

Making inference from sequential data

- The series of models are linked by a **markov assumption**: the state, $\mathbf{X}_t$, depends only on the recent state, e.g. $\mathbf{X}_{t-1}$.



- As we deal with time-series variables, **the question is**:

What is the posterior probability for a set of query variables, $\mathbf{X}_t$, given past and present evidence variables, $\mathbf{E}_{1:t}$?

$$P(\mathbf{X}_t \mid \mathbf{E}_{1:t}) = ?$$

Example

A professor wants to know if students are getting enough sleep.
→ $P(S)$

- **Each day**, the professor observes:
  - whether they have red eyes, and
  - whether the students sleep in class.

- The professor has the domain theory on:
  1. The initial state of "enough sleep"
  2. The transition model
  3. The observation model

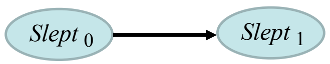
  … as detailed in the following:

1. Initial state / 2. State transition

- The prior probability of getting enough sleep, $P(S)$, with no observation is 0.7.

$$P(S) = P(S_0) = 0.7$$

$P(S_0)$
0.7



S = "Slept" = Enough sleep

S_{t-1}	$P(S_t)$
t	0.8
f	0.3

Conditional Probability Table (CPT)

- The probability of getting enough sleep at night t is 0.8 given that the student got enough sleep the previous night, and 0.3 if not.

$$P(S_t \mid S_{t-1}) = 0.8$$

$\neg S$ = Not enough sleep

$$P(S_t \mid \neg S_{t-1}) = 0.3$$

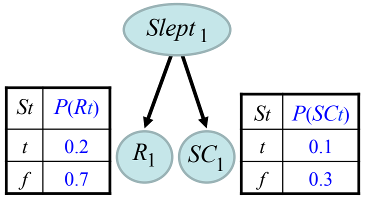
3. The observation model

- The probability of having red eyes is 0.2 if the student got enough sleep, and 0.7 if not.

$$P(R_t \mid S_t) = 0.2, \quad P(R_t \mid \neg S_t) = 0.7$$

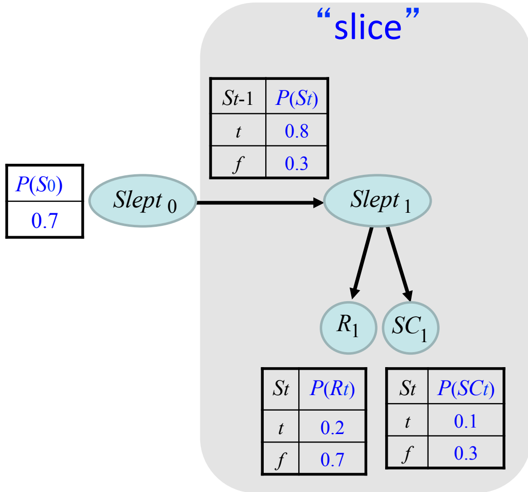
- The probability of sleeping in class is 0.1 if the student got enough sleep, and 0.3 if not.

$$P(SC_t \mid S_t) = 0.1, \quad P(SC_t \mid \neg S_t) = 0.3$$



Conditional Probability Table (CPT)

2.Transitions + 3.Observations = Slice



Compute $P(S_t \mid \mathbf{e}_{1:t})$

We wanted to know if students are getting enough sleep.
Using the evidence values: $\mathbf{e}_1$, $\mathbf{e}_2$, $\mathbf{e}_3$

$\mathbf{e}_1$ = not red eyes, not sleeping in class
$\mathbf{e}_2$ = red eyes, not sleeping in class
$\mathbf{e}_3$ = red eyes, sleeping in class

we would like to infer $P(S_t \mid \mathbf{e}_{1:t})$

Q. To start with, how does $P(S_1 \mid \mathbf{e}_1)$ compare to $P(S_0)$?

Compute $P(S_1 \mid \mathbf{e}_1)$

$\mathbf{e}_1$ = not red eyes, not sleeping in class

Conditional probability tables

S_{t-1}	$P(S_t)$
t	0.8
f	0.3

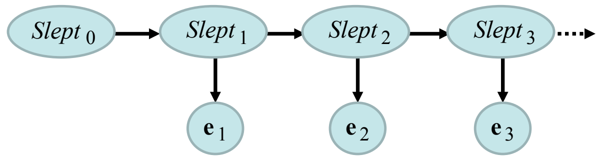
S_t	$P(R_t)$	$P(SC_t)$	$P(\mathbf{e}_1)$
t	0.2	0.1	0.72
f	0.7	0.3	0.21

Compute $P(S_1 \mid \mathbf{e}_1)$

- $P(S_0) = 0.7$
- $P(S_1) = P(S_1 \mid S_0)\,P(S_0) + P(S_1 \mid \neg S_0)\,P(\neg S_0) = 0.65$
- $P(S_1 \mid \mathbf{e}_1) = \alpha\, P(\mathbf{e}_1 \mid S_1)\,P(S_1) = \alpha < 0.72, 0.21 > < 0.65, 0.35 >$
  $= < 0.864, 0.136 >$

Answer $P(S_1 \mid \mathbf{e}_1) > P(S_0)$

- We would also like to infer
  - $P(S_2 \mid \mathbf{e}_{1:2})$
  - $P(S_3 \mid \mathbf{e}_{1:3})$
  - …

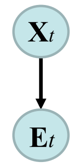


For this we can exploit **Markov processes**

Markov processes

- Basic idea:

  $\mathbf{X}_t$ = set of unobservable state variables at time t
  (enough sleep: S_t)
  $\mathbf{E}_t$ = set of observable evidence variables at time t
  (red eye R_t, sleep in class: SC_t)



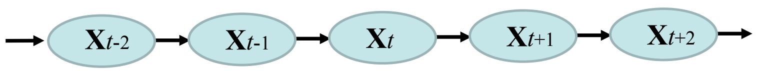
Copy state and evidence variables for each time step assuming **discrete time**; step size depends on problem.

N.B.
$\mathbf{X}_t$ and $\mathbf{E}_t$ are both vectors in general, and can contain many variables

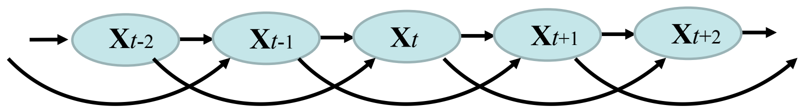
Markov processes (contd.)

- Markov assumption:
  $\mathbf{X}_t$ depends on **bounded subset** of $\mathbf{X}_{0:t-1}$

  - First-order Markov process: $P(\mathbf{X}_t \mid \mathbf{X}_{0:t-1}) = P(\mathbf{X}_t \mid \mathbf{X}_{t-1})$

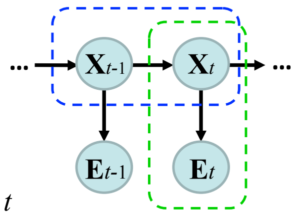


  - Second-order Markov process: $P(\mathbf{X}_t \mid \mathbf{X}_{0:t-1}) = P(\mathbf{X}_t \mid \mathbf{X}_{t-2,\, t-1})$



Markov processes (contd.)

- Sensor Markov assumption:
  $P(\mathbf{E}_t \mid \mathbf{X}_{0:t}, \mathbf{E}_{0:t-1}) = P(\mathbf{E}_t \mid \mathbf{X}_t)$

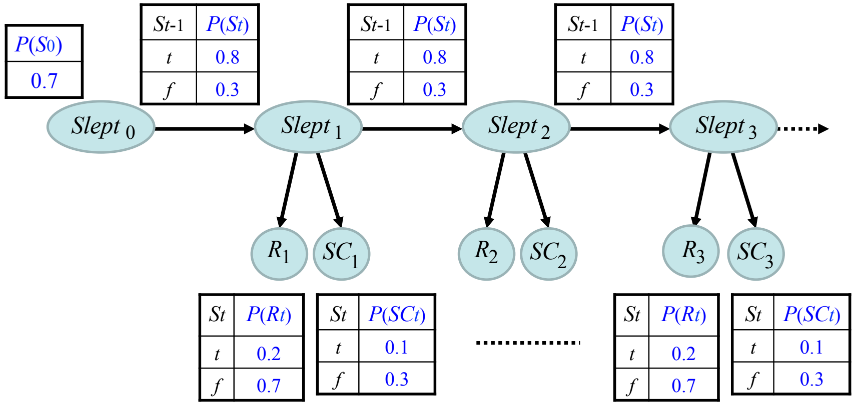


- Stationary process:
  Transition model $P(\mathbf{X}_t \mid \mathbf{X}_{t-1})$ and
  sensor model $P(\mathbf{E}_t \mid \mathbf{X}_t)$ fixed for all t

Knock-on effect:
The great attraction of Markov models is that they leverage a knock-on effect – that explicit short-range linkages give rise to implied long-range correlations.

Replicating slices: "unrolling"

until the observation sequence is accommodated



$P(S_0)$
0.7

S_{t-1}	$P(S_t)$
t	0.8
f	0.3

S_{t-1}	$P(S_t)$
t	0.8
f	0.3

S_{t-1}	$P(S_t)$
t	0.8
f	0.3

S_t	$P(R_t)$
t	0.2
f	0.7

S_t	$P(SC_t)$
t	0.1
f	0.3

…………….

S_t	$P(R_t)$
t	0.2
f	0.7

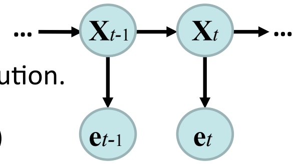
S_t	$P(SC_t)$
t	0.1
f	0.3

Dynamic Bayesian network for our example

State estimation: filtering

To compute the "belief state" : $P(\mathbf{X}_t \mid \mathbf{e}_{1:t})$

- A useful filtering algorithm maintains a current state estimate, $P(\mathbf{X}_t \mid \mathbf{e}_{1:t-1})$, and update it using present evidence, $\mathbf{e}_t$, rather than going back over the entire history.
- We would like to have a function f to find the posterior probability distribution.



$$P(\mathbf{X}_t \mid \mathbf{e}_{1:t}) = f\,(P(\mathbf{X}_t \mid \mathbf{e}_{1:t-1}), \mathbf{e}_t)$$

- We use the process of **recursive Bayesian inference**:
  1. Prediction step
  2. Update step

State estimation (contd.)

1. Prediction step

$$P(\mathbf{X}_t \mid \mathbf{e}_{1:t-1}) = \sum_{\mathbf{x}_{t-1}} P(\mathbf{X}_t \mid \mathbf{x}_{t-1})\, P(\mathbf{x}_{t-1} \mid \mathbf{e}_{1:t-1})$$

Transition   Prior

2. Update step

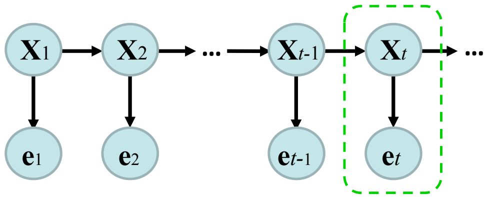
$$P(\mathbf{X}_t \mid \mathbf{e}_{1:t}) = P(\mathbf{X}_t \mid \mathbf{e}_t , \mathbf{e}_{1:t-1})$$
$$= \alpha P(\mathbf{e}_t \mid \mathbf{X}_t, \mathbf{e}_{1:t-1})\, P(\mathbf{X}_t \mid \mathbf{e}_{1:t-1})$$
$$= \alpha P(\mathbf{e}_t \mid \mathbf{X}_t)\, P(\mathbf{X}_t \mid \mathbf{e}_{1:t-1})$$ **… recursive form**

Bayes' theorem

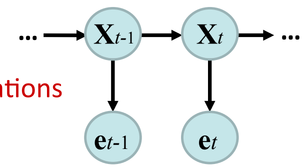
Likelihood

Sensor Markov assumption

α= Normalization constant



State estimation (contd.)



- Likelihood: $P(\mathbf{e}_t \mid \mathbf{X}_t)$
  - Measures how likely a set of observations are, given a state estimation.
- Transition: $P(\mathbf{X}_t \mid \mathbf{X}_{t-1})$
  - Governs the evolution of state estimate between two time steps.
  - Predicts the current state $\mathbf{X}_t$ from the previous state $\mathbf{X}_{t-1}$
- Prior term: $P(\mathbf{X}_{t-1} \mid \mathbf{e}_{1:t-1})$
  = The posterior probability for the previous time step
  - Provides knowledge of the past to give the **recursive Bayesian inference** a frame of reference.

Compute $P(S_t \mid \mathbf{e}_{1:t})$

- The evidence values, $\mathbf{e}_{1:3}$

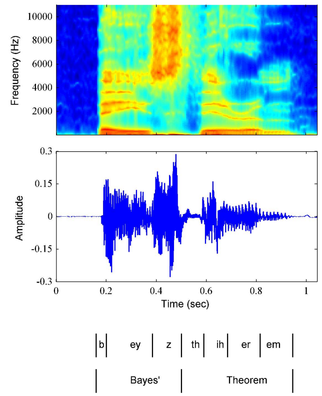
$\mathbf{e}_1$ = not red eyes, not sleeping in class
$\mathbf{e}_2$ = red eyes, not sleeping in class
$\mathbf{e}_3$ = red eyes, sleeping in class

- We had: $\boldsymbol{P}(S_1 \mid \mathbf{e}_1) = \alpha\ P(\mathbf{e}_1 \mid S_1)\ P(S_1) = <0.86, 0.136>$

- By applying the first-order Markov process
  - Prediction: $P(S_2 \mid \mathbf{e}_1) = \Sigma_{S_1} P(S_2 \mid S_1)\ P(S_1 \mid \mathbf{e}_1)$
    Update: $P(S_2 \mid \mathbf{e}_{1:2}) = \alpha\ P(\mathbf{e}_2 \mid S_2)\ P(S_2 \mid \mathbf{e}_1)$
  - Prediction: $P(S_3 \mid \mathbf{e}_{1:2}) = \Sigma_{S_2} P(S_3 \mid S_2)\ P(S_2 \mid \mathbf{e}_{1:2})$
    Update: $P(S_3 \mid \mathbf{e}_{1:3}) = \alpha\ P(\mathbf{e}_3 \mid S_3)\ P(S_3 \mid \mathbf{e}_{1:2})$

Special cases / applications

- Hidden Markov models (HMM)
  - A single discrete state variable
  - Used for speech recognition



(picture Taken from **Pattern Recognition and Machine Learning**, C. Bishop)

Summary

- Dynamic Bayesian networks:
  - An extension of Bayesian networks to handle temporal models
  - Specify prior distribution over the state variables, the transition model, and the sensor model
  - A concise graphical formalism for probabilistic inference using Markov process
  - Can contain arbitrary many query and evidence variables