Bayesian Network

• 2018/03/29

➢ Announcement:

☐ Slides for this lecture are here

http://www.public.asu.edu/~yzhan442/teaching/CSE471/Lectures/bayesnets-II.pdf

Slides are largely based on information from http://ai.berkeley.edu and Russel
Last time

• Hidden Markov model
  - Components of HMM
  - Independencies in HMM
  - Inference in HMM
    - Forward algorithm
    - Viterbi algorithm
  - HMM applications
    - Localization and particle filter
    - SLAM

• Required reading (red means it will be on your exams):
  - R&N: Chapter 15.3
Inference in probabilistic models: Review

What is a probabilistic model?

\[ P(X_1, X_2, ..., X_n) \]

What is inference in probabilistic models?

\[ P(X_1, X_2, ..., X_i | x_{i+1}, ..., x_n) \]

Exact inference vs approximate inference

- **General case:**
  - Evidence variables: \( E_1 \ldots E_k = e_1 \ldots e_k \)
  - Query* variable: \( Q \)
  - Hidden variables: \( H_1 \ldots H_r \)

- **We want:**
  \[ P(Q | e_1 \ldots e_k) \]

- **Step 1:** Select the entries consistent with the evidence

- **Step 2:** Sum out \( H \) to get joint of Query and evidence

- **Step 3:** Normalize

\[
P(Q, e_1 \ldots e_k) = \sum_{h_1 \ldots h_r} P(Q, h_1 \ldots h_r, e_1 \ldots e_k) \]

\[
Z = \sum_q P(Q, e_1 \ldots e_k)
\]

\[
P(Q | e_1 \ldots e_k) = \frac{1}{Z} P(Q, e_1 \ldots e_k)
\]
What is a probabilistic model of HMM?

\[ P(X_1, E_1, \ldots, X_T, E_T) = P(X_1)P(E_1|X_1)\prod_{t=2}^{T} P(X_t|X_{t-1})P(E_t|X_t) \]

What is inference in probabilistic models?
Two inference that we have discussed so far:
- a. Filtering \( P(X_t|e_1:t) \)
- b. Most likely explanation \( \text{argmax} P(x_1:t|e_1:t) \)

Exact inference vs approximate inference

- **General case:**
  - Evidence variables: \( E_1 \ldots E_k = e_1 \ldots e_k \)
  - Query* variable: \( Q \)
  - Hidden variables: \( H_1 \ldots H_r \)

- **We want:**
  - *Works fine with multiple query variables, too*
  - \( P(Q|e_1 \ldots e_k) \)

- **Step 1:** Select the entries consistent with the evidence

- **Step 2:** Sum out \( H \) to get joint of Query and evidence

- **Step 3:** Normalize

\[
P(Q|e_1 \ldots e_k) = \sum_{h_1 \ldots h_r} P(Q,h_1 \ldots h_r,e_1 \ldots e_k)X_1,X_2,\ldots X_n\]

\[
Z = \sum_{q} P(Q,e_1 \ldots e_k)\]

\[
P(Q|e_1 \ldots e_k) = \frac{1}{Z} P(Q,e_1 \ldots e_k)\]
State trellis

Given \( u_1, u_2, u_3 \ldots \)

- State trellis: graph of states and transitions over time

Path weight = joint probability of the corresponding state trajectory.

\[
P(s_1)P(u_1|s_1)P(s_2|s_1)P(u_2|x_2)
\]

- Each arc represents some transition \( x_{t-1} \rightarrow x_t \)
- Each path is a sequence of states
- The product of weights on a path is that sequence’s probability along with the evidence

\[
P(X_1, E_1, \ldots, X_T, E_T) = P(X_1)P(E_1|X_1) \prod_{t=2}^{T} P(X_t|X_{t-1})P(E_t|X_t)
\]

\[
= P(s_1)P(u_1|s_1)\ldots \text{ Referred as the probability of a path (state trajectory) in the trellis}
\]

- Forward algorithm computes sums of paths, Viterbi computes best paths
Forward algorithm

Filtering

\[
P(X_t | e_1:t) = \sum_{X_{1:t-1}} P(X_t, X_{1:t-1} | e_1:t) \\
\propto \sum_{X_{1:t-1}} P(X_t, X_{1:t-1}, e_1:t)
\]
Forward algorithm

\[
P(X_2 | u_{1:2}) \propto P(X_2, u_{1:2})
= \sum_{X_1} P(X_2, X_1, u_{1:2})
= \sum_{X_1} P(u_2 | X_1, X_2, u_1) P(X_1, X_2, u_1)
= \sum_{X_1} P(u_2 | X_2) P(X_2 | X_1, u_1) P(X_1, u_1)
= \sum_{X_1} P(u_2 | X_2) P(X_2 | X_1) P(X_1, u_1)
= P(u_2 | s_2) \sum_{X_1} P(X_2 | X_1) P(X_1, u_1)
\]
Forward Algorithm

$f_t(X_t) = P(X_t, u_{1:t})$

$P(X_t | u_{1:t}) \propto P(X_t, u_{1:t})$

$= f_t(X_t)$

$= P(u_t | X_t) \sum_{X_{t-1}} P(X_t | X_{t-1}) P(X_{t-1}, u_{1:t-1})$

$= P(u_t | X_t) \sum_{X_{t-1}} P(X_t | X_{t-1}) f_{t-1}(X_{t-1})$
Viterbi algorithm

Viterbi Algorithm

Most likely explanation:
\[
\text{argmax } P(x_1:t | e_1:t)
\]

\[
P(x_1:t | e_1:t) \propto P(x_1:t, e_1:t)
\]

So the same as \( \text{argmax } P(x_1:t, e_1:t) \)

Inference could be exponential!
Viterbi algorithm

\begin{align*}
\max_{X_1:t} P(X_1:t | u_1:t) \\
\propto \max_{X_1:t} (X_1:t, u_1:t) \\
= \max_{X_1:t} P(X_1:t-1, X_t, u_1:t) \\
= \max_{X_1:t} P(X_t | X_{t-1}) P(u_t | X_t) P(X_{1:t-1}, u_1:t) \\
= \max\{ \max_{X_1:t-1} P(X_1:t-1, s_t, u_1:t), \\
\max_{X_1:t-1} P(X_1:t-1, r_t, u_1:t) \} \\
m_t(X_t) \\
m_t(s_t) = \\
\max_{X_1:t-1} P(u_t | s_t) P(s_t | X_{t-1}) P(X_{1:t-1}, u_1:t-1) = \\
P(u_t | s_t) \max\{ P(s_t | s_{t-1}) \max_{X_1:t-2} P(X_{1:t-2}, s_{t-1}, u_1:t-1), \\
P(s_t | r_{t-1}) \max_{X_1:t-2} P(X_{1:t-2}, r_{t-1}, u_1:t-1) \} \\
m_{t-1}(X_{t-1})
\end{align*}
Forward and Viterbi algorithms

Forward Algorithm (Sum)

\[ f_t[x_t] = P(x_t, e_1:t) \]

\[ = P(e_t|x_t) \sum_{x_{t-1}} P(x_t|x_{t-1}) f_{t-1}[x_{t-1}] \]

Viterbi Algorithm (Max)

\[ m_t[x_t] = \max_{x_1:t-1} P(x_1:t-1, x_t, e_1:t) \]

\[ = P(e_t|x_t) \max_{x_{t-1}} P(x_t|x_{t-1}) m_{t-1}[x_{t-1}] \]
Outline for today

• Bayesian network
  o Semantics of BN
  o Probabilities in BN
  o Dynamic BN

• Required reading (red means it will be on your exams):
  o R&N: Chapter 14.1-2
Bayesian network (Bayes net)
Two problems with using full joint distribution tables as our probabilistic models:

- Unless there are only a few variables, the joint is WAY too big to represent explicitly
- Hard to learn (estimate) anything empirically about more than a few variables at a time

Bayes’ nets: a technique for describing complex joint distributions (models) using simple, local distributions (conditional probabilities)

- More properly called graphical models
- We describe how variables locally interact
- Local interactions chain together to give global, indirect interactions
- For about 10 min, we’ll be vague about how these interactions are specified
Models describe how (a portion of) the world works

What do we do with probabilistic models?
- We (or our agents) need to reason about unknown variables, given evidence
  - Example: explanation (diagnostic reasoning)
  - Example: prediction (causal reasoning)
  - Example: value of information

Models are always simplifications
- May not account for every variable
- May not account for all interactions between variables
  - “All models are wrong; but some are useful.”
    - George E. P. Box
BN example: insurance
BN example: car

- Battery age
  - Battery dead
    - Battery meter
      - Lights
    - Battery flat
      - Oil light
      - Gas gauge
      - Car won’t start
  - Alternator broken
  - Fanbelt broken
  - No charging
    - No oil
    - No gas
    - Fuel line blocked
    - Starter broken
  - Dipstick
Graphical model

• Nodes: variables (with domains)
  – Can be assigned (observed) or unassigned (unobserved)

• Arcs: interactions
  – Indicate “direct influence” between variables
  – Formally: encode conditional independence (more later)

• For now: imagine that arrows mean direct causation (in general, they don’t!)
Coin flips

N independent coin flips

\[ X_1, X_2, \ldots, X_n \]

No interactions between variables: \textit{absolute independence}
Traffic

Variables:
- R: It rains
- T: There is traffic

Model 1: independence

Model 2: rain causes traffic

Why is an agent using model 2 better?
Let’s build a causal graphical model!

Variables
- T: Traffic
- R: It rains
- L: Low pressure
- D: Roof drips
- B: Ballgame
- C: Cavity
Alarm

- Variables
  - B: Burglary
  - A: Alarm goes off
  - M: Mary calls
  - J: John calls
  - E: Earthquake!
Semantics of BN
Semantics of BN

- A set of nodes, one per variable $X$
- A directed, **acyclic** graph
- A conditional distribution for each node
  - A collection of distributions over $X$, one for each combination of parents’ values
  - $P(X|a_1 \ldots a_n)$
  - CPT: conditional probability table
  - Description of a noisy “causal” process

A Bayes net = Topology (graph) + Local Conditional Probabilities
Probabilities in BN

- Bayes’ nets implicitly encode joint distributions
  - As a product of local conditional distributions
  - To see what probability a BN gives to a full assignment, multiply all the relevant conditionals together:

\[
P(x_1, x_2, \ldots x_n) = \prod_{i=1}^{n} P(x_i | \text{parents}(X_i))
\]

- Example:

\[
P(+\text{cavity}, +\text{catch}, -\text{toothache})
\]
Why are we guaranteed that setting

\[ P(x_1, x_2, \ldots x_n) = \prod_{i=1}^{n} P(x_i|\text{parents}(X_i)) \]

results in a proper joint distribution?

Chain rule (valid for all distributions):

\[ P(x_1, x_2, \ldots x_n) = \prod_{i=1}^{n} P(x_i|x_1 \ldots x_{i-1}) \]

Assume conditional independences:

\[ P(x_i|x_1, \ldots x_{i-1}) = P(x_i|\text{parents}(X_i)) \]

\[ \rightarrow \text{Consequence:} \quad P(x_1, x_2, \ldots x_n) = \prod_{i=1}^{n} P(x_i|\text{parents}(X_i)) \]

Not every BN can represent every joint distribution

– The topology enforces certain conditional independencies
Joint distribution of HMM

- Joint distribution:
\[ P(X_1, E_1, X_2, E_2, X_3, E_3) = P(X_1)P(E_1|X_1)P(X_2|X_1)P(E_2|X_2)P(X_3|X_2)P(E_3|X_3) \]

- More generally:
\[ P(X_1, E_1, \ldots, X_T, E_T) = P(X_1)P(E_1|X_1) \prod_{t=2}^{T} P(X_t|X_{t-1})P(E_t|X_t) \]

HMM is a Bayesian Network!
Coin flips

Only distributions whose variables are absolutely independent can be represented by a Bayesian net with no arcs.
Traffic

\[
P(R) = \begin{array}{c|c}
  +r & 1/4 \\
  -r & 3/4 \\
\end{array}
\]

\[
P(T | R) = \begin{array}{c|cc}
  +r & +t & 3/4 \\
  +r & -t & 1/4 \\
  -r & +t & 1/2 \\
  -r & -t & 1/2 \\
\end{array}
\]

\[
P(+r, -t) = \begin{array}{c|cc}
  +r & +t & 3/16 \\
  +r & -t & 1/16 \\
  -r & +t & 6/16 \\
  -r & -t & 6/16 \\
\end{array}
\]
Reverse Traffic?

Reverse causal direction

\[
P(T)
\]
+\(t\) 9/16
-\(t\) 7/16

\[
P(R|T)
\]
+\(t\) 1/3
-\(r\) 2/3

+\(-t\) 1/7
-\(-r\) 6/7

\[
P(T, R)
\]
+\(+r\) +\(t\) 3/16
+\(+r\) -\(t\) 1/16
-\(-r\) +\(t\) 6/16
-\(-r\) -\(t\) 6/16
Alarm

<table>
<thead>
<tr>
<th>B</th>
<th>P(B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>+b</td>
<td>0.001</td>
</tr>
<tr>
<td>-b</td>
<td>0.999</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>E</th>
<th>P(E)</th>
</tr>
</thead>
<tbody>
<tr>
<td>+e</td>
<td>0.002</td>
</tr>
<tr>
<td>-e</td>
<td>0.998</td>
</tr>
</tbody>
</table>

| B  | E  | A   | P(A|B,E) |
|----|----|-----|---------|
| +b | +e | +a  | 0.95    |
| +b | +e | -a  | 0.05    |
| +b | -e | +a  | 0.94    |
| +b | -e | -a  | 0.06    |
| -b | +e | +a  | 0.29    |
| -b | +e | -a  | 0.71    |
| -b | -e | +a  | 0.001   |
| -b | -e | -a  | 0.999   |

| A  | J  | P(J|A) |
|----|----|-------|
| +a | +j | 0.9   |
| +a | -j | 0.1   |
| -a | +j | 0.05  |
| -a | -j | 0.95  |

| A  | M  | P(M|A) |
|----|----|-------|
| +a | +m | 0.7   |
| +a | -m | 0.3   |
| -a | +m | 0.01  |
| -a | -m | 0.99  |
Causality

• When Bayes’ nets reflect the true causal patterns:
  – Often simpler (nodes have fewer parents)
  – Often easier to think about
  – Often easier to elicit from experts

• BNs need not actually be causal
  – Sometimes no causal net exists over the domain (especially if variables are missing)
  – E.g. consider the variables Traffic and Drips
  – End up with arrows that reflect correlation, not causation

• What do the arrows really mean?
  – Topology may happen to encode causal structure
  – Topology really encodes conditional independence (or correlations)

\[
P(x_i|x_1, \ldots x_{i-1}) = P(x_i|\text{parents}(X_i))
\]
Bayesian network

• So far: how a Bayes’ net encodes a joint distribution

• Next: how to answer queries about that distribution
  – Today:
    • First assembled BNs using an intuitive notion of conditional independence as causality
    • Then saw that key property is conditional independence
  – Main goal: answer queries about conditional independence and influence

• After that: how to answer numerical queries (inference)
Dynamic Bayesian network
Dynamic Bayesian network

- We want to track multiple variables over time, using multiple sources of evidence.
- Idea: Repeat a fixed Bayes net structure at each time.
- Variables from time t can condition on those from t-1.
Joint distribution of HMM

- Joint distribution:
  \[ P(X_1, E_1, X_2, E_2, X_3, E_3) = P(X_1)P(E_1|X_1)P(X_2|X_1)P(E_2|X_2)P(X_3|X_2)P(E_3|X_3) \]

- More generally:
  \[ P(X_1, E_1, \ldots, X_T, E_T) = P(X_1)P(E_1|X_1) \prod_{t=2}^{T} P(X_t|X_{t-1})P(E_t|X_t) \]

HMMs is a special type of DBN!
Particle filter for DBN

A particle is a complete sample for a time step

**Initialize**: Generate prior samples for the $t=1$ Bayes net
   Example particle: $G_1^a = (3,3)$ $G_1^b = (5,3)$

**Elapse time**: Sample a successor for each particle
   Example successor: $G_2^a = (2,3)$ $G_2^b = (6,3)$

**Observe**: Weight each *entire* sample by the likelihood of the evidence conditioned on the sample
   Likelihood: $P(E_1^a | G_1^a) \times P(E_1^b | G_1^b)$

**Resample**: Select prior samples (tuples of values) in proportion to their likelihood
Outline for today

• Bayesian network
  o Semantics of BN
  o Probabilities in BN
  o Dynamic BN

• Required reading (red means it will be on your exams):
  o R&N: Chapter 14.1-2