Lecture 9: Reasoning over Time
CS486/686 Intro to Artificial Intelligence

2023-6-8

Pascal Poupart
David R. Cheriton School of Computer Science
Outline

- Reasoning under uncertainty over time
- Hidden Markov Models
- Dynamic Bayesian Networks
So far...

- Assume the world doesn’t change
- Static probability distribution
  - Ex: when repairing a car, whatever is broken remains broken during the diagnosis

But the world evolves over time...

- How can we use probabilistic inference for weather predictions, stock market predictions, patient monitoring, etc?
Dynamic Inference

- Need to reason over time
  - Allow the world to evolve
  - Set of states (encoding all possible worlds)
  - Set of time-slices (snapshots of the world)
  - Different probability distribution over states at each time slice
  - Dynamics encoding how distributions change over time
Stochastic Process

- Definition
  - Set of States: $S$
  - Stochastic dynamics: $\Pr(s_t|s_{t-1}, \ldots, s_0)$

- Can be viewed as a Bayes net with one random variable per time slice
Stochastic Process

- Problems:
  - Infinitely many variables
  - Infinitely large conditional probability tables

- Solutions:
  - Stationary process: dynamics do not change over time
  - Markov assumption: current state depends only on a finite history of past states
**K-order Markov Process**

- Assumption: last k states sufficient

- First-order Markov Process
  - \( \Pr(s_t|s_{t-1}, ..., s_0) = \Pr(s_t|s_{t-1}) \)

- Second-order Markov Process
  - \( \Pr(s_t|s_{t-1}, ..., s_0) = \Pr(s_t|s_{t-1}, s_{t-2}) \)
K-order Markov Process

- Advantage:
  - Can specify entire process with finitely many time slices

- Two slices sufficient for a first-order Markov process...
  - Graph: $s_{t-1} \rightarrow s_t$
  - Dynamics: $\Pr(s_t|s_{t-1})$
  - Prior: $\Pr(s_0)$
Mobile Robot Localisation

- Example of a first-order Markov process

- Problem: uncertainty grows over time...
Hidden Markov Models

• Robot could use sensors to reduce location uncertainty...

• In general:
  – States not directly observable, hence uncertainty captured by a distribution
  – Uncertain dynamics increase state uncertainty
  – Observations made via sensors reduce state uncertainty

• Solution: Hidden Markov Model
First-order Hidden Markov Model

- **Definition:**
  - Set of states: $S$
  - Set of observations: $O$
  - Transition model: $\Pr(s_t|s_{t-1})$
  - Observation model: $\Pr(o_t|s_t)$
  - Prior: $\Pr(s_0)$
Mobile Robot Localisation

• (First-order) Hidden Markov Model:
  – $S$: $(x,y)$ coordinates of the robot on a map
  – $O$: distances to surrounding obstacles (measured by laser range finders or sonars)
  – $\Pr(s_t|s_{t-1})$: movement of the robot with uncertainty
  – $\Pr(o_t|s_t)$: uncertainty in the measurements provided by laser range finders and sonars

• Localisation corresponds to the query: $\Pr(s_t|o_t, \ldots, o_1)$?
Inference in temporal models

- Four common tasks:
  - Monitoring: $\Pr(s_t|o_t, \ldots, o_1)$
  - Prediction: $\Pr(s_{t+k}|o_t, \ldots, o_1)$
  - Hindsight: $\Pr(s_k|o_t, \ldots, o_1)$ where $k < t$
  - Most likely explanation: $\arg\max_{s_t, \ldots, s_1} \Pr(s_t, \ldots, s_1|o_t, \ldots, o_1)$

- What algorithms should we use?
  - First 3 tasks can be done with variable elimination and 4th task with a variant of variable elimination
• \( \Pr(s_t|o_t, \ldots, o_1) \): distribution over current state given observations

• Examples: robot localisation, patient monitoring

• **Forward algorithm**: corresponds to variable elimination
  – Factors: \( \Pr(s_0), \Pr(s_i|s_{i-1}), \Pr(o_i|s_i), 1 \leq i \leq t \)
  – Restrict \( o_1, \ldots, o_t \) to the observations made
  – Summout \( s_0, \ldots, s_{t-1} \)
  – \( \Sigma_{s_0 \ldots s_{t-1}} \Pr(s_0) \prod_{1 \leq i \leq t} \Pr(s_i|s_{i-1}) \Pr(o_i|s_i) \)
Prediction

• $\Pr(s_{t+k}|o_t, \ldots, o_1)$: distribution over future state given observations

• Examples: weather prediction, stock market prediction

• **Forward algorithm**: corresponds to variable elimination
  - Factors: $\Pr(s_0)$, $\Pr(s_i|s_{i-1})$, $\Pr(o_i|s_i)$, $1 \leq i \leq t+k$
  - Restrict $o_1, \ldots, o_t$ to the observations made
  - Summout $s_0, \ldots, s_{t+k-1}$, $o_{t+1}, \ldots, o_{t+k}$
  - $\sum_{s_0 \ldots s_{t+k-1}, o_{t+1} \ldots o_{t+k}} \Pr(s_0) \prod_{1 \leq i \leq t+k} \Pr(s_i|s_{i-1}) \Pr(o_i|s_i)$
Hindsight

- \( \Pr(s_k | o_t, ..., o_1) \) for \( k < t \): distribution over a past state given observations

- Example: crime scene investigation

- **Forward-backward algorithm**: corresponds to variable elimination
  - Factors: \( \Pr(s_0), \Pr(s_i | s_{i-1}), \Pr(o_i | s_i), 1 \leq i \leq t \)
  - Restrict \( o_1, ..., o_t \) to the observations made
  - Summout \( s_0, ..., s_{k-1}, s_{k+1}, ..., s_t \)
  - \( \sum_{s_0,..s_{k-1},s_{k+1},...,s_t} \Pr(s_0) \prod_{1 \leq i \leq t} \Pr(s_i | s_{i-1}) \Pr(o_i | s_i) \)
Most likely explanation

- \( \text{Argmax}_{s_0 \ldots s_t} \Pr(s_0, \ldots, s_t | o_t, \ldots, o_1) \): most likely state sequence given observations

- **Example**: speech recognition

- **Viterbi algorithm**: corresponds to a variant of variable elimination
  - Factors: \( \Pr(s_0), \Pr(s_i | s_{i-1}), \Pr(o_i | s_i), 1 \leq i \leq t \)
  - Restrict \( o_1, \ldots, o_t \) to the observations made
  - Maxout \( s_0, \ldots, s_t \)
  - \( \max_{s_0 \ldots s_t} \Pr(s_0) \prod_{1 \leq i \leq t} \Pr(s_i | s_{i-1}) \Pr(o_i | s_i) \)
• Hidden Markov Models are Bayes nets with a polytree structure

• Hence, variable elimination is
  – Linear with respect to # of time slices
  – Linear with respect to largest conditional probability table (Pr(s_t|s_{t-1}) or Pr(o_t|s_t))

• What if # of states or observations are exponential?
Dynamic Bayesian Networks

• Idea: encode states and observations with several random variables

• Advantage: exploit conditional independence to save time and space

• HMMs are just DBNs with one state variable and one observation variable
Mobile Robot Localisation

- States: (x,y) coordinates and heading $\theta$
- Observations: laser and sonar
DBN complexity

- Conditional independence allows us to write transition and observation models **very compactly**!

- Time and space of inference: conditional independence rarely helps...
  - Inference tends to be exponential in the number of state variables
  - Intuition: all state variables eventually get correlated
  - No better than with HMMs 😞
Non-Stationary Process

- What if the process is not stationary?
- Solution: add new state components until dynamics are stationary
- Example:
  - Robot navigation based on \((x,y,\theta)\) is non-stationary when velocity varies...
  - Solution: add velocity to state description e.g. \((x,y,v,\theta)\)
  - If velocity varies... then add acceleration
  - Where do we stop?
Non-Markovian Process

- What if the process is not Markovian?
- Solution: add new state components until dynamics are Markovian
- Example:
  - Robot navigation based on \((x,y,\theta)\) is non-Markovian when influenced by battery level...
  - Solution: add battery level to state description e.g. \((x,y,\theta,b)\)
Markovian Stationary Process

- Problem: adding components to the state description to force a process to be Markovian and stationary may significantly increase computational complexity

- Solution: try to find the smallest state description that is self-sufficient (i.e., Markovian and stationary)
Applications of static and temporal inference are virtually limitless

Some examples:
- mobile robot navigation
- vacuum cleaners
- speech recognition
- patient monitoring
- help system under Windows
- fault diagnosis in Mars rovers
- etc.
Robot localisation

- Demo at 1:15
Neato Robotics

Uses particle filtering (approximate inference technique based on sampling) for simultaneous localisation and mapping

See patent: http://www.faqs.org/patents/assignee/neato-robotics-inc/
Comparison

• Comparison at 4:34