CS262a Bayesian Networks

© Tingfeng Xia @ UCLA

Winter Quarter, 2022

Preface

This document is intentionally kept short, and is not a complete summary of what is covered in this course. Excerpt from online:

> The objective of this class is to provide an in-depth exposition of knowledge representation, reasoning, and machine learning under uncertainty using the framework of Bayesian networks. Both theoretical underpinnings and practical considerations will be covered, with a special emphasis on constructing and learning graphical models, and on various exact and approximate inference algorithms. Additional topics include logical approaches to probabilistic inference, compilation techniques, sensitivity analysis, undirected graphical models, and statistical relational learning.

Instructor: Professor Adnan Darwiche

Book: Adnan Darwiche. Modeling and Reasoning with Bayesian Networks. Cambridge University Press 2009.

1 Propositional Logic

1.1 Principle Logical Forms

Inconsistent. Something that never holds; $\text{Mods}(\cdot) =$; $\Pr(\alpha) = 0$

Valid. Something that always holds; $\text{Mods}(\cdot) = \Omega$; $\Pr(\alpha) = 1$

Equivalent. $\text{Mods}(\alpha) = \text{Mods}(\beta)$

Mutual Exclusive. $\text{Mods}(\alpha) \cap \text{Mods}(\beta) = \emptyset$

Exhaustive. $\text{Mods}(\alpha) \cup \text{Mods}(\beta) = \Omega$

Entailment / Implication. $\alpha \models \beta \triangleq \text{Mods}(\alpha) \subseteq \text{Mods}(\beta)$

1.2 Equivalent Forms

- $\text{Mods}(\alpha \wedge \beta) = \text{Mods}(\alpha) \cap \text{Mods}(\beta)$
- $\text{Mods}(\alpha \vee \beta) = \text{Mods}(\alpha) \cup \text{Mods}(\beta)$
- $\text{Mods}(\neg\alpha) = \overline{\text{Mods}(\alpha)}$

1.3 Instantiation Agreement

Two instantiation, each of which can cover a subset of different varaibles, are said to be compatible with each other if they argree on all common variables. Denoted as $\mathbf{x} \sim \mathbf{y}$.

1.4 Information Theory

Entropy.

$$\text{ENT}(X) = -\sum_{x} \Pr(x) \log \Pr(x) \tag{1.1}$$

where $0 \log 0 = 0$ by convention. With a higher entropy, we say that it is more chaotic.

Conditional Entropy.

$$\text{ENT}(X|Y) = \sum_{y} \Pr(y)\text{ENT}(X|y) \quad \text{where} \quad \text{ENT}(X|y) = -\sum_{x} \Pr(x|y) \log \Pr(x|y) \tag{1.2}$$

Conditioning never increases the entropy, i.e.

$$\text{ENT}(X|Y) \leq \text{ENT}(X) \tag{1.3}$$

Mutual Information

$$\text{MI}(X;Y) = \sum_{x,y} \Pr(x,y) \log \frac{\Pr(x,y)}{\Pr(x)\Pr(y)} \tag{1.4}$$

$$= \text{ENT}(X) - \text{ENT}(X|Y) \tag{1.5}$$

$$= \text{ENT}(Y) - \text{ENT}(Y|X) \tag{1.6}$$

Conditional Mutual Information

$$\begin{aligned}\mathrm{MI}(X;Y|Z) &= \sum_{x,y,z} \Pr(x,y,z) \log \frac{\Pr(x,y|z)}{\Pr(x|z)\Pr(y|z)} && (1.7)\\ &= \mathrm{ENT}(X|Z) - \mathrm{ENT}(X|Y,Z) && (1.8)\\ &= \mathrm{ENT}(Y|Z) - \mathrm{ENT}(Y|X,Z) && (1.9)\end{aligned}$$

2 Probability Calculus

2.1 Bayesian Conditioning

Bayesian Condition is specified by the formula

$$\Pr(\alpha|\beta) = \frac{\Pr(\alpha \wedge \beta)}{\Pr(\beta)} \tag{2.1}$$

In particular, not to be confused with Bayesian inference (to be added later).

2.2 Independence and Notations

Independence.

$$\begin{aligned}\alpha \perp\!\!\!\perp \beta &\iff \Pr(\alpha|\beta) = \Pr(\alpha) \vee \Pr(\beta) = 0 && (2.2)\\ &\iff \Pr(\alpha \wedge \beta) = \Pr(\alpha)\Pr(\beta) && (2.3)\end{aligned}$$

Conditional Independence.

$$\begin{aligned}(\alpha \perp\!\!\!\perp \beta)|\gamma &\iff \Pr(\alpha|\beta \wedge \gamma) = \Pr(\alpha|\gamma) \vee \Pr(\beta \wedge \gamma) = 0 && (2.4)\\ &\iff \Pr(\alpha \wedge \beta|\gamma) = \Pr(\alpha|\gamma)\Pr(\beta|\gamma) \vee \Pr(\gamma) = 1 && (2.5)\end{aligned}$$

Set Independence

$$I_{\Pr}(X,Z,Y) \iff (x \perp\!\!\!\perp y)|z, \quad \forall x,y,z \in X,Y,Z \tag{2.6}$$

3 Bayesian Networks

3.1 Soft Evidence

3.1.1 All Things Considered Method

We normalize / rescale w according to new evidence.

$$\Pr'(w) = \begin{cases} \frac{\Pr'(\beta)}{\Pr(\beta)}\Pr(w) & \text{if} w \models \beta \\ \frac{\Pr'(\neg\beta)}{\Pr(\neg\beta)}\Pr(w) & \text{if} w \models \neg\beta \end{cases} \tag{3.1}$$

The closed form is called the Jefferey's Rule.

Jeffery's Rule

$$\Pr'(\alpha) = q\Pr(\alpha|\beta) + (1-q)\Pr(\alpha|\neg\beta) \tag{3.2}$$

Jeffery's Rule - General Case

$$\Pr'(\alpha) = \sum_{i=1}^{n} \Pr'(\beta_i)\Pr(\alpha|\beta_i) \tag{3.3}$$

3.1.2 Nothing-else Considered Method

Odds.

$$O(\beta) = \frac{\Pr(\beta)}{\Pr(\neg\beta)} \tag{3.4}$$

Bayes Factor

$$k = \frac{O'(\beta)}{O(\beta)} = \frac{Pr'(\beta)/\Pr'(\neg\beta)}{\ldots} \tag{3.5}$$

from where we can expand and organize

$$\Pr'(\beta) = \frac{k\Pr(\beta)}{k\Pr(\beta) + \Pr(\neg\beta)} \tag{3.6}$$

Closed Form Solution.

$$\Pr'(\alpha) = \frac{k\Pr(\alpha \wedge \beta) + \Pr(\alpha \wedge \neg\beta)}{k\Pr(\beta) + \Pr(\neg\beta)} \tag{3.7}$$

3.2 Noisy Sensors

$$O'(\beta) = \underbrace{\frac{1-f_n}{f_p}}_{k^+} O(\beta) \qquad O'(\beta) = \underbrace{\frac{f_n}{1-f_p}}_{k^-} O(\beta) \tag{3.8}$$

3.3 Markov Assumptions

$$\text{Markov}(G) = \{I_{\Pr}(V, \text{Parents}(V), \text{ND}(V)\}_V \tag{3.9}$$

where ND means non-descendants, and includes all nodes except for $V, \text{Parents}(V)$ and $\text{Descendants}(V)$ (all the way till leaf)

3.4 Graphoid Axioms

Symmetry.

$$I_{\Pr}(X, Z, Y) \iff I_{\Pr}(Y, Z, X) \tag{3.10}$$

Decomposition.

$$I_{\Pr}(X, Z, Y \cup W) \implies I_{\Pr}(X, Z, Y) \wedge I_{\Pr}(X, Z, W) \tag{3.11}$$

Weak Union.

$$I_{\Pr}(X, Z, Y \cup W) \implies I_{\Pr}(X, Z \cup Y, W) \tag{3.12}$$

Contraction.

$$I_{\Pr}(X, Z, Y) \wedge I_{\Pr}(X, Z \cup Y, W) \implies I_{\Pr}(X, Z, Y \cup W) \tag{3.13}$$

Triviality.

$$I_{\Pr}(X, Z, \emptyset) \tag{3.14}$$

3.5 Positive Graphoid Axioms

... includes everything from Graphoid Axioms (Section 3.4) and in addition has

Intersection.

$$I_{\Pr}(X, Z \cup W, Y) \wedge I_{\Pr}(X, Z \cup Y, W) \implies I_{\Pr}(X, Z, Y \cup W) \tag{3.15}$$

3.6 D-seperation Linear Prune Theorem

3.7 D-seperation Properties

Soundness.

$$\text{dsep}_G(X, Z, Y) \implies I_{\text{Pr}}(X, Z, Y) \tag{3.16}$$

(Weak) Completeness. There exists a parametrization Θ that for every DAG G such that

$$I_{\text{Pr}}(X, Z, Y) \iff \text{dsep}_G(X, Z, Y) \tag{3.17}$$

4 Inference by Factor Elimination

4.1 Elimination Trees

Variables. $vars(i)$ denotes the variables mentioned at node i. $vars(i, j)$ denotes all variables mentioned in nodes to the i-side of the graph (inclusive). Hence, it holds that $vars(i) \subseteq vars(i, j)$.

Separators.

$$S_{ij} \triangleq vars(i, j) \cap vars(j, i) \tag{4.1}$$

Clusters.

$$C_i \triangleq vars(i) \cup \bigcup_j S_{ij} \tag{4.2}$$

5 Inference by Conditioning

5.1 Run time Comparison

5.1.1 Variable Elimination (VE)

Let w be the width of the tree, n denote the number of variables, and $|Q|$ as query variable size.

Time Complexity. $O(n \exp(w))$

Space Complexity.

$$O(n \exp(w) + n \exp(|Q|)) \equiv O(n \exp(w)), \quad \text{if } |Q| < \infty \tag{5.1}$$

5.1.2 Massage Passing

One Message. The cost to pass one message is

$$O(\exp(|C_i|)) \tag{5.2}$$

where $|C_i|$ is the cluster size.

Every Message.

$$O(\exp(w)) \tag{5.3}$$

what is this again?

where w is the width of the elimination tree.

Amount of Messages. The total amount of messages is

$$O(2(m-1)) \qquad \text{where } m = |V| \tag{5.4}$$

since we have a tree structure, meaning we have exactly $(m-1)$ edges and each edge can have forward backward each once.

All Messages - All Cluster Marginals. The total time to pass all messages and compute all cluster marginals is

$$O(m\exp(w)) \qquad \text{or} \qquad O(n\exp(w)), \text{ for } O(n) \text{ edges.} \tag{5.5}$$

5.1.3 Polytree / Belief Propagation

Runtime. Define k as the max number of parents in the poly tree, then k is the same as the width of elimination tree. Let, also, n denote the number of nodes in the polytree. The algorithm has runtime

$$O(n\exp(k)) \tag{5.6}$$

5.1.4 Cut Set Conditioning

Time and Space (Total) Complexity.

$$O(n\exp(k)) \qquad \text{where } n = |N| \text{ and } k \text{ is width} \tag{5.7}$$

5.1.5 Any Space Recursive Cut Set

	no cache	all cache	Δ *no* $\rightarrow$ *all*
space	$O(wn)$	$O(n \exp(w))$	$\uparrow$
time	$O(n \exp(w \log n)))$	$O(n \exp(w))$	$\downarrow$

6 Compiling Bayesian Networks

6.1 Network Polynomials

The network polynomial is a summation over all instantiations of a network,

$$f \triangleq \sum_{z} \prod_{\theta_{x|u} \sim z} \theta_{x|u} \prod_{\lambda_x \sim z} \lambda_x \tag{6.1}$$

6.2 AC Properties

AC Size. of an AC is defined as the number of edges in the circuit.

AC Complexity. is the size of smallest AC that represents the network polynomial.

Decomposable. At each $\star$ node, we need

$$vars(AC_A) \cap vars(AC_B) = \emptyset \tag{6.2}$$

Deterministic. At each + node, we require at most one positive input is non-zero for all *complete instantiation*.

Smooth. At each + node, we require

$$vars(AC_A) = vars(AC_B) \tag{6.3}$$

AC for Marginals. requires decomposable and smooth. This guarantees that sub-circuits are of complete variable instantiations.

AC for Marginals and MPE. requires all three above: decomposable, deterministic, and smooth. The additional determinism guarantees a 1-to-1 mapping between sub-circuits and complete variable instantiations.

6.3 AC Derivative Probabilistic Implications

$$\frac{\partial f}{\partial \lambda_{\mathbf{x}}}(\mathbf{e}) = \Pr(\mathbf{x}, \mathbf{e} - X) \tag{6.4}$$

and

$$\theta_{\mathbf{x}|\mathbf{u}} \frac{\partial f}{\partial \theta_{\mathbf{x}|\mathbf{u}}}(\mathbf{e}) = \Pr(\mathbf{x}, \mathbf{u}, \mathbf{e}) \tag{6.5}$$