Lecture 2
Section 1: Probabilistic Logic

Fabio G. Cozman
Universidade de São Paulo - Brazil

Probabilistic Logic

- Nilsson (1986): set of sentences $\mathbb{P}(A) \geq \alpha$.
- Solution by linear programming.
- Done before by Hailperin (1965) and Gilio (1980).

Probabilistic satisfiability (Boole 1854)

- Propositional formula ϕ:
  1. propositions
  2. operators ($\neg$, $\wedge$, $\vee$, $\rightarrow$).
- Take Ω as the set of 2^n truth assignments for *n* propositions.
- Interpret $\mathbb{P}(\phi) \geq \alpha$ as

$$\sum_{\omega \models \phi} \mathbb{P}(\omega) \geq \alpha.$$

Probabilistic satisfiability (PSAT)

- Given *m* assessments, is there a probability measure over Ω?
  - Each assessments is a linear constraint.
  - Must satisfy $\mathbb{P}(\omega) \geq 0$ and $\sum_{\omega \in \Omega} \mathbb{P}(\omega) = 1$.
- This is a *linear program*!
- Somewhat surprisingly, NP-complete problem.

Example

- Consider assessments:
  - $\mathbb{P}(A) \geq \alpha$.
  - $B \to C$.
  - $\mathbb{P}(B) = \beta$.
- Can you give bounds for $\mathbb{P}(A \wedge B \wedge C)$?

Solution

$\mathbb{P}(A) \geq \alpha$, $B \to C$, $\mathbb{P}(B) = \beta$.

ω_i	A	B	C
1	0	0	0
2	0	0	1
3	0	1	0
4	0	1	1
5	1	0	0
6	1	0	1
7	1	1	0
8	1	1	1

Then ω_3 and ω_7 are impossible; thus

$$p_5 + p_6 + p_8 = \alpha, \quad p_4 + p_8 = \beta, \quad p_i \geq 0, \quad \sum_i p_i = 1.$$

Boole's challenge problem

1. Binary A_1, A_2, A_3.
2. Assessments: $\mathbb{P}(A_1) \in [l_1, u_1]$, $\mathbb{P}(A_2) \in [l_2, u_2]$, $\mathbb{P}(A_1 \wedge A_3) \in [l_3, u_3]$, $\mathbb{P}(A_2 \wedge A_3) \in [l_4, u_4]$, $\mathbb{P}(\neg A_1 \wedge \neg A_2 \wedge A_3) = 0$.
3. Then find $\mathbb{P}(A_3)$:

$$\max / \min \; p_1 + p_3 + p_5 + p_7, \quad \text{subject to}$$
$$p_1 + p_2 + p_3 + p_4 = \pi_1, \quad p_1 + p_2 + p_5 + p_6 = \pi_2,$$
$$p_1 + p_3 = \pi_3, \quad p_1 + p_5 = \pi_4,$$
$$p_7 = 0, p_1 + \cdots + p_8 = 1, l_i \leq \pi_i \leq u_i, p_k \geq 0.$$

ω_i	A_1	A_2	A
8	0	0	0
7	0	0	1
6	0	1	0
5	0	1	1
4	1	0	0
3	1	0	1
2	1	1	0
1	1	1	1

Using the language of events

- Given m assessments over events H_i, how about the probability for some other event H_0?
- Theorem: $\mathbb{P}(H_0)$ belongs to an interval with constraints given by other assessments.
- This is a linear program (Gilio 1980).

Example (Coletti and Scozzafava 1999)

- Take H_1, H_2, H_3.
- Assume $H_3 \subset H_1^c \cap H_2$.
- Assessments $\mathbb{P}(H_1) = 1/2$, $\mathbb{P}(H_2) = 1/5$, $\mathbb{P}(H_3) = 1/8$.

Build linear program.

Example (Coletti and Scozzafava 1999)

- Take H_1, H_2, H_3.
- Assume $H_3 \subset H_1^c \cap H_2$.
- Assessments $\mathbb{P}(H_1) = 1/2$, $\mathbb{P}(H_2) = 1/5$, $\mathbb{P}(H_3) = 1/8$.

Build linear program.

- $x_1 = \mathbb{P}(A_1)$; $A_1 = H_1 \cap H_2 \cap H_3^c$.
- $x_2 = \mathbb{P}(A_2)$; $A_2 = H_1 \cap H_2^c \cap H_3^c$.
- $x_3 = \mathbb{P}(A_3)$; $A_3 = H_1^c \cap H_2 \cap H_3^c$.
- $x_4 = \mathbb{P}(A_4)$; $A_4 = H_1^c \cap H_2 \cap H_3$.
- $x_5 = \mathbb{P}(A_5)$; $A_5 = H_1^c \cap H_2^c \cap H_3^c$.

Example (Coletti and Scozzafava 1999)

- Take H_1, H_2, H_3.
- Assume $H_3 \subset H_1^c \cap H_2$.
- Assessments $\mathbb{P}(H_1) = 1/2$, $\mathbb{P}(H_2) = 1/5$, $\mathbb{P}(H_3) = 1/8$.

Build linear program.

$$
\begin{array}{ccccccccccc}
x_1 & + & x_2 & & & & & & & = & 1/2 \\
x_1 & & & + & x_3 & + & x_4 & & & = & 1/5 \\
& & & & & & x_4 & & & = & 1/8 \\
x_1 & + & x_2 & + & x_3 & + & x_4 & + & x_5 & = & 1 \\
x_1 & \geq 0, & x_2 & \geq 0, & x_3 & \geq 0, & x_4 & \geq 0, & x_5 & \geq 0. &
\end{array}
$$

Conditional probabilities

- Assessment $\mathbb{P}(A|B) \geq \alpha$.
- Transform to (Hailperin (1965) and many others later):

$$\mathbb{P}(A \wedge B) \geq \alpha \mathbb{P}(B) \,.$$

- Or use the language of events.
- Still a linear program!

Example

- Take H_1, H_2, H_3.
- Assume $H_3 \subset H_1^c \cap H_2$.
- Assessments $\mathbb{P}(H_1) = 1/2$, $\mathbb{P}(H_2) = 1/5$, $\mathbb{P}(H_3) = 1/8$.
- Also, $\mathbb{P}(H_2 | H_1 \cup H_2) \geq 1/2$.

Build linear program.

Computing probabilities

1. In practice, column generation linear programming.
2. Conditional probabilities:
   - Charnes-Cooper transformation.
   - Dinkelbach-Jagannahan algorithm (Walley's generalized Bayes rule; Lavine's algorithm).
3. Imprecise likelihood / priors can be reduced to linear programming.

Column generation

- Probabilistic satisfiability is

$$\min 0p \quad \text{subject to} \quad Ap \geq \alpha, p \geq 1.$$

- General problem minimizes cp.
- The difficulty is that p has 2^n elements (for a problem with n propositions).
- The usual technique is *column generation*.
  - That is, generate only those columns of A that are necessary
    - (at any given time, simplex only needs m columns where m is number of lines of A).

The mechanics of column generation

- Use the revised simplex algorithm.
  - That is, keep only a basis ($m \times m$).
  - Must decide whether to bring a column into the basis.
- Then choose the column using a nonlinear subproblem:
  - Solve $\min_j c_B A_B^{-1} A_j$.
  - Note that A_j contains a set of logical formulas.
  - This is a MAXSAT problem.
  - Replace:

$$X \wedge Y \doteq XY, \quad X \vee Y \doteq X + Y - XY, \quad \neg X \doteq 1 - X.$$

  - It can be reduced to *linear (integer) programming*!

Integer programming

- *Very* useful fact:
  - Consider product $a \times b$, where
    - $a \in [0, 1]$.
    - b is either 0 or 1.
  - Create a new variable c, replace $a \times b$ by c and add

$$0 \leq c \leq b;$$

$$a - 1 + b \leq c \leq a.$$

- Now solve by linear (integer) programming!

PSAT with column generation

- Best results in the literature: hundreds of propositions, hundreds of assessments (Perron et al 2004), using lots of special tricks.
- There are also a few special cases that are "easy" and several variants, etc.
  - For instance, when formulas can be put in a "tree" structure (Andersen & Pretolani 1999).
  - Also if formulas can be organized in junction trees (van der Gaag 1991).
- (Also, approximation methods based on local search for large problems, but really no guarantees yet...)

Computing conditional probabilities

- Now suppose we wish $\underline{\mathbb{P}}(A|B) = \min \mathbb{P}(A|B)$.
- This is not a linear program (it is a linear fractional program).
- However, it can be solved through linear programming:
  - Charnes-Cooper transformation (similar solutions by White, Snow).
  - Dinkelbach-Jagannathan algorithm (similar solutions by Walley, Lavine).

Charnes-Cooper transformation

- Wish to solve:

$$\min_p \frac{\sum_i f_i \alpha_i p_i}{\sum_i \alpha_i p_i} \quad \text{s.t. } Ap \geq 0, \sum_i p_i = 1, p_i \geq 0.$$

  where $\sum_i \alpha_i p_i > 0$.
- Change variables to

$$q_i = \frac{p_i}{\sum_i \alpha_i p_i}.$$

- Now:

$$\min_q \sum_i f_i \alpha_i q_i \quad \text{s.t. } Aq \geq 0, \sum_i \alpha_i q_i = 1, q_i \geq 0.$$

Exercise

- Take H_1, H_2, H_3.
- Assume $H_3 \subset H_1^c \cap H_2$.
- Assessments $\mathbb{P}(H_1) = 1/2$, $\mathbb{P}(H_2) = 1/5$, $\mathbb{P}(H_3) = 1/8$.

Build linear program to compute $\underline{\mathbb{P}}(H_1 | H_1 \cup H_2)$, applying the Charnes-Cooper transformation.

Solution

- Take H_1, H_2, H_3, assume $H_3 \subset H_1^c \cap H_2$.
- Assessments $\mathbb{P}(H_1) = 1/2$, $\mathbb{P}(H_2) = 1/5$, $\mathbb{P}(H_3) = 1/8$.

Build linear program to compute $\underline{\mathbb{P}}(H_1 | H_1 \cup H_2)$.

First,

$$\min (x_1 + x_2)/(x_1 + x_2 + x_3 + x_4) \quad \text{s.t.}$$

$$x_1 + x_2 = 1/2; \quad x_1 + x_3 + x_4 = 1/5; \quad x_4 = 1/8; \quad x_i \geq 0; \quad \sum_i x_i = 1.$$

Then

$$\min (x_1 + x_2)/(x_1 + x_2 + x_3 + x_4) \quad \text{s.t.}$$

$$x_1/2 + x_2/2 - x_3/2 - x_4/2 - x_5/2 = 0; \quad 4x_1/5 - x_2/5 + 4x_3/5 + 4x_4/5 - x_5/5 = 0;$$

$$-x_1/8 - x_2/8 - x_3/8 + 7x_4/8 - x_5/8 = 0; \quad x_i \geq 0; \quad \sum_i x_i = 1.$$

Solution

- Take H_1, H_2, H_3, assume $H_3 \subset H_1^c \cap H_2$.
- Assessments $\mathbb{P}(H_1) = 1/2$, $\mathbb{P}(H_2) = 1/5$, $\mathbb{P}(H_3) = 1/8$.

Build linear program to compute $\underline{\mathbb{P}}(H_1 | H_1 \cup H_2)$.

First,

$$\min(x_1 + x_2)/(x_1 + x_2 + x_3 + x_4) \quad \text{s.t.}$$

$$x_1 + x_2 = 1/2; \quad x_1 + x_3 + x_4 = 1/5; \quad x_4 = 1/8; \quad x_i \geq 0; \quad \sum_i x_i = 1.$$

Then

$$\min(y_1 + y_2) \quad \text{s.t.}$$

$$y_1/2 + y_2/2 - y_3/2 - y_4/2 - y_5/2 = 0; \quad 4y_1/5 - y_2/5 + 4y_3/5 + 4y_4/5 - y_5/5 = 0;$$

$$-y_1/8 - y_2/8 - y_3/8 + 7y_4/8 - y_5/8 = 0; \quad y_i \geq 0; \quad \sum_{i=1}^{4} y_i = 1.$$

Dinkelbach-Jagannathan for probability

- Note:
$$\lambda = \min \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)},$$
iff
$$\min\left(\mathbb{P}(A \cap B) - \lambda \mathbb{P}(B)\right) = 0,$$
assuming $\mathbb{P}(B) > 0$.
- The left side is strictly decreasing function of λ.
- So, we can bracket λ.

Dinkelbach-Jagannathan for expectation

- Also,
$$\lambda = \min \frac{E[f(X)B]}{\mathbb{P}(B)},$$
iff
$$\min \left(E[f(X)B] - \lambda \mathbb{P}(B)\right) = 0$$
or, rather,
$$\min E[(f(X) - \lambda)B] = 0;$$
that is,
$$\underline{E}[(f(X) - \lambda)B] = 0.$$
- This is Walley's Generalized Bayes Rule (GBR).
  - Walley proposed iteration:
    $\mu_{i+1} = \mu_i + 2\underline{E}[(f(X) - \mu_i)B] / (\overline{\mathbb{P}}(B) + \underline{\mathbb{P}}(B))$.

Lavine's algorithm

- In 1991, Lavine published a paper on robust statistics with the same algorithm, apparently unaware of the literature.
- Lavine's algorithm became quite popular.
- Until Lavine's algorithm, calculation of posterior lower expectations in robust statistics usually relied on very special arguments.
  - Often, minimax theory.

Now, imprecise likelihoods

- Suppose we have $\mathbb{K}(X)$ ("prior") and $\mathbb{K}(Y|X=x)$ for each x ("likelihood").
- Suppose $\mathbb{K}(Y|X=x)$ is *separately specified* (important condition!).
- If $\underline{\mathbb{P}}(Y=y)>0$, $\underline{E}[f(X)|Y=y]$ is the unique solution of the equation

$$\underline{E}[(f(X)-\lambda)p_\lambda(y|X)]=0,$$

where

$$p_\lambda(y|X)=\begin{cases}\underline{E}[y|x] & \text{if} \quad f(x)\geq\lambda\\ \overline{E}[y|x] & \text{if} \quad f(x)<\lambda\end{cases}$$

Dealing with imprecise likelihoods

$$\underline{E}[f(X)|Y=y] = \min_{p',p''} \left[\frac{\sum_i (f_i L_y(x_i) p'_i + f_i U_y(x_i) p''_i)}{\sum_j \left(L_y(x_j) p'_j + U_y(x_j) p''_j \right)} \right],$$

subject to:

$$A(p' + p'') \leq 0,$$

$$\sum_i (p'_i + p''_i) = 1, \qquad p'_i \geq 0, p''_i \geq 0.$$

Example based on White (1986)

- Variable with 4 values $\{\theta_1, \theta_2, \theta_3, \theta_4\}$,

$$2.5p(\theta_1) \geq p(\theta_4) \geq 2p(\theta_1)\,,$$

$$10p(\theta_3) \geq p(\theta_2) \geq 9p(\theta_3)\,, \quad p(\theta_2) = 5p(\theta_4)\,.$$

- Also, bounds on likelihood:

$$\begin{array}{ll} L(x|\theta_1) = 0.9, & L(x|\theta_2) = 0.1125, \\ L(x|\theta_3) = 0.05625, & L(x|\theta_4) = 0.1125, \\ U(x|\theta_1) = 0.95, & U(x|\theta_2) = 0.1357, \\ U(x|\theta_3) = 0.1357, & U(x|\theta_4) = 0.1357. \end{array}$$

Example: solution

$\underline{\mathbb{P}}(\theta_1|x) = \min_{p',p''}(0.9p'_1 + 0.95p''_1)$,
$p' \geq 0,\ p'' \geq 0$,

$$\begin{bmatrix} -\frac{5}{2} & 0 & 0 & 1 \\ 2 & 0 & 0 & -1 \\ 0 & -1 & 0 & 5 \\ 0 & 1 & 0 & -5 \\ 0 & -1 & 9 & 0 \\ 0 & 1 & -10 & 0 \end{bmatrix} [p' + p''] \leq 0,$$

$F_1\alpha' + F_2\alpha'' = 1$, where
$F_1 = [0.9, 0.1125, 0.0562, 0.1125], F_2 = [0.95, 0.1357, 0.1357, 0.1357]$.
By linear programming: $\underline{\mathbb{P}}(\theta_1|x) = 0.2881$.

Independence relations

1. Introduce independence so as to have stronger constraints.
   - A and B independent, $\mathbb{P}(A) = \mathbb{P}(B) = 1/2$; then $\mathbb{P}(A \wedge B) = 1/4$.
2. Independence leads to
   - *nonlinear* constraints,
   - open problems concerning complexity.
3. Idea: organize independence relations using graphs.
   - This will take us to credal networks and the like; this is for **other** talks...

First-order probabilistic logic

- Now we have constants, relations, functions, quantifiers:
  man(Socrates) $\vee$ mortal(Socrates)
  $\forall x$: man(x) $\rightarrow$ mortal(x).
- Nilsson (1986) advocated: $\mathbb{P}(\phi) \geq \alpha$ where ϕ is sentence.
  - Solve by linear programming... but there are *decidability* questions.
  - Recent study by Jaumard et al (2007) for decidable fragments.

Example (Jaumard et al 2007)

Assessments:

- $\mathbb{P}(\forall x : \exists y : t(x, y) \wedge s(y)) = 0.9$.
- $\mathbb{P}(\exists x : \neg r(x)) = 0.6$.
- $\mathbb{P}(\exists y : \neg s(y)) = 0.6$.
- $\mathbb{P}(\forall x : \forall y : \neg t(x, y) \wedge r(x) \wedge s(y)) = 0.7$.

Compute $\mathbb{P}(\exists x : \exists y : \neg t(x, y))$.

- Only 12 "possible worlds".
- Possible to apply linear program; extension to column generation method is open problem.

Things one might say

- $\mathbb{P}(\text{fly}(\text{Tweety})) \geq 0.9$.
- $\mathbb{P}(\forall x : \text{Bird}(x) \rightarrow \text{Fly}(x)) > 0.9$.
- And more....
- $\forall x : \mathbb{P}(\text{Bird}(x) \rightarrow \text{Fly}(x)) > 0.9$.
- $\forall x : \mathbb{P}(\text{Fly}(x)|\text{Bird}(x)) > 0.9$.
- And... $||\text{Fly}(x)|\text{Bird}(x)||_x > 0.9$.

Zero probabilities

- This is one of the most embarrassing challenges in the world of credal sets.
- In the standard theory of probabilities, it is easy to ignore null events (events with probability zero).
  - Such events “will never happen”.

Zero probabilities

- But there may be events with zero lower probability and nonzero upper probability.
  - For instance, if $\overline{\mathbb{P}}(B) \leq \alpha$, then $\mathbb{P}(B)$ may be zero.
- So, we may observe A and we may need to do something about $\mathbb{P}(A|B)$.
- Linear fractional programs "discard" distributions.

Full conditional measures

- The most elegant solution is to consider *full probability probabilities*.
- A full probability probability is a function $\mathbb{P}(\cdot|\cdot)$ on $\mathcal{E} \times \mathcal{E} \backslash \emptyset$ where $\mathcal{E}$ is an algebra of events, such that
  - $\mathbb{P}(\Omega|C) = 1$;
  - $\mathbb{P}(A|C) \geq 0$ for all A;
  - $\mathbb{P}(A \cup B|C) = \mathbb{P}(A|C) + \mathbb{P}(B|C)$ when $A \cap B = \emptyset$;
  - $\mathbb{P}(A \cap B|C) = \mathbb{P}(A|B \cap C)\,\mathbb{P}(B|C)$ when $B \cap C \neq \emptyset$.
- Full probability probabilities allow $\mathbb{P}(A|C)$ to be defined even if $\mathbb{P}(C) = 0$!

The Krauss-Dubins representation

- We can partition Ω into events $L_0, \ldots, L_K$, $K \leq N$,
- such that the full conditional probability is represented as a sequence of strictly positive probability measures $\mathbb{P}_0, \ldots, \mathbb{P}_K$, where the support of $\mathbb{P}_i$ is restricted to L_i.
- $\mathbb{P}(A|B) = \mathbb{P}(A|B \cap L_B)$, where L_B is the "layer" where B has nonzero probability.
- This representation has been advocated by Coletti & Scozzafava.

Example (note: $\mathbb{P}(A) = 0$, but $\mathbb{P}(B|A) = \beta$):

	A	A^c
B	0	α
B^c	0	$1-\alpha$

	A	A^c
B	β	
B^c	$1-\beta$	

Exercise

Consider assessments:

- $\mathbb{P}(A) \geq 1/2$.
- $\mathbb{P}(A^c \cap B^c) = 1/2$.
- $\mathbb{P}(C|A^c \cap B) = 1/3$.

What is the set of Krauss-Dubins representations?
What is $\mathbb{P}(C|B)$?
What is $\mathbb{P}(C^c|A^c \cap B)$?

Coletti-Scozzafava's method

- Run the usual linear program with assessments $\mathbb{P}(A_i|B_i) \geq \alpha_i$.
- If all B_i have $\mathbb{P}(B_i) > 0$ for all feasible solutions, stop (solution has been found).
- Otherwise:
  - Collect those B_i with $\mathbb{P}(B_i) = 0$ for all feasible solutions.
  - Then build another linear program *only* with those assessments with these B_i.
  - Repeat until there are no more assessments (inference is vacuous).

Improving the algorithm

- Coletti-Scozzafava's method has been optimized and expanded by Vantaggi, Capotorti and others.
  - Idea is to quickly detect/exploit zero probabilities.
  - Check coherence (CkC) package:
    http://www.dipmat.unipg.it/~upkd/paid/software.html
  - Vantaggi has dealt with independence as well.

- Overall, many tests to make, to detect whether events may are null.

Other approaches

- Sequence of $2m$ direct linear programs in the worst case (Walley, Pelessoni, Vicig (1999, 2004)).
  - But still, necessary to run additional linear programs to check whether to proceed.
- All of this is to check "coherence" in a strong sense.
  - There are weaker concepts of "coherence".