Credal Networks and the Like

Fabio G. Cozman
Universidade de São Paulo

ISIPTA 99:

Back then:

Friday 2 July 1999

⋮

17:00 - 18:00 Discussion session 4

The wonderful world of Imprecise Probabilities: Ideas towards a more integrated, more caring and more sharing community

(Chairman: Fabio G. Cozman)

The Wonderful World of Credal Networks

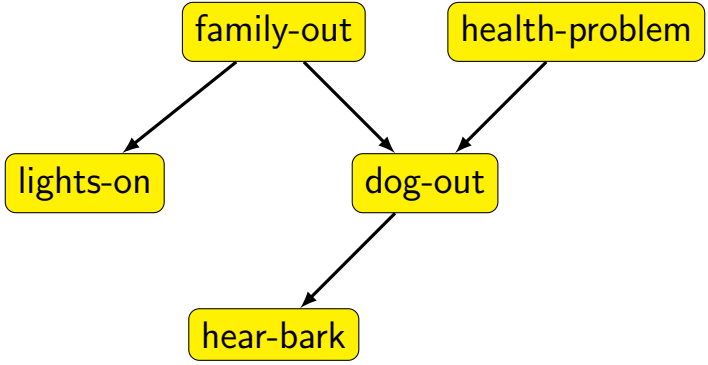
Fabio G. Cozman
Universidade de São Paulo

Artificial Intelligence

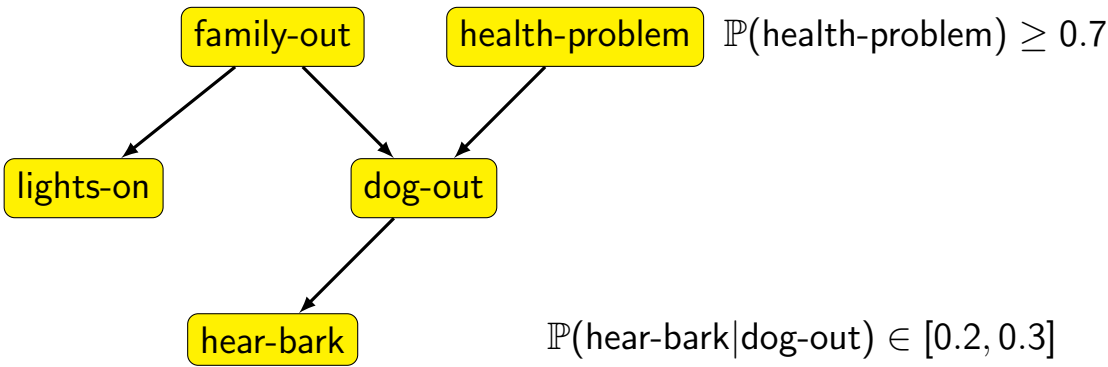
- Always an interest in search and decision making, learning from data... with a distinctive interest in *representation*.
- In the beginning, a resistance to probabilities.
- Back in the 80s: rules and uncertainty.

Bayesian networks

- Simplistic: a compact way to express joint distributions.
- A representation tool for reasoning about uncertainty.



Credal networks: a most reasonable move



Moral and colleagues, 1990-1991. Tessem 1992.

One can imagine general "local" pieces, but let's keep it simple.

Why was this reasonable?

- To guarantee robustness (from Robust Statistics).
- To accommodate incomplete/conflicting experts, scarce data.
- To accommodate assessments/beliefs expressed in many languages, some of them offered as alternatives to probability.
- A dream: that by looking at "coarser" models, one would simplify reasoning and decision making.

Extensions

- Strong extensions: combining all local pieces, getting a set of Bayesian networks, taking the convex hull.
  - Is this the right thing to do? Long debates back in the 90s.
- Walley's perspective: be flexible, let many extensions bloom.
  - Epistemic irrelevance, epistemic independence.

What is known about strong(/complete) extensions?

What is known about strong(/complete) extensions?

- They satisfy d-separation.

What is known about strong(/complete) extensions?

- They satisfy d-separation.
- Inference algorithms (there are others for decision-making...):
  - For polytrees with binary variables: 2U.
  - For everything else, there are two strategies:
    - One may search for local vertices (if local sets are given as lists of vertices), by propagating sets of functions, by setting up an integer program.
    - One may set up an optimization problem and use multilinear programming on it.
  - Approximations can be produced from each one of these methods.

What is known about strong(/complete) extensions?

- They satisfy d-separation.
- Inference algorithms (there are others for decision-making...):
  - For polytrees with binary variables: 2U.
  - For everything else, there are two strategies:
    - One may search for local vertices (if local sets are given as lists of vertices), by propagating sets of functions, by setting up an integer program.
    - One may set up an optimization problem and use multilinear programming on it.
  - Approximations can be produced from each one of these methods.
- Complexity of inference:
  - P for 2U, for some Naive credal classifiers, and for limited HMMs;
  - NP for polytrees (and bounded treewidth, for which there is FPTAS);
  - $\mathrm{NP}^{\mathrm{PP}}$ in general.

What is known about strong(/complete) extensions?

- They satisfy d-separation.
- Inference algorithms (there are others for decision-making...):
  - For polytrees with binary variables: 2U.
  - For everything else, there are two strategies:
    - One may search for local vertices (if local sets are given as lists of vertices), by propagating sets of functions, by setting up an integer program.
    - One may set up an optimization problem and use multilinear programming on it.
  - Approximations can be produced from each one of these methods.
- Complexity of inference:
  - P for 2U, for some Naive credal classifiers, and for limited HMMs;
  - NP for polytrees (and bounded treewidth, for which there is FPTAS);
  - $\mathrm{NP}^{\mathrm{PP}}$ in general.

(Gentle tutorial: Chapter in 2014 IP book.)

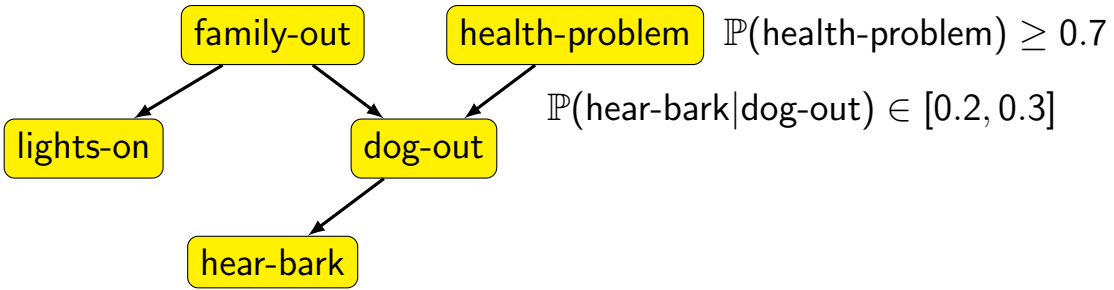
Applications, some learning

- Classification of debris, assessment of banking risks, decision support for military planning and farming practices.
- Versions of Naive Bayes (and similar) classifiers, with focus on detecting unreliable decisions.

- On learning: parameter learning using IDM and similar ideas; a few adapted algorithms for structure learning.

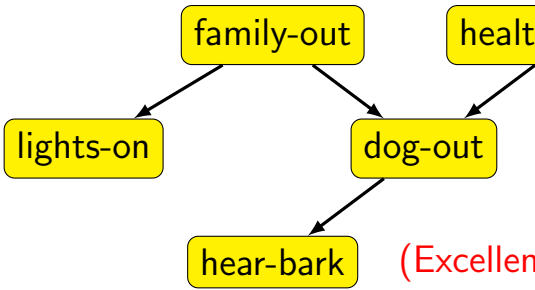
Epistemic irrelevance (... and others??)

- Condition: $\underline{\mathbb{E}}[f(X)|y,z] = \underline{\mathbb{E}}[f(X)|z]$ (not symmetric!).
- General algorithm based on linear programming.
- Polynomial algorithm for trees (and some other networks).
- Adapted "d-separation".



Epistemic irrelevance (... and others??)

- Condition: $\underline{\mathbb{E}}[f(X)|y,z] = \underline{\mathbb{E}}[f(X)|z]$ (not symmetric!).
- General algorithm based on linear programming.
- Polynomial algorithm for trees (and some other networks).
- Adapted "d-separation".



$\mathbb{P}(\text{health-problem}) \geq 0.7$

$\mathbb{P}(\text{hear-bark}|\text{dog-out}) \in [0.2, 0.3]$

(Excellent tutorial: Jasper's IJAR 2017 paper.)

Future?

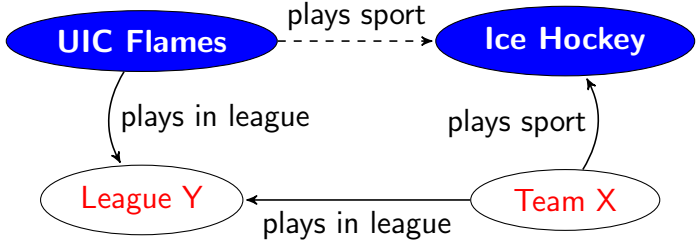
- Certainly, better algorithms and implementations.
  - Other "inference" problems, other concepts of independence.
  - Better approximate algorithms: waiting for "MCMC of credal networks".

But let's think ahead...

- Mere effect of priors is less important when big data is there; mere model simplification is less important when computers can process big data.
- Focus should be on small-data settings, matters of life and death.
- Opportunities lie in the *representation power* of *credal sets* (or lower previsions/desirability/choice functions) purpose is is to make things work...

An example: conversational agents

- Needed in the banking industry: chatbots that can really explain decisions.
- An example of explanation for link completion in the NELL knowledge base:



- Perhaps we need epistemic argumentation (Polberg, Hunter).

Some final thoughts

- We already know a lot about credal networks.
- We can certainly deal with small-data settings, matters of life and death.
- Opportunities lie in the *representation power* of *credal sets* (or XXX).
- AI is now a very practical thing, XXX should be a module within a larger system whose purpose is is to make things work, doing approximate reasoning whenever necessary.