Knowledge Representation: Probabilistic Graphs

Alexia Toumpa, Brandon Bennett

Knowledge Representation and Reasoning Module

November 21, 2019

Slides based on material from Daphne Koller, Adrian Weller, Zoubin Ghahramani, Henrik I. Christensen, and Karteek Alahari.

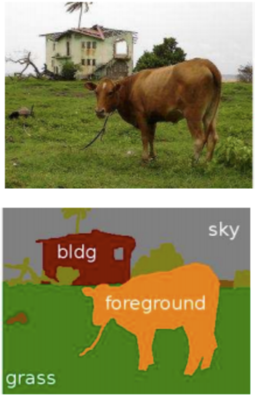
What is this class about ?

Introduction to **probabilistic graphical models** and modeling the uncertainty.

Introduction to ***ProbLog* Framework** .

Motivation

Image Segmentation



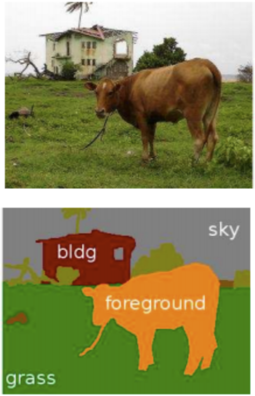
e.g., [He et al., 2004; Shotton et al., 2006; Gould et al., 2009]

Task : pixels $\rightarrow$ pixel labels {building, grass, cow, sky}

- many variables
- uncertainty of the correct answer, *i.e.* label

Motivation

Image Segmentation



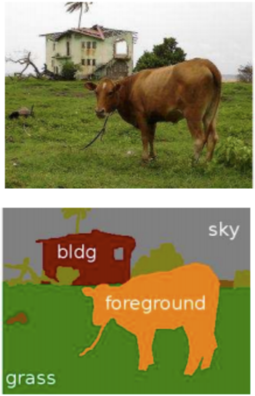
e.g., [He et al., 2004; Shotton et al., 2006; Gould et al., 2009]

Task : pixels $\rightarrow$ pixel labels {building, grass, cow, sky}

- many variables
- uncertainty of the correct answer, *i.e.* label

Motivation

Image Segmentation



e.g., [He et al., 2004; Shotton et al., 2006; Gould et al., 2009]

Task : pixels $\rightarrow$ pixel labels {building, grass, cow, sky}

- many variables
- uncertainty of the correct answer, *i.e.* label

Probabilistic Graphical Models provide a framework to address these problems.

Why probabilistic ?

To model **uncertainty**, due to :

- partial knowledge of state of the world
- noisy observations
- cases/phenomena not observed by the model

Probabilistic theory provides :

- standalone representation with clear semantics
- reasoning patterns
- learning methods

Why probabilistic ?

To model **uncertainty**, due to :

- partial knowledge of state of the world
- noisy observations
- cases/phenomena not observed by the model

Probabilistic theory provides :

- standalone representation with clear semantics
- reasoning patterns
- learning methods

Why probabilistic ?

To model **uncertainty**, due to :

- partial knowledge of state of the world
- noisy observations
- cases/phenomena not observed by the model

Probabilistic theory provides :

- standalone representation with clear semantics
- reasoning patterns
- learning methods

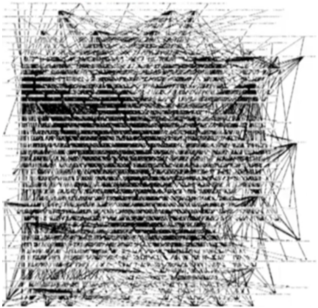
Why graphical ?

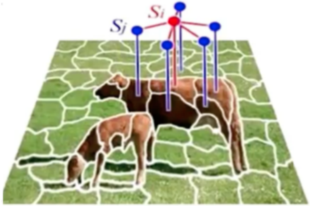
Intersection of ideas from probability theory and computer science.

For representing large number of variables

Random variables $Y_1, Y_2, \ldots Y_n$

Graphical models capture uncertainty through joint distribution $P(Y_1, Y_2, \ldots, Y_n)$.





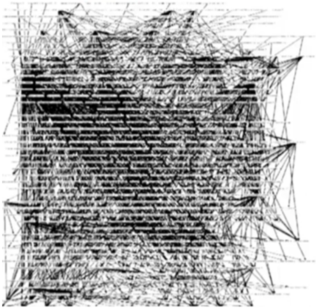
Why graphical ?

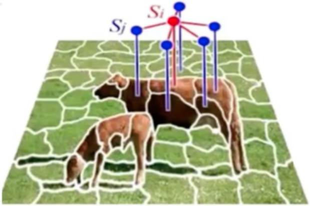
Intersection of ideas from probability theory and computer science.

For representing large number of variables

Random variables $Y_1, Y_2, \ldots Y_n$

Graphical models capture uncertainty through joint distribution $P(Y_1, Y_2, \ldots, Y_n)$ **.**





(Probabilistic) Graphical Models

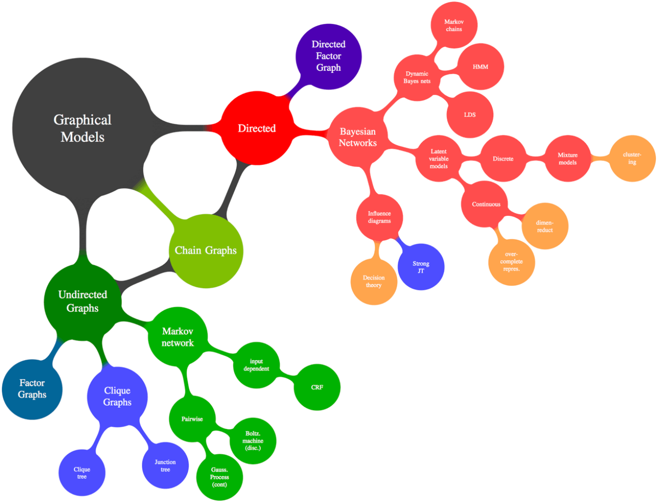


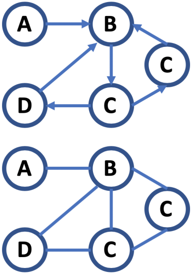
Figure – Image by David Barber

Graphs

The Basics

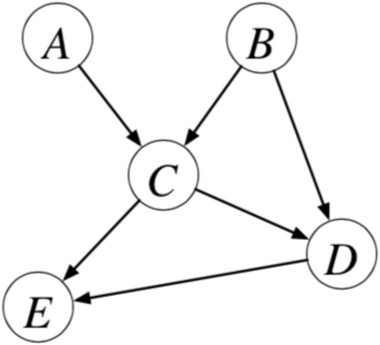
Basics concepts :

- Vertices/Nodes
- Edges
- Directed & Undirected
- Degree & In/Out Degree
- Neighbours & Parent/Child
- Walk, Path, and Cycle



Probabilistic Graphs

Knowledge Representation



Nodes represent **random variables**.

Edges represent **dependencies** between variables.

Directed Acyclic Graphical Model

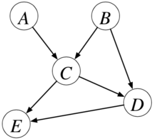
Bayesian Networks

A **Bayesian network** is specified by a DAG $= (V, \overrightarrow{E})$ with :

- one node $i \in V$ for each random variable X_i
- one conditional probability distribution (CPD) per node, $p(x_i | x_{Pa(i)})$, specifying the variable's probability conditioned on its parents' values

So the probability distribution which describes the DAG is :

$$p(x_i, ..., x_n) = \prod_{i \in V} p(x_i | x_{Pa(i)})$$



A DAG Model, as the one shown above, corresponds to the joint probability distribution :

$$p(A, B, C, D, E) = p(A)p(B)p(C|A, B)p(D|B, C)p(E|C, D)$$

Inference in a Graphical Model

Basic concepts

- Consider inference of $p(x, y)$ we can formulate this as :

$$p(x, y) = p(x|y)p(y) = p(y|x)p(x)$$

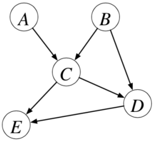
- We can further marginalize

$$p(y) = \sum_{x'} p(y|x')p(x')$$

- Using *Bayes Rule* we can reverse the inference

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}$$

Inference in a Graphical Model



Consider the graph shown above, which represents :

$$p(A,B,C,D,E) = p(A)p(B)p(C|A,B)p(D|B,C)p(E|C,D)$$

Inference : Evaluates the probability distribution over some set of variables, given the values of another set of variables.

For example, how can we compute $p(A|C=c)$? Assume each variable is binary.

Inference in a Graphical Model

$$p(A|C = c) = \frac{p(A,C=c)}{p(C=c)}$$

$p(C = c) = \sum_A p(A, C = c)$

Naive method :

$p(A, C = c) = \sum_{B,D,E} p(A, B, C = c, D, E)$

More efficient method :

$p(A, C = c)$

$= \sum_{B,D,E} p(A)p(B)p(C = c|A, B)p(D|B, C = c)p(E|C = c, D)$

$= \sum_B p(A)p(B)p(C = c|A, B) \sum_D p(D|B, C = c) \sum_E p(E|C = c, D)$

$= \sum_B p(A)p(B)p(C = c|A, B)$

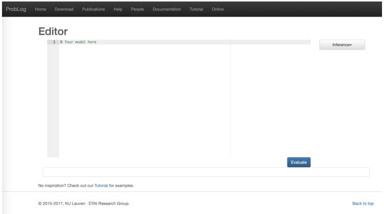
Introduction to *ProbLog*

Used for :

- computing marginal probabilities given evidences
- learn form partial interpretations
- solve inference tasks
- solve decision theoretic problems
- model uncertainty
- express complex probabilistic models

Installation

- Online Editor
  https://dtai.cs.kuleuven.be/problog/editor.html



- Import as a python package

```
pip3 install problog --user
```

- Standalone tool

ProbLog Program

A *ProbLog* program consists of two parts :

1. a set of ground probabilistic facts
   i.e. ***p : :f***
2. a logic program, a set of logic rules and non-probabilistic facts

More examples : https://dtai.cs.kuleuven.be/problog/tutorial/basic/08_rule_probs.html

Coding in *ProbLog*

Example *Heads*

Calculate the probability of both coins being head.

$$P(\text{allHeads}) = P(\text{heads1}) * P(\text{heads2})$$

```
% Probabilistic facts:
0.5::heads1.
0.6::heads2.

% Rules:
allHeads :- heads1, heads2.

% Queries:
query(allHeads).
```

Coding in *ProbLog*

Example *Heads*

Calculate the probability of neither of the coins being head.

$$P(\text{noHeads}) = (1 - P(\text{heads1})) * (1 - P(\text{heads2}))$$

```
% Probabilistic facts:
0.5::heads1.
0.6::heads2.

% Rules:
noHeads :- \+heads1, \+heads2.

% Queries:
query(noHeads).
```

Coding in *ProbLog*

Example *Heads*

Calculate the probability of at least one of the two coins is head; one of the two is bend(biased).

$$P(\text{someHeads}) = (1 - P(\text{noHeads}))$$
$$P(\text{noHeads}) = (1 - P(\text{heads1})) * (1 - P(\text{heads2}))$$

```
% Probabilistic facts:
0.5::heads1.
0.6::heads2.

% Rules:
someHeads :- heads1.
someHeads :- heads2.

% Queries:
query(someHeads).
```

Coding in *ProbLog*

Example *Smokers*

```
0.3::stress(X) :- person(X).
0.2::influences(X,Y) :- person(X), person(Y).
smokes(X) :- stress(X).
smokes(X) :- friend(X,Y), influences(Y,X), smokes(Y).

0.4::asthma(X) :- smokes(X).

person(angelika).
person(joris).
person(jonas).
person(dimitar).

friend(joris,jonas).
friend(joris,angelika).
friend(joris,dimitar).
friend(angelika,jonas).
```

Coding in *ProbLog*

Example *Smokers*

What is the probability of Dimitar smoking ?

```
0.3::stress(X) :- person(X).
0.2::influences(X,Y) :- person(X), person(Y).
smokes(X) :- stress(X).
smokes(X) :- friend(X,Y), influences(Y,X), smokes(Y).
0.4::asthma(X) :- smokes(X).

person(angelika).
person(joris).
person(jonas).
person(dimitar).

friend(joris,jonas).
friend(joris,angelika).
friend(joris,dimitar).
friend(angelika,jonas).

query(smokes(dimitar)).
```

Coding in *ProbLog*

Example *Smokers*

What is the probability of Dimitar smoking ?

```
0.3::stress(X) :- person(X).
0.2::influences(X,Y) :- person(X), person(Y).
smokes(X) :- stress(X).
smokes(X) :- friend(X,Y), influences(Y,X), smokes(Y).
0.4::asthma(X) :- smokes(X).

person(angelika).
person(joris).
person(jonas).
person(dimitar).

friend(joris,jonas).
friend(joris,angelika).
friend(joris,dimitar).
friend(angelika,jonas).

query(smokes(dimitar)).
```

Coding in *ProbLog*

Example *Smokers*

What is the probability of Dimitar smoking ?

```
0.3::stress(X) :- person(X).
0.2::influences(X,Y) :- person(X), person(Y).
smokes(X) :- stress(X).
smokes(X) :- friend(X,Y), influences(Y,X), smokes(Y).
0.4::asthma(X) :- smokes(X).

person(angelika).
person(joris).
person(jonas).
person(dimitar).

friend(joris,jonas).
friend(joris,angelika).
friend(joris,dimitar).
friend(angelika,jonas).

query(smokes(dimitar)).
```

$$P(\text{smokes}(\text{Dimitar})) = P(\text{stress}(\text{Dimitar}))$$

Coding in *ProbLog*

Example *Smokers*

What is the probability of Angelika smoking ?

```
0.3::stress(X) :- person(X).
0.2::influences(X,Y) :- person(X), person(Y).
smokes(X) :- stress(X).
smokes(X) :- friend(X,Y), influences(Y,X), smokes(Y).
0.4::asthma(X) :- smokes(X).

person(angelika).
person(joris).
person(jonas).
person(dimitar).

friend(joris,jonas).
friend(joris,angelika).
friend(joris,dimitar).
friend(angelika,jonas).

query(smokes(angelika)).
```

Coding in *ProbLog*

Example *Smokers*

What is the probability of Angelika smoking ?

```
0.3::stress(X) :- person(X).
0.2::influences(X,Y) :- person(X), person(Y).
smokes(X) :- stress(X).
smokes(X) :- friend(X,Y), influences(Y,X), smokes(Y).
0.4::asthma(X) :- smokes(X).

person(angelika).
person(joris).
person(jonas).
person(dimitar).

friend(joris,jonas).
friend(joris,angelika).
friend(joris,dimitar).
friend(angelika,jonas).

query(smokes(angelika)).
```

Coding in *ProbLog*

Example *Smokers*

What is the probability of Angelika smoking ?

```
0.3::stress(X) :- person(X).
0.2::influences(X,Y) :- person(X), person(Y).
smokes(X) :- stress(X).
smokes(X) :- friend(X,Y), influences(Y,X), smokes(Y).
0.4::asthma(X) :- smokes(X).

person(angelika).
person(joris).
person(jonas).
person(dimitar).

friend(joris,jonas).
friend(joris,angelika).
friend(joris,dimitar).
friend(angelika,jonas).

query(smokes(angelika)).
```

$$P(\text{smokes}(\text{Angelika})) =$$
$$1 - \left[\left(1 - P(\text{stress}(\text{Angelika}))\right) * \left(1 - \left(P(\text{influences}(\text{Jonas}, \text{Angelika})) * P(\text{smokes}(\text{Jonas}))\right)\right)\right]$$

Useful Resources

- *ProbLog* Editor : https://dtai.cs.kuleuven.be/problog/editor.html
- *ProbLog* python package installation : https://problog.readthedocs.io/en/latest/install.html#installing-python
- Python installation : https://docs.python.org/3.5/using/windows.html
- Use *ProbLog* as a python library : https://dtai.cs.kuleuven.be/problog/tutorial/advanced/01_python_interface.html
- *ProbLog* Documentation : https://problog.readthedocs.io/en/latest/
- *ProbLog* examples : https://dtai.cs.kuleuven.be/problog/tutorial/basic/01_coins.html