
**Outline**

- Introduction
- Reminder: Probability theory
- Basics of Bayesian Networks
- Modeling Bayesian networks
- Inference (VE, Junction tree)
- [Excourse: Markov Networks]
- Learning Bayesian networks
- Relational Models

---

### Bayesian Networks

1. Finite, acyclic graph
2. Nodes: (discrete) random variables
3. Edges: direct influences
4. Associated with each node: a table representing a conditional probability distribution (CPD), quantifying the effect the parents have on the node

\[ P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | \text{pa}(X_i)) \]

---

The "ICU alarm" network

- 37 binary random variables
- 509 parameters instead of \(2^{37} = 10^{12}\)
Bayesian Networks

- Effective (and to some extend efficient) inference algorithms
  - Variable elimination
  - Junction Trees
  - MAP
- Effective (and to some extend efficient) learning approaches
  - Expectation Maximization
  - Gradient Ascent

Knowledge Acquisition Bottleneck, Data cheap

Bayesian Networks: Problem

- Bayesian nets use propositional representation
- Real world has objects, related to each other

These “instances” are not independent!

Intelligence, Difficulty, Grade

How to Craft and Publish Papers

- Are there similar papers?
- Which papers are relevant?
- Keywords Extraction
- Does anybody know L. D. Raedt?
Blood Type / Genetics / Breeding

- 2 Alleles: A and a
- Probability of Genotypes AA, Aa, aa ?

Prior for founders

Offsprings

CEPH Genotype DB, http://www.cephb.fr/
Bongard’s Problems

Noise?

Some objects are opaque?
(e.g. in relation is not always observed)

... Other Application Areas

- Planning
- Activity Recognition
- Social Networks
- Biinformatics
- Natural Language Processing
- Robotics
- Data Cleaning
- Games

Why do we need relational models?

- Rich Probabilistic Models
- Comprehensibility
- Generalization (similar situations/individuals)
- Knowledge sharing
- Parameter Reduction / Compression
- Learning
  - Reuse of experience (training one RV might improve prediction at other RV)
  - More robust
  - Speed-up
When to apply relational models?

- When it is **impossible to elegantly** represent your problem in attribute value form
  - variable number of ‘objects’ in examples
  - relations among objects are important

<table>
<thead>
<tr>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
<th>A6</th>
</tr>
</thead>
<tbody>
<tr>
<td>true</td>
<td>true</td>
<td>?</td>
<td>true</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>?</td>
<td>true</td>
<td>?</td>
<td>?</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>true</td>
<td>false</td>
<td>?</td>
<td>false</td>
<td>true</td>
<td>?</td>
</tr>
</tbody>
</table>

Statistical Relational Learning

... deals with machine **learning** and data mining in **relational** domains where observations may be **missing, partially observed**, and/or **noisy** ...

and is one of the key open questions in AI.

BNs = Probabilistic Propositional Logic

\[ P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | pa(X_i)) \]

Logic Programming

\[
\begin{align*}
\text{father}(rex,fred). & \quad \text{mother}(ann,fred). \\
\text{father}(brian,doro). & \quad \text{mother}(utta,doro). \\
\text{father}(fred,henry). & \quad \text{mother}(doro,henry). \\
\text{pc}(rex,a). & \quad \text{mc}(rex,a). \\
\text{pc}(ann,a). & \quad \text{mc}(ann,b). \\
\end{align*}
\]

The maternal information \( mc/2 \) depends on the maternal and paternal \( pc/2 \) information of the mother \( \text{mother}/2 \):

\[
\begin{align*}
\text{mchrom}(fred,a). & \quad \text{mchrom}(fred,b), \\
\end{align*}
\]

or better

\[
\begin{align*}
\text{mc}(P,a) & \gets \text{mother}(M,P), \text{pc}(M,a), \text{mc}(M,a). \\
\text{mc}(P,a) & \gets \text{mother}(M,P), \text{pc}(M,a), \text{mc}(M,b). \\
\text{mc}(P,b) & \gets \text{mother}(M,P), \text{pc}(M,a), \text{mc}(M,b). \\
\end{align*}
\]

Placeholder Could be rex, freud, doro,
Use general rules with placeholders.

\[ \text{sameAuthor}(A1, A2) :\] nth-author-of(A1, P1),
\[ \text{sameTitle}(P1, P2),\]
\[ \text{nth-author-of}(A2, P2). \]

Relational Models
- Probabilistic Relational Models
- Bayesian Logic Programs
- Relational Markov networks
- Markov Logic

Probabilistic Relational Models (PRMs)
- Database theory
- Entity-Relationship Models
  - Attributes = RVs

Database
Table
Attribute

Earthquake
Burglary
Alarm
MaryCalls
JohnCalls

Bloodtype
M-chromosome
P-chromosome

Person
Bloodtype
M-chromosome
P-chromosome
Person

Table
Probabilistic Relational Models (PRMs)

Bayesian Networks

Advanced

PRM Application: Collaborative Filtering

- User preference relationships for products/information.
- Traditionally: single dyadic relationship between the objects.

Relational Naive Bayes

- Relational - Relational

[Koller, Pfeffer, Getoor]

[Getoor, Sahami; simplified representation]
Probabilistic Relational Models (PRMs)

- Database View
- Unique Probability Distribution over finite Herbrand interpretations
  - No self-dependency
- Discrete and continuous RV
- BN used to do inference
- Graphical Representation

Relational Models
- Probabilistic Relational Models
- Bayesian Logic Programs
- Relational Markov networks
- Markov Logic

Bayesian Logic Programs (BLPs)

Rule Graph
Bayesian Networks

Answering Queries

\[ P(\text{bt(ann)})? \]

Bayesian Network induced over least Herbrand model

Bayes’ rule

\[
P(\text{bt(ann)} \mid \text{bt(fred)}) = \frac{P(\text{bt(ann)}, \text{bt(fred)})}{P(\text{bt(fred)})}
\]

Combining Partial Knowledge

\[ \text{prepared(Student,Topic)} \mid \text{read(Student,Book)}, \text{discusses(Book,Topic)}. \]

\[ \text{passes(Student)} \mid \text{prepared(Student, bn)}, \text{prepared(Student, logic)}. \]

- variable number of parents for \text{prepared/2}
due to \text{read/2}
  - whether a student prepared a topic depends on the books she read
- CPD only for one book-topic pair
Combining Rules

\[ P(A|B) \text{ and } P(A|C) \]
\[ P(A|B,C) \]

**CR**

- Any algorithm which
  - has an empty output if and only if the input is empty
  - combines a set of CPDs into a single (combined) CPD
- E.g. noisy-or, regression, ...

Aggregates

Map multisets of values to summary values (e.g., sum, average, max, cardinality)

**Experiments**

**KDD Cup 2001 localization task**
- predict the localization based on local features and interactions
  - 862 training genes
  - 381 test genes
  - >1000 interactions
  - 16 classes

**WebKB**
- predict the type of web pages
  - 877 web pages from 4 CS department
  - 1516 links
  - 6 classes
Bayesian Networks

KDD Cup: Protein Localization

RFK (72.89%) better than Hayashi et al.’s KDD Cup 2001 winning nearest-neighbour approach (72.18%)

WebKB: Web Page Classification

Collective NB ~ PRMs [Getoor et al. 02]
RFK outperforms PRMs
PRM with structural uncertainty over the links, best acc. (68%) on Washington

Bayesian Logic Programs (BLPs)

• Unique probability distribution over Herbrand interpretations
  – Finite branching factor, finite proofs, no self-dependency
• Highlight
  – Separation of qualitative and quantitative parts
  – Functors
• Graphical Representation
• Discrete and continuous RV

Learning Tasks

• Parameter Estimation
  – Numerical Optimization Problem
• Model Selection
  – Combinatorial Search

Leave-one-university-out cross-validation
What is the data about?

RVs + States = (partial) Herbrand interpretation

Probabilistic learning from interpretations

Family(1)
- pc(brian)=b,
- bt(ann)=a,
- bt(brian)=?, bt(dorothy)=a

Family(2)
- bt(cecily)=ab,
- pc(henry)=a,
- mc(fred)=?, bt(kim)=a,
- pc(bob)=b

Family(3)
- pc(rex)=b,
- bt(doro)=a,
- bt(brian)=?

Background
- m(ann,dorothy),
- f(brian,dorothy),
- m(cecily,fred),
- f(henry,fred),
- f(fred,bob),
- m(kim,bob),
- ...

Parameter Estimation

Initial Parameters θ₀

Logic Program L

Expected counts of a clause

EM-algorithm: iterate until convergence

Current Model (M,k)

P( head(GI), body(GI) | DC )

M

DataCase

GI

P( body(GI) | DC )

M

DataCase

GI

Expectation Maximization

EM-algorithm:
iterate until convergence

Inference

Expected counts of a clause

Maximization

Update parameters (ML, MAP)
Model Selection

• Combination of ILP and BN learning
• Modify the general rules syntactically:
  – Add atoms: $b(X,a)$
  – Delete atoms
  – Unify placeholders: $m(X,Y) \rightarrow m(X,X)$
  – ...
• Add, (reverse, and) delete bunches of edges simultaneously

Example

Original program

```
mc(X) | m(M,X), mc(M), pc(M).
pct(X) | f(F,X), mc(F), pc(F).
b(X) | mc(X), pc(X).
```

Data cases

```
{mc(ann)=true, m(ann)=a, mc(ann)=?, f(eric,john)=true, pc(eric)=b, mc(eric)=a, 
m(ann,john)=ab, pc(john)=a, bt(john) = ? } 
...```

Initial hypothesis

```
mc(X) | m(M,X).
pct(X) | f(F,X).
b(X) | mc(X).
```
Bayesian Networks

Original program

mc(X) | m(M,X), mc(M), pc(M).
pc(X) | f(F,X), mc(F), pc(F).
bt(X) | mc(X), pc(X).

Initial hypothesis

mc(X) | m(M,X).
pc(X) | f(F,X).
bt(X) | mc(X).

Refinement

mc(X) | m(M,X). pc(X) | f(F,X). bt(X) | mc(X), pc(X).

Refinement

mc(X) | m(M,X). pc(X) | f(F,X). bt(X) | mc(X). pc(X).

Refinement

mc(X) | m(M,X). pc(X) | f(F,X). bt(X) | mc(X), pc(X).
Outline Relational Models

• Relational Models
  – Probabilistic Relational Models
  – Bayesian Logic Programs
  – Relational Markov networks
  – Markov Logic

Undirected Relational Models

• So far, directed graphical models only
• Impose acyclicity constraint

• Undirected graphical models do not impose the acyclicity constraint

Undirected Relational Models

• Two approaches
  – Relational Markov Networks (RMNs)
    • (Taskar et al.)
  – Markov Logic Networks (MLNs)
    • (Anderson et al.)
• Idea
  – Semantics = Markov Networks
  – More natural for certain applications
• RMNs \sim\ undirected PRM
• MLNs \sim\ undirected BLP

Markov Networks

• To each clique $c$, a potential $\phi_c$ is associated
• Given the values $v$ of all nodes in the Markov Network
  $$ P(v) = \frac{1}{Z} \prod_{c \in C(r)} \phi_c(v_c) $$
  $$ Z = \sum_v \prod_{c \in C(r)} \phi_c(v_c) $$

  \[ \log P(v) = \sum_c w_c \cdot f_c(v_c) - \log Z = w \cdot f(v) - \log Z \]
Bayesian Networks

SELECT doc1.Category, doc2.Category
FROM doc1, doc2, Link link
WHERE link.From = doc1.key and link.To = doc2.key

Markov Logic Networks

1.5 \( \forall x \, \text{Smokes}(x) \Rightarrow \text{Cancer}(x) \)
1.1 \( \forall x, y \, \text{Friends}(x, y) \Rightarrow (\text{Smokes}(x) \Leftrightarrow \text{Smokes}(y)) \)

Suppose we have two constants: Anna (A) and Bob (B)

Cancer(A)
Smokes(A)
Friends(A, A)
Friends(A, B)

Cancer(A)
Smokes(A)
Friends(A, A)
Friends(A, B)
Suppose we have two constants: **Anna** (A) and **Bob** (B)

Slides by Pedro Domingos

### Applications

- **Computer Vision**
  - (Taskar et al.)

- **Citation Analysis**
  - (Taskar et al., Singla&Domingos)

- **Activity Recognition**
  - (Liao et al.)

### Learning Undirected PRMs

- **Parameter estimation**
  - discriminative (gradient, max-margin)
  - generative setting using pseudo-likelihood

- **Structure learning**
  - Similar to PRMs, BLPS

Activity Recognition

[Fox et al. IJCAI03]

Will you go to the AdvancedAI lecture or will you visit some friends in a cafe?
3D Scan Data Segmentation
[Anguelov et al. CVPR05, Triebel et al. ICRA06]

How do you recognize the lecture hall?

Outline Relational Models

- Relational Models
  - Probabilistic Relational Models
  - Bayesian Logic Programs
  - Relational Markov networks
  - Markov Logic

Conclusions

- **SRL = Probability + Logic + Learning**
- Covers full AI spectrum: Logic, probability, learning, kernels, sequences, planning, reinforcement learning, ...
- Considered to be a revolution in ML
- Logical variables/Placeholders: group random variables/states
- Unification: context-specific prob. information

Thanks

... for your attention
... and enjoy the other parts of the lecture!