Bayesian Networks

Prof. Dr. Lars Schmidt-Thieme
Institute for Computer Science
Albert-Ludwigs-University of Freiburg

I. What is a bayesian network?

II. Overview

III. Organizational stuff
propositional logic:
• we can infer new statements from a given set of statements

• example:
  
  \[
  \begin{align*}
  \text{it rains} & \rightarrow \text{the lawn is wet} \\
  \text{the lawn is not wet} & \}
  \end{align*}
  \]  
  \[\Rightarrow \text{it does not rain}\]

• knowledge base is often represented as a set of rules $A \rightarrow B$.
• there exist efficient algorithms for inference (resolution mechanism, rule chaining, etc.; see, e.g., [CGH97, p. 28ff])
• variables are mapped to \{0, 1\} (truth values).

propositional logic with uncertainties:
• variables are mapped to \([0, 1]\) (certainties).

• rules are associated with a certainty $x \in [0, 1]$ (uncertain implications; generalized rules):

  \[A \rightarrow B \text{ with certainty } x.\]

Figure 1: Types of uncertain implications [CGH97, p. 86].
Rule-based inference with uncertainties (2/3)

- combination: how can certainties be combined?
  - I take a cup of coffee → I will stay awake during lecture with certainty 0.5
  - I take a walk → I will stay awake during lecture with certainty 0.8
  - I take a cup of coffee and I take a walk
    ⇒ I will stay awake during lecture with certainty . . . ?

- chaining: how can certainties be chained?
  - I take a cup of coffee → I will stay awake during lecture with certainty 0.5
  - I stay awake during lecture → I will get good marks in the exam with certainty 0.8
  - I take a cup of coffee
    ⇒ I will get good marks in the exam with certainty . . . ?

Rule-based inference with uncertainties (3/3)

- abduction: how can certainties be used in modus tollens?
  - I take a cup of coffee → I will stay awake during lecture with certainty 0.5
  - I do not stay awake during lecture.
    ⇒ I did not take a cup of coffee with certainty . . . ?

- Functions for combining and chaining certainties require ad-hoc adjustments and assumptions and thus are problematic (see [Nea90]).

- Systems using uncertain implications:
  - PROSPECTOR [DHN76]
  - MYCIN [BS84]
• The main task in data mining / statistical modelling is predictive modelling, i.e., the prediction of
  – the value of a continuous target variable (regression) or
  – the state of a categorical target variable (classification)
  based upon the knowledge of the values or states of other variables (predictor variables).

• In case of a categorical target variable, one often is more interested in predicting the probabilities of the different states instead of the most-probable state only (e.g., if cost info is available etc.).

Mathematically the prediction model is represented as a function from the domains of predictor variables in the domain of the target variable or $[0,1]$ respectively. If $X_i$ are the domains of the predictor variables ($i \in I$) and $Y$ is the domain of the target variable, then a model is described by

$$f : \prod_{i \in I} X_i \rightarrow Y \quad \text{[continuous target with domain $Y$]}$$

or

$$f : \prod_{i \in I} X_i \rightarrow [0,1]^Y \quad \text{[probabilities for the states $Y$ of a categorical target]}$$

• To make the task of learning such functional dependencies from data feasible, the class of admissible functions is restricted (linear models, neural nets, decision trees, etc.).

• Inference is done by evaluating the function $f$ for given values of the predictor variables.
Example: detection fraudulent transactions in banking.

\( X_1 = [0, 24] \): Time of day of transaction.
\( X_2 = [0, 10000] \): Amount of transaction (in EUR).
\( Y = \{\text{yes, no}\} \): transaction is fraudulent.

\[
f(x_1, x_2) := \frac{1}{1 + e^{10 + 10 \cdot \sin^2(x_1 \pi/24) - 0.0005 \cdot x_2}}
\]

\( f \) gives \( p(Y = \text{yes}) \).
As \( Y \) is binary, trivially

\[
p(Y = \text{no}) = 1 - p(Y = \text{yes})
\]

For example,

\[
f(10, 100) = 4.2 \cdot 10^{-9}
\]
\[
f(2, 10000) = 0.0034
\]

![Figure 2: Predicted probability \( y \) for given predictors \( x_1, x_2 \).](image)

We can infer probabilities for patients suffering from adeno from observed symptoms (\( V = \) vomiting, \( W = \) weightloss, \( P = \) pain):

- If the state of none of the symptoms is known, then

\[
p(\text{adeno} = \text{Y}) = \frac{700}{1000} = 0.7
\]
Inference using the Joint Probability Distribution

Figure 3: Number of patients classified by one disease (Adeno) and three symptoms [CGH97, p. 81].

We can infer probabilities for patients suffering from adeno from observed symptoms (V = vomiting, W = weightloss, P = pain):

- If $V = Y$, and we know there are no other symptoms ($W = N$, $P = N$), then
  
  \[ p(\text{adeno}=Y|V = Y, W = N, P = N) = \frac{10}{10 + 50} = 0.17 \]

- If $V = Y$, and we do not know the states of the other symptoms, then
  
  \[ p(\text{adeno}=Y|V = Y) = \sum_{w,q} p(\text{adeno}=Y, W = w, P = q|V = Y) \]
  
  \[ = \frac{220 + 25 + 95 + 10}{224 + 30 + 126 + 60} = \frac{350}{440} = 0.80 \]

In general, it is a bad idea to use the observed frequencies as estimation for the full joint probability distribution (JPD):

- If the number of variables increases, it becomes infeasible to store the JPD (e.g., for 100 binary variables the JPD has $2^{100} \approx 10^{30}$ cells).

- As the JPD has so much parameters, it suffers extremely from overfitting.

- Domain experts often can tell us in advance, that certain variables cannot be related to some other variables.

$\Rightarrow$ Idea: break the JPD down in several smaller conditional probability distributions of a subset of related variables (i.e., factor the JPD).
Bayesian Network:

**structure:** The graph structure encodes the factors of the JPD. Each node represents a variable $x$, nodes pointing at it represent variables $y_i (i \in \text{fanin}(x))$ from which $x$ depends.

![Figure 4: Example Bayesian Network.](image-url)

**parameters:** to each node is attached a conditional probability table

\[
p(x | (y_i)_{i \in \text{fanin}(x)})
\]

<table>
<thead>
<tr>
<th>Cold</th>
<th>Y</th>
<th>N</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angina</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Fever</td>
<td>Y</td>
<td>0.8</td>
<td>0.4</td>
</tr>
<tr>
<td>Sore Throat</td>
<td>Y</td>
<td>0.9</td>
<td>0.2</td>
</tr>
<tr>
<td>See Spots</td>
<td>Y</td>
<td>0.2</td>
<td>0.001</td>
</tr>
</tbody>
</table>

Figure 4: Example Bayesian Network.
I. What is a bayesian network?

II. Overview

Exact Inference
For inference, evidence $E$ – i.e., the knowledge of the states of some of the variables – is entered into the net and propagated, s.t. the marginals of the resulting node tables give $p(x|E)$.

To compute this efficiently, special graph structures as the join tree have to be derived.

Figure 5: a) Example bayesian network, b) triangulated moral graph, c) join tree [BK02, p. 127, 129].
Approximate Inference

For large or dense networks, exact inference is too expensive to compute. Approximative results can be established based on stochastic simulation.

\[
P(X_1 = 1) = .5 \\
P(X_2 = 1 \mid X_1 = 1) = .8 \\
P(X_2 = 1 \mid X_1 = 2) = .1 \\
P(X_3 = 1 \mid X_1 = 1) = .7 \\
P(X_3 = 1 \mid X_1 = 2) = .4 \\
P(X_4 = 1 \mid X_2 = 1) = .6 \\
P(X_4 = 1 \mid X_2 = 2) = .1 \\
P(X_5 = 1 \mid X_1 = 1, X_3 = 1) = .1 \\
P(X_5 = 1 \mid X_1 = 1, X_3 = 2) = .2 \\
P(X_5 = 1 \mid X_1 = 2, X_3 = 1) = .3 \\
P(X_5 = 1 \mid X_1 = 2, X_3 = 2) = .4 \\
\]

<table>
<thead>
<tr>
<th>Case</th>
<th>(X_1)</th>
<th>(X_2)</th>
<th>(X_3)</th>
<th>(X_4)</th>
<th>(X_5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>\</td>
<td>\</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>\</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>\</td>
<td>\</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 6: Example bayesian network [Nea03, p. 211].

Figure 7: Sample drawn from bayesian network in fig. 6.

\[
\hat{p}(X_1 = 1 \mid X_3 = 1, X_4 = 2) = \frac{3}{4}
\]

Bayesian Network Analysis

Beneath inference, some other tasks can be accomplished with bayesian networks, as, e.g., generating explanations for specific observations by computing the configuration having maximal probability.

\[
\text{We observe } A = a. \text{ Then we can compute the configuration } B = b, C = y, D = d, \ldots \text{ with }
\]

\[
p(A = a, B = b, C = c, D = d, \ldots)
\]

maximal among all \(p(A = a, \ldots)\). Thus, \(B = b, C = c, D = d, \ldots\) is the most likely explanation for observing \(A = a\).

Figure 8: Example bayesian network.
Manually Building Models

In many application domains, domain knowledge can be used to specify the structure and/or the parameters of Bayesian networks.

Figure 9: Genealogical structure for the horses in the studfarm example [Jen01, p. 47].

<table>
<thead>
<tr>
<th></th>
<th>aa</th>
<th>aA</th>
<th>AA</th>
</tr>
</thead>
<tbody>
<tr>
<td>aa</td>
<td>(1, 0, 0)</td>
<td>(0.5, 0.5, 0)</td>
<td>(0, 1, 0)</td>
</tr>
<tr>
<td>aA</td>
<td>(0.5, 0.5, 0)</td>
<td>(0.25, 0.5, 0.25)</td>
<td>(0, 0.5, 0.5)</td>
</tr>
<tr>
<td>AA</td>
<td>(0, 1, 0)</td>
<td>(0, 0.5, 0.5)</td>
<td>(0, 0, 1)</td>
</tr>
</tbody>
</table>

Figure 10: \( p(\text{Child} | \text{Father, Mother}) \) for genetic inheritance. The numbers are the probabilities for (aa, aA, AA) [Jen01, p. 47].

Learning Parameters

If we know which variables may influence with others (the structure), but not the exact quantities (the parameters), and we have data, we can estimate the parameters from data.

If data is rare, we need background knowledge to estimate parameters (Bayesian estimation).

Figure 11: Bayesian network structure with unknown parameters.
Learning Structure

Learning structure requires

- the specification of a model selection criterion as well as
- a search procedure over a subspace of possible graph structures.

Figure 12: Different bayesian network structures.

Bayesian Networks

I. What is a bayesian network?

II. Overview

III. Organizational stuff
Text books


Bayesian Networks Software

open source:

open source, based on commercial software:

commercial software:
- Hugin (http://www.hugin.com)

... and there are many others (see http://www.ai.mit.edu/~murphyk/Bayes/bnsoft.html).
Exercises and tutorials

- There will be a biweekly sheet with two exercises.
- First sheet is due at October 28.
- No corrections.
- No tutorials.

About Computer-based New Media
Junior professor at Institute for Computer Science of University of Freiburg since October 1, 2003.

Data Mining / Machine Learning:
- Web Mining
- Association rule algorithms
- Predictive modelling

Internet Technologies:
- XML and XML-databases
- web services and semantic web technologies
- Java and several Java-based server technologies

E-commerce- and e-business applications:
- recommender systems
- business intelligence (e.g., web usage mining)
- methods for marketing research (e.g., adaptive conjoint analysis)

E-learning:
- analysis of the usage of e-learning products and services
- automatic structuring of information
Seminar on Spam

Wednesday, 14-16, same room

Topics (t = with technical focus, m = with focus on methods):

1. (t) Spam at transport level
2. (t) Reporting and tracing spam
3. (t) Architectures of spam filters
4. (t) CRM114 - a programming language for filters
5. (t) Filtering spam in the haystack framework
6. (m) Rule and memory based filtering
7. (m) Bayesian filtering
8. (m) Classifying spam using support vector machines
9. (m) Classifying spam using association rules
10. (m) Improving spam classifiers by genetic algorithms
11. (m) Improving spam classifiers by stacking and boosting
12. (m) Text classification using background knowledge

References


