Learning Dynamic Bayesian Networks: Algorithms and Issues

Alex Black, Kevin Korb, Ann Nicholson

Clayton School of Information Technology, Monash University
Introduction

• Part 1: Background
  – Dynamic Bayesian Networks
  – DBN Learning

• Part 2: Our Research
  – CaMML DBN Learning
  – Experimental Evaluation
Dynamic Bayesian Networks

• Extension of BNs to time domain

• Why DBNs?
  – Temporal aspect to data or underlying process
  – Prediction over time
Dynamic Bayesian Networks

- Structure: same for all time steps
- Arcs span at most one time step
- Discrete data
2 Time Slice DBN

• Simplifies learning and representation
  – Learn $t=0$ and $t=T+1$ arcs/parameters

• For prediction
  – Unroll DBN
Learning Dynamic Bayesian Networks

Two learning approaches:

• Constraint Based
  – MIT, PC
  – Uses conditional independence tests

• Metric based - ‘Search and Score’
  – BIC, BDe, CaMML
  – Sample model space, score whole network
Learning DBNs

• BIC – Bayesian Information Criterion
• BDe
• MIT – Mutual Information Test
• CaMML – Causal Minimum Message Length

Static BN learners can be used

Implementations
• Research software
  – Variety of languages, input and output formats
• Issues: Maintenance? Documentation? GUI? Limitations?
Using Static BN Learners for DBNs

- Create new data set: Duplicate variables, offset by 1
  - Use tier prior constraints if available. (PC/Tetrad, CaMML)

<table>
<thead>
<tr>
<th>Time</th>
<th>X</th>
<th>Y</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>true</td>
<td>yes</td>
<td>high</td>
</tr>
<tr>
<td>1</td>
<td>false</td>
<td>yes</td>
<td>low</td>
</tr>
<tr>
<td>2</td>
<td>false</td>
<td>no</td>
<td>medium</td>
</tr>
<tr>
<td>3</td>
<td>false</td>
<td>no</td>
<td>low</td>
</tr>
<tr>
<td>4</td>
<td>true</td>
<td>yes</td>
<td>medium</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>N-2</td>
<td>false</td>
<td>yes</td>
<td>medium</td>
</tr>
<tr>
<td>N-1</td>
<td>true</td>
<td>yes</td>
<td>high</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Time</th>
<th>X0</th>
<th>Y0</th>
<th>Z0</th>
<th>X1</th>
<th>Y1</th>
<th>Z1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>true</td>
<td>yes</td>
<td>high</td>
<td>true</td>
<td>yes</td>
<td>high</td>
</tr>
<tr>
<td>1</td>
<td>false</td>
<td>yes</td>
<td>low</td>
<td>false</td>
<td>yes</td>
<td>low</td>
</tr>
<tr>
<td>2</td>
<td>false</td>
<td>no</td>
<td>medium</td>
<td>false</td>
<td>no</td>
<td>medium</td>
</tr>
<tr>
<td>3</td>
<td>false</td>
<td>no</td>
<td>low</td>
<td>false</td>
<td>no</td>
<td>low</td>
</tr>
<tr>
<td>4</td>
<td>true</td>
<td>yes</td>
<td>medium</td>
<td>true</td>
<td>yes</td>
<td>medium</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>N-2</td>
<td>false</td>
<td>yes</td>
<td>medium</td>
<td>false</td>
<td>yes</td>
<td>medium</td>
</tr>
</tbody>
</table>

- Why have separate DBN learners?
  - Computational speed
  - Improved results
PART 2: RESEARCH
Our Research: CaMML DBN Learning

CaMML – Causal Minimum Message Length

• Capable BN learner
• Extended for learning DBNs

Minimum Message Length:

• Information theoretic approach to statistical and inductive inference.
### DBN Learning Algorithms

Q: How well do they work in practice?

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Software</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CaMML – DBN Learner</td>
<td>CaMML</td>
<td></td>
</tr>
<tr>
<td>CaMML – BN Learner</td>
<td>CaMML</td>
<td>BN learner with tier prior</td>
</tr>
<tr>
<td>MIT</td>
<td>GlobalMIT</td>
<td>No intraslice arcs</td>
</tr>
<tr>
<td>BDe</td>
<td>Banjo</td>
<td>No parameters</td>
</tr>
<tr>
<td>BIC (2-Step)</td>
<td>Bayes Net Toolbox</td>
<td>Learn interslice + interslice arcs separately. Not ideal. No complete BIC implementation available.</td>
</tr>
<tr>
<td>PC</td>
<td>Tetrax</td>
<td>BN learner with tier prior</td>
</tr>
</tbody>
</table>
Experimental Design

Goal: Evaluate performance of DBN learning algorithms

Step 1: Start with existing DBN

Step 2: Generate Data

Step 3: Re-learn DBN using each algorithm

Our Experiments:
7 DBNs with 6 to 28 variables

Step 4: Compare
Evaluating BNs/DBNs

• Edit Distance
  – Score 1 for every arc error

• Kullback-Leibler (KL) Divergence

• Causal Kullback-Leibler (CKL) Divergence
KL Divergence & CKL

• Kullback-Leiber Divergence

\[ D_{KL}(P||Q) = \sum_i P(x_i) \log_e \left( \frac{P(x_i)}{Q(x_i)} \right) \]

– Difference between probability distributions
– But: Ignores network structure

• Causal Kullback-Leibler Divergence

– Both structure and probability
Results – Metastatic Cancer DBN

![Graph showing data for Metastatic Cancer DBN with different tiers and scores.]
Results – Threat DBN

Data Size

Edit Distance

CKL

- CaMML DBN
- CaMML Tier
- PC
- BIC (2-Step)
- BDe
DBNs: No Arcs Within Time Slice

• Some learners: no arcs within time slice
  – MIT, some BIC and BDe software
  – Common in bioinformatics

• Assumption holds: Good results, fast
• Assumption violated: Poor results
Learning DBNs: No Arcs Within Time Slices

Edit Distance - Water DBN

- CaMML DBN
- PC
- BIC
- MIT
- BDe

Data Size

Edit Distance

500 1000 2000 4000 8000 16000 32000

5000 6000 7000 8000 9000 10000 11000 12000 13000 14000 15000

C_NI_0 → C_NI_1
C烘干_0 → C烘干_1
C_水_0 → C_水_1
C_外_0 → C_外_1
C_非_0 → C_非_1
C_干_0 → C_干_1
C_和_0 → C_和_1
C_干_0 → C_干_1
No Intraslice Arcs: Invalid Assumption

True Network

MIT, BDe
data = 100 obs.

BIC, MIT, BDe
data = 1000 obs.
Summary of DBN Learning Results

• CaMML DBN learner
  – Better performance than other algorithms
  – Conservative: Errors mostly due to missing arcs

• BDe: Can overfit with small data sets.

• No arcs within time slice:
  – Some BIC/BDe implementations may be much faster
  – Poor results if assumption does not hold
Final Thoughts

• CaMML
  Supports BN and DBN learning
  [bayesian-intelligence.com/software/](https://bayesian-intelligence.com/software/)

• Paper on CaMML / DBN Learning – 2014

• Practical applications of CaMML DBN underway

• Learning DBNs? Time series data? Talk to us.
# DBN Structure Learning Software

<table>
<thead>
<tr>
<th>Name</th>
<th>Structure Learning</th>
<th>Parameter Learning</th>
<th>DBN Algorithms</th>
<th>GUI</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>CaMML</td>
<td>Yes</td>
<td>Yes</td>
<td>CaMML</td>
<td>Yes</td>
<td><a href="http://bayesian-intelligence.com">bayesian-intelligence.com</a></td>
</tr>
<tr>
<td>Bayes Net Toolbox*</td>
<td>Partial¹/2-Step¹</td>
<td>Yes</td>
<td>BIC¹,², BDe²</td>
<td>No</td>
<td><a href="https://code.google.com/p/bnt/">code.google.com/p/bnt/</a></td>
</tr>
<tr>
<td>BNFinder</td>
<td>Partial¹</td>
<td>?</td>
<td>BIC, BDe, MIT</td>
<td>No</td>
<td><a href="http://bioputer.mimuw.edu.pl/software/bnf/">bioputer.mimuw.edu.pl/software/bnf/</a></td>
</tr>
<tr>
<td>GlobalMIT*</td>
<td>Partial¹</td>
<td>No</td>
<td>MIT</td>
<td>No</td>
<td><a href="http://code.google.com/p/globalmit/">code.google.com/p/globalmit/</a></td>
</tr>
<tr>
<td>Tetrad</td>
<td>No/Manual³</td>
<td>Yes</td>
<td>(PC, others)</td>
<td>Yes</td>
<td><a href="http://www.phil.cmu.edu/tetrad/">www.phil.cmu.edu/tetrad/</a></td>
</tr>
<tr>
<td>GeNIe</td>
<td>No/Manual³,⁴</td>
<td>Yes</td>
<td>(PC, K2, other)</td>
<td>Yes</td>
<td><a href="http://genie.sis.pitt.edu">genie.sis.pitt.edu</a></td>
</tr>
<tr>
<td>Banjo</td>
<td>Yes</td>
<td>No</td>
<td>BDe</td>
<td>No</td>
<td><a href="http://cs.duke.edu/~amink/software/banjo/">cs.duke.edu/~amink/software/banjo/</a></td>
</tr>
</tbody>
</table>

Kevin Murphy’s Bayesian Network Software List:  

* Requires Matlab. GlobalMIT: May be possible to use Octave (free) instead of Matlab
1 Supports DBN learning with interslice arcs only (i.e. no arcs within time slices)
2 With DBmcmc extension (bioss.ac.uk/~dirk/software/DBmcmc/) but binary/ternary data/attributes only
3 No official support for learning DBNs. Can adapt BN algorithms using tier priors etc.
4 DBN parameter learning, but no structure learning. Supports DBN inference, unrolling etc.
References


