A Framework for the Scoring of Operators on the Search Space of Equivalence Classes of Bayesian Network Structures

Rónán Daly (University of Edinburgh)
Qiang Shen (University of Wales, Aberystwyth)

5th September 2005
Introduction

• Bayesian Networks

• Learning Bayesian Networks—Problems

• Equivalence Classes of Bayesian Networks

• Learning Equivalence Classes—Problems

• Proposed solution

• Preliminary Results

• Conclusions
Bayesian Networks
Bayesian Networks

- Probabilistic model that represents a joint probability distribution

- Consists of a pair \((G, P)\), with \(G\) a directed acyclic graph (DAG) and \(P\) a joint probability distribution on a set of variables \(V\)

- \(G = (V, E)\), i.e. the graph is made up of nodes that represent the variables \(V\) with directed edges \(E\) connecting them

- \((G, P)\) must satisfy the Markov condition—\(\forall X \in V : IP(\{X\}, ND_X | PA_X)\)

- If this is the case then the joint probability distribution can be factored into the product of the conditional distributions of each variable given its parents

- \(P(x_1, x_2, \ldots, x_n) = P(x_1 | PA_{x_1}) P(x_2 | PA_{x_2}) \cdots P(x_n | PA_{x_n})\)
\[ P(Cloudy, Sprinkler, Rain, Wet Grass) = P(Cloudy)P(Sprinkler|Cloudy)P(Rain|Cloudy)P(Wet Grass|Sprinkler, Rain) \]
Explanations

• All probabilities can be calculated from joint.

\[ P(x_1|x_2) = \frac{P(x_1,x_2)}{P(x_2)} = \frac{\sum_{X_3,\ldots,X_n} P(x_1,x_2,X_3,\ldots,X_n)}{\sum_{X_1,X_3,\ldots,X_n} P(X_1,x_2,X_3,\ldots,X_n)} \]

• So why use a network?

• More compact representation—joint distribution exponential in number of variables
  
  – Requires a lot of storage space and computation time.
  – Also practically impossible to obtain all the probabilities.

• Brings an element of causality into the structure.

  – Create a causal DAG \( G = (V, E) \) and assume the distribution of \( V \) satisfies the Markov condition. Then given an edge \( X \rightarrow Y \), we can say \( X \) causes \( Y \).
Learning Structure
Learning Structure

- Two general methods used in learning structure
  - Bayesian scoring methods, which score a candidate DAG against data
  - Constraint-based methods, which use conditional independencies in data to identify $d$-separations and from these construct a structure

- First method uses scoring function, e.g. the Bayesian scoring criterion
  - $score_B(d, G) = P(d|G)$

- Graph with highest score wins

- Most algorithms use state based search—modify current state until stop criteria met
  - E.g. Greedy Searches, Simulated Annealing etc.
Problems

- Space of DAGs is huge
  - With 10 variables there are roughly $10^{18}$ possible DAGs

- Redundancy of DAGs
  - Certain DAGs can be equivalent to each other in that they capture the same conditional independencies

- Increases search space

- Causes connectivity problems between states which represent different conditional independencies
Equivalence Classes of Bayesian Networks
Equivalence Classes

• Can capture equivalent DAGs into an equivalence class

• Two DAGs are equivalent if they have the same skeleton and same set of v-structures

• Can represent an equivalence class using a partially directed acyclic graph (PDAG)

A graph consisting of directed and undirected edges and no directed cycles

• Can extend notion to completed partially directed acyclic graph (CPDAG)
  – Uniquely describes an equivalence class using a “minimal” representation
Searching Through Equivalence Classes

- Similar to searching through a space of Bayesian networks

- Given a certain state, can apply a move and examine the resulting state for validity and "goodness" of the move (score difference)

- Efficiency problems—Small changes can "cascade"—Results in many score operations needed

- For certain move operators can get validity and score difference without having to generate and examine the resulting state

- However, this means the operators have to be defined and analysed beforehand

- Need way to check if a particular resulting state is valid and to find the score for that state efficiently
Proposed Solution
Scoring in a Generic Fashion

• Based on algorithm of Dor and Tarsi
  – Used to find a consistent extension of a PDAG, if it exists

• A consistent extension of a PDAG $\mathcal{P}$ is a DAG $\mathcal{G}$, such that $\mathcal{G}$ has the same skeleton and set of v-structures as $\mathcal{P}$

• Direct arcs towards same nodes in before and after states

• Try to make sure that as many nodes have the same parent set in both states

• This will minimise the number of scoring operations needed
  – Scoring functions are, in general, decomposable
Scoring in a Generic Fashion

• General idea of algorithm

**Input:** PDAG $\mathcal{P}_B$, PDAG $\mathcal{P}_A$

**Output:** PDAG $\mathcal{P}_B$, PDAG $\mathcal{P}_A$

$A := $ Nodes in $\mathcal{P}_B$

while $A$ has nodes {

Select a node $a$, such that $a \in A$ and the undirected arcs in both $\mathcal{P}_B$ and $\mathcal{P}_A$

  can be directed towards $a$ without creating any new v-structures

Direct all undirected arcs towards $a$ in both $\mathcal{P}_B$ and $\mathcal{P}_A$

$A := A\setminus a$

}

• Algorithm in paper more complicated to minimize iterations and deal with special cases
Preliminary Results

- Results show a small comparison study

- Compares the running time of the proposed framework and predefined operators in two learning schemes

<table>
<thead>
<tr>
<th>Samples</th>
<th>Chickering 02</th>
<th>GES</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>263</td>
<td>327</td>
</tr>
<tr>
<td>1000</td>
<td>317</td>
<td>374</td>
</tr>
<tr>
<td>1500</td>
<td>470</td>
<td>535</td>
</tr>
<tr>
<td>2000</td>
<td>489</td>
<td>534</td>
</tr>
</tbody>
</table>

Table 1: Running times of algorithms in seconds
Conclusions and Future Directions

Conclusions

• Framework doesn’t seem to introduce too much overhead

• Will help in the definition of other learning schemes
  – Won’t need to define and analyse theory of operator moves

• Could lead to more rapid development of algorithms

Future Directions

• Find ways of checking both the validity and score of arbitrary moves
Questions