Bayesian network learning by compiling to weighted MAX-SAT

James Cussens, University of York

Mathematical Aspects of Graphical Models, Durham, 2008-07-07
Outline

Introduction

Weighted MAX-SAT

Encoding BN model selection as weighted CNF

Pre-computing scores

Experiments

Results

Recent work
Model selection as combinatorial optimisation

- Model selection for Bayesian networks (using a decomposable score) is **combinatorial optimisation**.
- In this work the score is marginal likelihood with a Dirichlet parameter prior.
- \[ P(D|G) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(n_{ij}+\alpha_{ij})}{\Gamma(\alpha_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(n_{ijk}+\alpha_{ijk})}{\Gamma(\alpha_{ijk})} \]
- \[ \text{Score}(G) \overset{\text{def}}{=} \log P(D|G) = \sum_{i=1}^{n} \text{Score}_i(\text{Pa}_i(G)) \]
- For each variable choose high-scoring parents subject to the constraint that no cycle is formed.
The basic idea

- Given that BN model selection is combinatorial optimisation
The basic idea

- Given that BN model selection is combinatorial optimisation...
- ...we can use state-of-the-art algorithms for combinatorial optimisation...
The basic idea

- Given that BN model selection is combinatorial optimisation...
- ...we can use state-of-the-art algorithms for combinatorial optimisation...
- ...if we are prepared to do a little encoding.
Outline

Introduction

**Weighted MAX-SAT**

Encoding BN model selection as weighted CNF

Pre-computing scores

Experiments

Results

Recent work
The SAT problem

- Is a given set of propositional clauses (a CNF formula) satisfiable?

\[
\begin{align*}
\overline{x_{12}} \lor \overline{x_{23}} \lor x_{13} \\
x_{12} \lor x_{23} \lor \overline{x_{13}} & \quad \text{OK} : (x_{12}, x_{23}, x_{13}), (x_{12}, \overline{x_{23}}, x_{13}), \ldots
\end{align*}
\]
The SAT problem

- Is a given set of propositional clauses (a CNF formula) satisfiable?

\[
\overline{x_{12}} \lor \overline{x_{23}} \lor x_{13} \\
x_{12} \lor x_{23} \lor \overline{x_{13}} \quad \text{OK} : (x_{12}, x_{23}, x_{13}), (x_{12}, \overline{x_{23}}, x_{13}), \ldots \\
x_{12} \quad \text{OK} : (x_{12}, x_{23}, x_{13}) \\
x_{23}
\]
The SAT problem

- Is a given set of propositional clauses (a CNF formula) satisfiable?

\[ \overline{x_{12}} \lor \overline{x_{23}} \lor x_{13} \]
\[ x_{12} \lor x_{23} \lor \overline{x_{13}} \]
\[ x_{12} \]
\[ x_{23} \]
\[ \overline{x_{13}} \]

OK : \((x_{12}, x_{23}, x_{13}), (x_{12}, \overline{x_{23}}, x_{13}), \ldots\)

OK : \((x_{12}, x_{23}, x_{13})\)

Unsatisfiable

- \(x_{12}, x_{23}\) and \(x_{13}\) are called *atoms*. (Short for atomic formulae.)
The weighted MAX-SAT problem

▶ Add weights to each clause (to get weighted CNF).
▶ Each assignment has a cost: the sum of the weights of the unsatisfied clauses.
▶ An infinite cost gives a ‘hard’ clause. (In practice a big number is used.)
▶ Goal: find an assignment with minimal cost.

\[
\begin{align*}
9999 & \quad \overline{x_{12}} \lor \overline{x_{23}} \lor x_{13} \\
9999 & \quad x_{12} \lor x_{23} \lor \overline{x_{13}} \\
12 & \quad x_{12} \\
34 & \quad x_{23} \\
1 & \quad \overline{x_{13}}
\end{align*}
\]
Weighted MAX-SAT as mode finding for log-linear distributions

- Given weighted CNF $\lambda_1 C_1, \lambda_2 C_2, \ldots$
- Define $f_i(x) = 1$ if $x$ breaks clause $C_i$; else $= 0$
- $P(x) = Z^{-1} \exp \left( \sum_i -\lambda_i f_i(x) \right)$

This connection has been exploited by those working on Markov logic where weighted first-order clauses are used.
Here are the SAT solving algorithms available in UBCSAT.

19 have weighted MAX-SAT variants

- Adaptive G2WSAT
- Adaptive G2WSAT+p
- Adaptive Novelty+
- Conflict-Directed Random Walk
- DDFW: Divide and Distribute Fixed Weights
- Deterministic Conflict-Directed Random Walk
- Deterministic Adaptive Novelty+
- G2WSAT: Gradient-based Greedy WalkSAT
- G2WSAT+p: Gradient-based Greedy WalkSAT with look-ahead
- GSAT: Greedy Search for SAT
- GSAT/TABU: GSAT with Tabu search
- GWSAT: GSAT with Random Walk
- HSAT: GSAT with History Information
Weighted MAX-SAT solvers

- HWSAT: HSAT with Random Walk
- IRoTS: Iterated Robust TABU Search
- Novelty
- Novelty+: Novelty with Random Walk
- Novelty++: Novelty with Diversification Probability
- Novelty+p: Novelty+ with look-ahead
- PAWS: Pure Additive Weighting Scheme
- RoTS: Robust Tabu Search
- R-Novelty
- R-Novelty+: R-Novelty with Random Walk
- RGSAT: Restarting GSAT
- RSAPS: Reactive SAPS
- SAMD: Steepest Ascent Mildest Descent
- SAPS: Scaling and Probabilistic Smoothing
- SAPS/NR: De-randomized version of SAPS
- Uniform Random Walk
- VW1: Variable Weighting Scheme One
- VW2: Variable Weighting Scheme Two
- WalkSAT
- WalkSAT/TABU: WalkSAT with TABU search
Outline

Introduction

Weighted MAX-SAT

Encoding BN model selection as weighted CNF

Pre-computing scores

Experiments

Results

Recent work
Choosing parents incurs a cost, but we must choose

- Create atoms: “$X_i$ has parent set $P_a$”
- Create weighted clauses: $-\text{Score}_i(P_a) : X_i \text{ has parent set } P_a$
- Create ‘hard’ clauses:
  $(X_i \text{ has parent set } P_{ai_1}) \lor (X_i \text{ has parent set } P_{ai_2}) \lor \cdots \lor (X_i \text{ has parent set } P_{ai_{mi}})$
- Choosing parents for each variable determines the DAG.
Ruling out cycles with a total order

- Encode variable orderings as well as DAGs (à la Friedman and Koller)
- Create $n(n - 1)/2$ atoms: $\text{ord}(X_i, X_j)$ meaning $X_i$ and $X_j$ are lexicographically ordered in the variable ordering.
- Create hard clauses:
  
  $X_j$ has parent set $\{X_i, X_k\} \rightarrow \text{ord}(X_i, X_j)$
  
  $X_j$ has parent set $\{X_i, X_k\} \rightarrow \text{ord}(X_j, X_k)$

- Create $n(n - 1)(n - 2)/3$ hard clauses:
  
  $\text{ord}(X_i, X_j) \lor \text{ord}(X_j, X_k) \lor \text{ord}(X_i, X_k)$
  
  $\text{ord}(X_i, X_j) \lor \text{ord}(X_j, X_k) \lor \text{ord}(X_i, X_k)$
Outline

Introduction

Weighted MAX-SAT

Encoding BN model selection as weighted CNF

Pre-computing scores

Experiments

Results

Recent work

James Cussens, University of York  
BN learning via weighted MAX-SAT
Pre-computing scores

- All weighted MAX-SAT solvers (that I know of) require all weights to be known before solving begins.
- So compute and store $Score_i(Pa)$ for every variable $i$ and candidate parent set $Pa$.
- I used a limit of 3 parents.
- With their more efficient code (and 4 dual-core machines) Silander and Myllymäki’s bene system took 6 hours 16 minutes to compute all parent scores when there were 29 variables.
- In an example with 17 variables bene took under 18 seconds.
Filtering ‘family’ scores

If

\[
\begin{align*}
\text{Score}_4 \left(\begin{array}{c}
X_1 \\
\downarrow \quad \downarrow \\
X_3 \\
X_4
\end{array}\right) & > \text{Score}_4 \\
\left(\begin{array}{c}
X_1 \\
X_2 \\
\downarrow \\
X_4 \\
X_3
\end{array}\right)
\end{align*}
\]

then throw RHS score away.
Introduction
Weighted MAX-SAT
Encoding BN model selection as weighted CNF
Pre-computing scores

Experiments
Results
Recent work

BN learning via weighted MAX-SAT
Datasets of size 100, 1000 and 10000 were produced by forward sampling from the following 7 BNs.

| Name    | n  | \(|Pa| | r  |
|---------|----|------|----|
| Mildew  | 35 | 3    | 100|
| Water   | 32 | 5    | 4  |
| alarm   | 37 | 4    | 4  |
| asia    | 8  | 2    | 2  |
| carpo   | 60 | 5    | 4  |
| hailfinder | 56 | 4    | 11 |
| insurance | 27 | 3    | 5  |
Size of WCNF

- These are sizes for an alternative encoding using a partial order over variables.

<table>
<thead>
<tr>
<th>Data</th>
<th>atoms</th>
<th>clauses</th>
<th>lits</th>
</tr>
</thead>
<tbody>
<tr>
<td>ca_2</td>
<td>8,609</td>
<td>226,406</td>
<td>661,551</td>
</tr>
<tr>
<td>ca_3</td>
<td>7,368</td>
<td>221,365</td>
<td>651,469</td>
</tr>
<tr>
<td>ca_4</td>
<td>19,932</td>
<td>269,367</td>
<td>747,473</td>
</tr>
<tr>
<td>ha_2</td>
<td>3,325</td>
<td>170,009</td>
<td>509,305</td>
</tr>
<tr>
<td>ha_3</td>
<td>3,842</td>
<td>171,400</td>
<td>512,087</td>
</tr>
<tr>
<td>ha_4</td>
<td>6,849</td>
<td>181,545</td>
<td>532,377</td>
</tr>
<tr>
<td>in_2</td>
<td>982</td>
<td>18,926</td>
<td>56,049</td>
</tr>
<tr>
<td>in_3</td>
<td>1,477</td>
<td>20,346</td>
<td>58,889</td>
</tr>
<tr>
<td>in_4</td>
<td>4,355</td>
<td>30,344</td>
<td>78,885</td>
</tr>
</tbody>
</table>
The MaxWalkSAT algorithm

while still_trying:
    somehow_assign_truth_values_to_all_atoms
while cost <= target:
    c = random_choice(unsat_clauses)
    lits = lits_of(c)
    if random_flip:
        lit = random_choice(lits)
    else:
        lit = lowest_cost_flip(lits)
    flip_truth_value(lit)
    update_cost
Running MaxWalkSAT

newmaxwalksat version 20 (Huge)
seed = 99955222
cutoff = 10000000
tries = 100
numsol = 1
targetcost = 503040
heuristic = best, noise 50 / 100, init initfile
allocating memory...
clauses contain explicit costs
numatom = 6848, numclause = 181544, numliterals = 529296
wff read in

<table>
<thead>
<tr>
<th>lowest cost</th>
<th>worst clause</th>
<th>number #unsat</th>
<th>#flips model</th>
<th>success rate</th>
<th>tries until assign</th>
</tr>
</thead>
<tbody>
<tr>
<td>506076</td>
<td>16968</td>
<td>56 10000000</td>
<td>*</td>
<td>0</td>
<td>*</td>
</tr>
<tr>
<td>501973</td>
<td>23318</td>
<td>56 2913803</td>
<td>2913803</td>
<td>50</td>
<td>12913803</td>
</tr>
</tbody>
</table>

total elapsed seconds = 75.428415
average flips per second = 171206
number of solutions found = 1
mean flips until assign = 12913803.000000
mean seconds until assign = 75.428415
mean restarts until assign = 2.000000
ASSIGNMENT ACHIEVING TARGET 503040 FOUND
Nature of the search space

- If the current assignment of truth values to the atoms breaks at least one hard clause, then this assignment does not correspond to a DAG.
- The search (temporarily) visits cyclic graphs and ‘graphs’ were a variable’s parent set may be undefined.
- Breaking hard constraints is OK; they will be fixed eventually.
Outline

Introduction

Weighted MAX-SAT

Encoding BN model selection as weighted CNF

Pre-computing scores

Experiments

Results

Recent work
Searching for high scoring BNs

<table>
<thead>
<tr>
<th>Data</th>
<th>True</th>
<th>Ancestor</th>
<th>Total order</th>
<th>Long</th>
<th>&gt; True</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mi_2</td>
<td>-7,786</td>
<td>-5,711</td>
<td>-5,708</td>
<td>-5,705</td>
<td>Y</td>
</tr>
<tr>
<td>Mi_3</td>
<td>-63,837</td>
<td>-47,229</td>
<td>-47,194</td>
<td>-47,120</td>
<td>Y</td>
</tr>
<tr>
<td>Mi_4</td>
<td>-470,215</td>
<td>-409,641</td>
<td>-410,159</td>
<td>-408,282</td>
<td>Y</td>
</tr>
<tr>
<td>Wa_2</td>
<td>-1,801</td>
<td>-1,488</td>
<td>-1,486</td>
<td>-1,484</td>
<td>Y</td>
</tr>
<tr>
<td>Wa_3</td>
<td>-13,843</td>
<td>-13,293</td>
<td>-13,284</td>
<td>-13,247</td>
<td>Y</td>
</tr>
<tr>
<td>Wa_4</td>
<td>-129,655</td>
<td>-129,274</td>
<td>-128,916</td>
<td>-128,812</td>
<td>Y</td>
</tr>
<tr>
<td>al_2</td>
<td>-1,410</td>
<td>-1,368</td>
<td>-1,368</td>
<td>-1,336</td>
<td>Y</td>
</tr>
<tr>
<td>al_3</td>
<td>-11,305</td>
<td>-11,599</td>
<td>-11,501</td>
<td>-11,339</td>
<td>N</td>
</tr>
<tr>
<td>al_4</td>
<td>-105,303</td>
<td>-107,205</td>
<td>-106,503</td>
<td>-105,907</td>
<td>N</td>
</tr>
</tbody>
</table>
### Searching for high scoring BNs

<table>
<thead>
<tr>
<th>Data</th>
<th>True</th>
<th>Ancestor</th>
<th>Total order</th>
<th>Long</th>
<th>&gt; True</th>
</tr>
</thead>
<tbody>
<tr>
<td>as_2</td>
<td>-247</td>
<td>-241</td>
<td>-241</td>
<td>-241</td>
<td>Y</td>
</tr>
<tr>
<td>as_3</td>
<td>-2,318</td>
<td>-2,312</td>
<td>-2,312</td>
<td>-2,312</td>
<td>Y</td>
</tr>
<tr>
<td>as_4</td>
<td>-22,466</td>
<td>-22,462</td>
<td>-22,462</td>
<td>-22,462</td>
<td>Y</td>
</tr>
<tr>
<td>ca_2</td>
<td>-1,969</td>
<td>-1,849</td>
<td>-1,852</td>
<td>-1,824</td>
<td>Y</td>
</tr>
<tr>
<td>ca_3</td>
<td>-17,739</td>
<td>-17,938</td>
<td>-17,891</td>
<td>-17,731</td>
<td>Y</td>
</tr>
<tr>
<td>ca_4</td>
<td>-173,682</td>
<td>-175,832</td>
<td>-176,456</td>
<td>-174,605</td>
<td>N</td>
</tr>
</tbody>
</table>
Outline

Introduction

Weighted MAX-SAT

Encoding BN model selection as weighted CNF

Pre-computing scores

Experiments

Results

Recent work
Working directly on total orders

- Given a total ordering the best parents for each variable are easy to find.

Parent sets for Cancer

<table>
<thead>
<tr>
<th>Parent set</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>{TB, TU}</td>
<td>-2.24772935188</td>
</tr>
<tr>
<td>{TB}</td>
<td>-3.00976537207</td>
</tr>
<tr>
<td>{SM, XR}</td>
<td>-8.07036732971</td>
</tr>
<tr>
<td>{TU, XR}</td>
<td>-9.37534407212</td>
</tr>
<tr>
<td>{XR}</td>
<td>-9.38063760741</td>
</tr>
<tr>
<td>{SM, TU}</td>
<td>-21.6756460345</td>
</tr>
<tr>
<td>{SM}</td>
<td>-21.6903150436</td>
</tr>
<tr>
<td>{}</td>
<td>-25.2333385745</td>
</tr>
</tbody>
</table>
Introduction
Weighted MAX-SAT
Encoding BN model selection as weighted CNF
Pre-computing scores
Experiments
Results
Recent work

Decision tree for choosing parents

\begin{figure}
\centering
\begin{tikzpicture}
\node (root) {Tb < Ca?}
    child {node (t1) {XR < Ca?}
        child {node (t2) {Sm < Ca?}
            child {node (t3) {Tu < Ca?}
                child {node (t4) {Sm < Ca?}
                    child {node (t5) {Tu < Ca?}
                        child {node (t6) {Sm < Ca?}
                            child {node (t7) {Tu < Ca?}
                                child {node (t8) {Sm < Ca?}
                                    child {node (t9) {Tu < Ca?}
                                        child {node (t10) {Sm < Ca?}
                                            child {node (t11) {Tu < Ca?}}}
                                        child {node (t12) {Sm < Ca?}
                                            child {node (t13) {Tu < Ca?}}}}}}}}}
                        child {node (t14) {Tu < Ca?}}}}}}
            child {node (t15) {Sm < Ca? \{Tb\}}}
        child {node (t16) {Tu < Ca?}}}
    child {node (t17) {Tu < Ca? \{Tb, Tu\}}}
\end{tikzpicture}
\end{figure}
Encoding as WCNF

2 : \( (Tb < Ca) \lor (Tu < Ca) \)
3 : \( (Tb < Ca) \lor (Tu < Ca) \)
8 : \( (Tb < Ca) \lor (XR < Ca) \lor (Sm < Ca) \)
9 : \( (Tb < Ca) \lor (XR < Ca) \lor (Sm < Ca) \lor (Tu < Ca) \)
9 : \( (Tb < Ca) \lor (XR < Ca) \lor (Sm < Ca) \lor (Tu < Ca) \)
21 : \( (Tb < Ca) \lor (XR < Ca) \lor (Sm < Ca) \lor (Tu < Ca) \)
21 : \( (Tb < Ca) \lor (XR < Ca) \lor (Sm < Ca) \lor (Tu < Ca) \)
25 : \( (Tb < Ca) \lor (XR < Ca) \lor (Sm < Ca) \)

James Cussens, University of York

BN learning via weighted MAX-SAT
Using the irots solver and the new encoding get:

- Score of -132,951 for insurance 10,000 dataset. Beats best previous score of -133,934 and score of true BN which is -133,489.
- Score of -497,652 for hailfinder 10,000 dataset. Beats best previous score of -498,739.