Bayesian Networks Learning: A Survey

JIAQI YANG
Department of Computer Science, Graduate Center, CUNY
jyang2@gradcenter.cuny.edu

Bayesian Networks

alarm1000.csv

```
0,2,0,1,2,2,1,3,3,1,3,1,1,1,0,0,0,2,0,0,0,0,0,0,2,2,0,0,0,1,0,0,0,3,0,0,2
0,2,0,1,1,1,0,3,1,1,0,1,0,0,0,1,0,1,0,0,0,0,0,0,2,2,2,0,1,1,0,0,0,0,0,0,2
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,2,0,0,0,0,0,0,2,2,2,0,2,1,0,0,0,3,0,0,2
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,2,0,0,0,0,0,0,2,2,2,0,0,1,1,1,2,3,0,0,2
0,2,0,1,2,2,0,2,0,1,0,1,0,0,0,1,0,0,1,2,0,1,0,0,1,1,2,2,2,0,0,0,1,1,0,1,1
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,2,0,0,0,0,0,0,2,2,0,0,2,1,0,1,0,3,0,0,2
0,2,0,0,2,2,0,1,3,1,1,1,1,0,0,0,0,1,0,0,0,0,0,0,2,2,2,0,1,1,0,0,0,1,1,0,2
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,1,0,0,0,0,0,0,2,2,2,0,1,1,0,2,0,3,0,0,2
0,2,0,1,1,1,0,3,1,1,0,1,0,0,0,1,0,1,0,0,0,0,0,0,2,1,2,0,1,1,0,0,0,0,0,0,2
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,2,0,0,0,0,0,0,2,2,1,1,1,1,0,2,0,0,0,0,2
1,2,0,1,2,2,0,1,3,0,3,1,1,1,0,0,0,1,0,0,0,0,0,0,2,2,2,0,1,1,0,2,0,3,0,0,2
0,2,0,1,0,1,0,3,1,1,0,1,0,0,0,1,0,1,0,0,0,0,0,0,2,2,2,0,1,1,0,0,0,0,0,0,2
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,2,0,0,0,0,0,0,2,2,2,0,0,1,0,1,0,3,0,0,2
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,2,0,2,0,1,1,0,0,1,2,2,2,1,0,0,1,3,0,1,0
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,0,0,0,0,0,0,0,2,2,2,0,0,1,0,1,0,3,0,0,2
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,1,0,0,0,0,0,1,1,1,1,1,1,1,0,0,0,3,1,0,1
0,2,0,1,2,2,0,1,3,1,3,0,1,1,0,0,0,1,0,0,0,0,0,0,2,2,2,0,1,1,0,2,0,3,0,0,2
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,0,0,0,0,0,0,0,2,2,2,0,0,1,0,1,0,3,0,0,2
0,2,0,1,2,2,1,3,1,1,0,1,0,0,0,1,0,0,0,0,1,2,0,1,1,1,1,1,1,1,0,2,0,0,1,2,1
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,0,0,0,0,0,0,0,2,2,2,0,0,1,0,2,0,3,0,0,2
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,2,0,0,0,0,1,0,0,0,1,1,1,1,0,0,0,3,0,0,0
0,2,0,1,2,2,1,3,3,1,3,0,1,1,0,0,0,2,0,0,0,0,0,0,2,2,2,0,0,1,0,2,0,3,0,0,2
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,1,0,0,0,0,1,0,0,2,2,0,1,1,0,0,0,1,0,0,0
0,2,0,1,2,1,0,3,1,1,0,1,0,0,1,1,0,1,0,0,1,2,0,0,2,2,2,0,1,1,0,0,0,0,0,2,2
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,2,0,0,0,2,1,0,0,0,1,1,1,1,1,0,2,3,0,0,0
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,2,0,0,0,0,1,0,0,0,1,2,2,0,0,1,0,3,0,0,0
0,0,1,1,2,2,0,1,3,1,3,0,1,1,0,0,0,0,0,0,0,0,0,0,2,2,2,0,0,1,0,1,0,3,0,0,2
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,2,0,0,0,0,0,0,2,2,2,0,0,1,1,0,2,3,0,0,2
0,2,0,1,2,2,0,1,3,1,3,1,1,0,0,0,0,0,1,2,0,1,1,0,0,0,2,2,0,1,0,2,1,3,0,1,0
0,0,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,1,0,3,1,0,1
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,0,0,0,0,2,0,0,1,1,2,0,0,1,0,2,0,3,0,0,1
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,2,0,0,1,2,0,0,1,1,2,0,0,1,0,2,0,3,0,2,1
0,2,0,1,2,2,0,1,3,1,3,1,1,1,0,0,0,2,0,0,0,0,1,0,0,0,1,1,1,1,0,0,0,3,0,0,0
```

Line 1, Column 1

- The raw dataset for learning. (37 * 1000)

BN	p	n	palim	|PaSets|	1.4.1	1.6	1.6.1
Diabetes	413	100	2	4441	1135	191	170
Diabetes	413	1000	2	21493	2754	550	437
Diabetes	413	10000	2	262129	5497	2007	1479
Link	724	100	2	7431402	411	496	358
Link	724	1000	2	2379361	1150	892	1424
Link	724	10000	2	2421283	9005	6373	4502
Mildew	35	100	3	3520	17	8	10
Mildew	35	1000	3	161	64	25	12
Mildew	35	10000	3	463	167	79	19
Pigs	441	100	2	2692	151	99	63
Pigs	441	1000	2	15847	203	189	213
Pigs	441	10000	2	304219	2082	1270	832
Water	32	100	3	482	1	1	1
Water	32	1000	3	573	1	1	1
Water	32	10000	3	961	7	2	1
alarm	37	100	3	907	1	1	1
alarm	37	1000	3	1928	3	2	1
alarm	37	10000	3	6473	16	6	2
asia	8	100	∞	41	1	1	1
asia	8	1000	∞	112	1	1	1
asia	8	10000	∞	169	1	1	1
carpo	60	100	3	5068	5	8	2
carpo	60	1000	3	3827	14	12	7
carpo	60	10000	3	16391	99	24	6
hailfinder	56	100	3	244	20	17	3
hailfinder	56	1000	3	761	39	49	10
hailfinder	56	10000	3	3768	124	214	26
insurance	27	100	3	279	1	1	1
insurance	27	1000	3	774	1	1	1
insurance	27	10000	3	3652	5	2	1
kredit	18	1000	3	70	1	1	1
kredit	18	1000	∞	70	248	2370	207

- Different databases.
- Some are synthetic data

- https://www.cs.york.ac.uk/aig/sw/gobnilp/

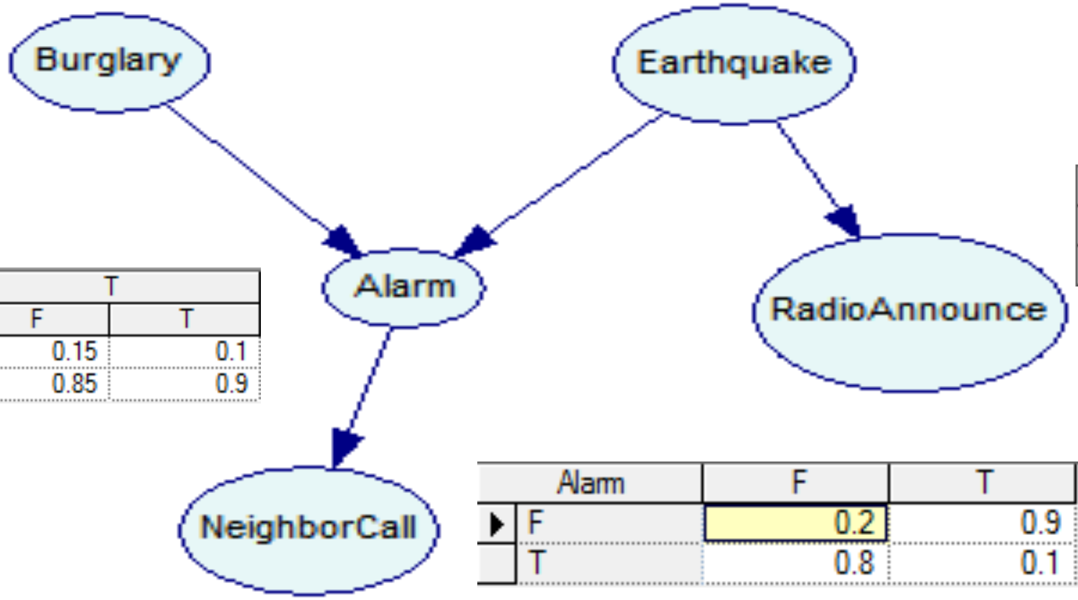
Bayesian Networks

$P(B)$

F	0.8
T	0.2

$P(E)$

F	0.9
T	0.1



$P(R|E)$

Earthquake	F	T
F	0.95	0.99
T	0.05	0.01

$P(A|B,E)$

Burglary	F		T	
Earthquake	F	T	F	T
F	0.8	0.2	0.15	0.1
T	0.2	0.8	0.85	0.9

$P(N|A)$

Alarm	F	T
F	0.2	0.9
T	0.8	0.1

- A Bayesian Network is a **directed acyclic graph (DAG)** learned from given dataset.
- A Bayesian network joint probability distribution:

$$P(B,E,A,R,N) = P(N \mid A)P(R \mid E)P(A \mid B, E)P(E)P(B)$$

Local Scores

- Formulate the learning task as minimize local scores problem.
- Score-based structure learning(Minimum Description Length):

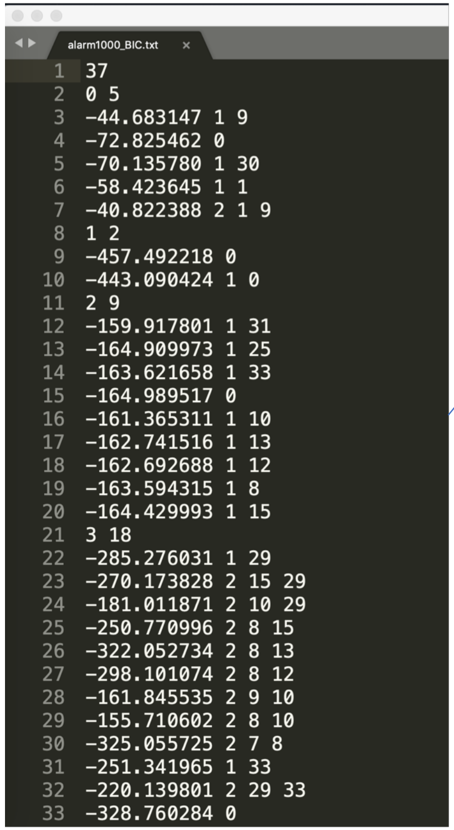
$$
\begin{aligned}
MDL(X_i|PA_i) &= H(X_i|PA_i) + \frac{\log N}{2} K(X_i|PA_i), \\
H(X_i|PA_i) &= -\sum_{x_i, pa_i} N_{x_i, pa_i} \log \frac{N_{x_i, pa_i}}{N_{pa_i}}, \\
K(X_i|PA_i) &= (r_i - 1) \prod_{X_l \in PA_i} r_l.
\end{aligned}
$$

$$
MDL(G) = \sum_i MDL(X_i|PA_i),
$$

- Shortest path = Minimum score of (X|PA).
- Optimal Bayesian Network

Local Scores

- Bayesian Networks learning problem NP-hard, and a combinatorial problem. (TSP for example)
- Local score table is also combinatorial. Calculate score for each variable with any possible parents combination.


alarm1000_BIC.txt

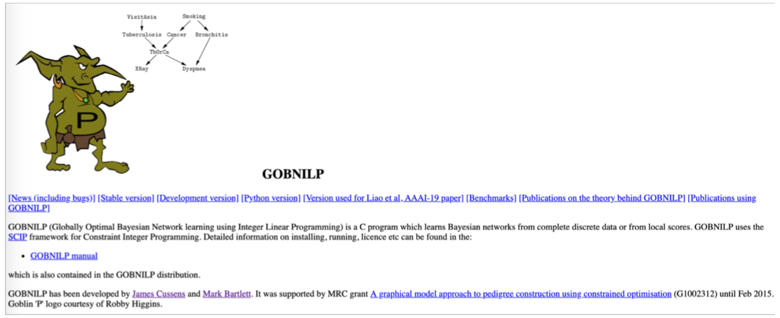
```
37
0 5
-44.683147 1 9
-72.825462 0
-70.135780 1 30
-58.423645 1 1
-40.822388 2 1 9
1 2
-457.492218 0
-443.090424 1 0
2 9
-159.917801 1 31
-164.909973 1 25
-163.621658 1 33
-164.989517 0
-161.365311 1 10
-162.741516 1 13
-162.692688 1 12
-163.594315 1 8
-164.429993 1 15
3 18
-285.276031 1 29
-270.173828 2 15 29
-181.011871 2 10 29
-250.770996 2 8 15
-322.052734 2 8 13
-298.101074 2 8 12
-161.845535 2 9 10
-155.710602 2 8 10
-325.055725 2 7 8
-251.341965 1 33
-220.139801 2 29 33
-328.760284 0
```

- Format of local score: POPS
- Using some pruning tech to constraint the possible parent sets.
- For massive datasets, we can determine the in-degree number of parents.
- Calculating the POPS is very time-consuming for large dataset.

Algorithms(Local Score Based)

- Integer Linear Programming(GOBNIP):

  https://www.cs.york.ac.uk/aig/sw/gobnilp/

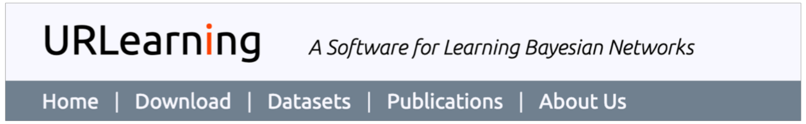


- Depth-first Branch-and-Bound(DFBnB)

  https://cs.uwaterloo.ca/~vanbeek/Research/research_ml.html

- Breadth-first BnB (improve the scalability)
- A* search

  http://urlearning.org/



- Dynamic Programming(DP)

  2005 by Ajit P. Singh and Andrew W. Moore.

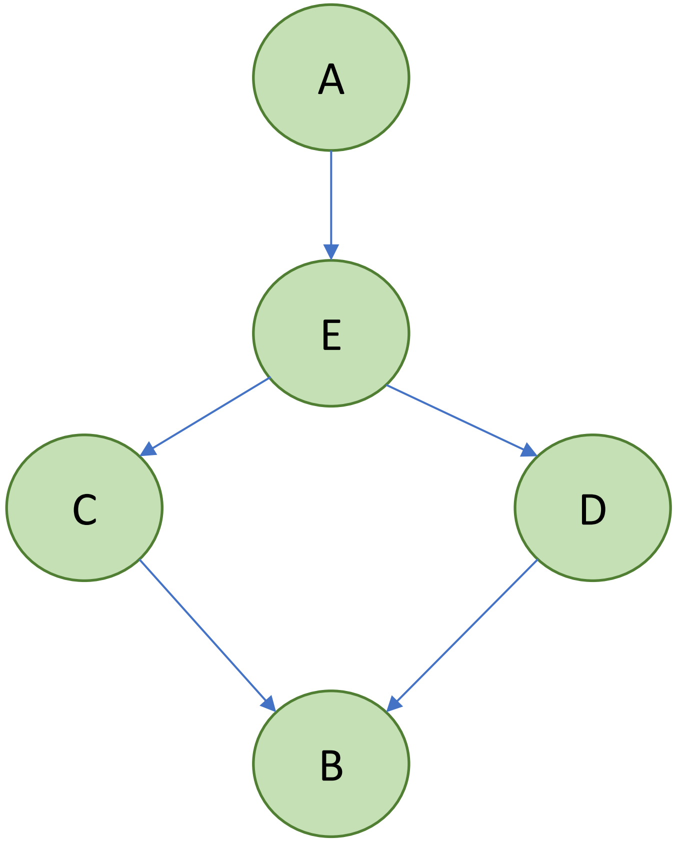
Algorithms(Local Score Based)

- Neural Network
  - I am working one this area now
  - Not find anyone done this before
  - Other combinatorial problem was solved by this method (Google Brain)
  - RNN/ PointerNetwork/ Transformer …
  - Hopefully will deal with thousands of variables with better network compared to other methods.
  - Increase the scalability using GPU.
  - In the future, skip the local score form, combining with reinforcement learning to achieve goal.

A* search and constraints

➢ **A representation of global optimal search**

A	{D} 9.6	{C} 9.9	{E} 10.0	{} 15.4	
B	{C,D} 12.1	{C} 12.2	{E} 12.3	{} 14.1	
C	{E} 3.6	{D} 5.2	{A,B} 10.9	{A} 11.4	{} 17.0
D	{E} 3.6	{C} 5.2	{A,B} 10.9	{A} 11.4	{} 17.0
E	{D} 3.7	{A} 4.2	{A,B} 11.2	{C} 11.6	{} 17.0

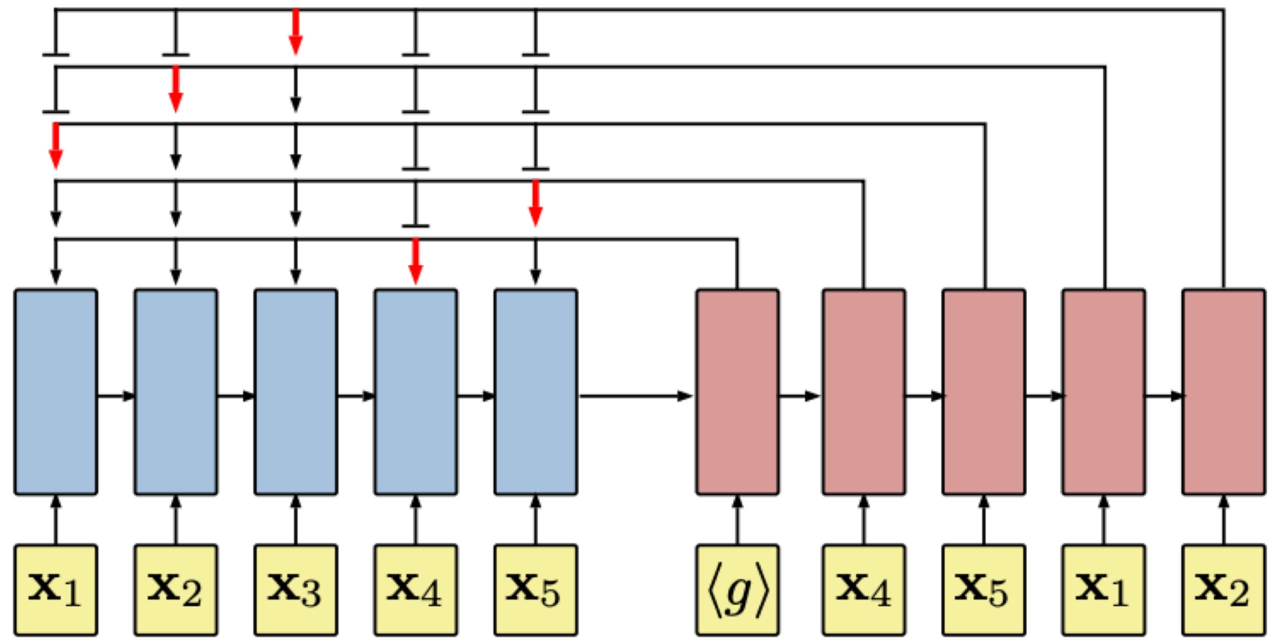


Score: 38.9

- Heuristic (DP)
- Upper bound (local search)

- Symmetry-breaking
- Dominance constraints

NN for BNs Learning



- By *'NEURAL COMBINATORIAL OPTIMIZATION WITH REINFORCEMENT LEARNING'*
- Applied on TSP and Knapsack

- Learning order of the variables.
- Using the order output to search through POPS for scores.
- Optimize the NN via minimizing the score.
- Better score → Better networks

Structure of Survey

- Abstract
- Introduction:
  - Global optimal search
  - Local search
  - Neural network related problems
- Background:
  - Local Score Functions
  - Learning Networks and NP-hard explanation
- Order-based local search algorithm (ILP)
- Dynamic Programming
- Shortest-path concept search
- Mixture of algorithms and their improvement
- Neural Network related search
- Comparison of different algorithms
- References

Papers

- Irwan Bello, Hieu Pham, Quoc V. Le, Mohammad Norouzi, Samy Bengio. '*Neural Combinatorial Optimization With Reinforcement Learning'.*
- Oriol Vinyal, Meire Fortunato, Navdeep Jaitly. '*Pointer Networks*'.
- Mauro Scanagatta, Cassio P. de Campos, Giorgio Corani. '*Learning Bayesian Networks with Thousands of Variables*'.
- Peter van Beek, Hella-Franziska Hoffmann. '*Machine learning of Bayesian networks using constraint programming*'
- Changhe Yuan, Brandon Malone, and Xiaojian Wu. '*Learning Optimal Bayesian Networks Using A* Search'*

Questions?