Structure Learning of Bayesian Networks using Genetic Algorithms

(Final Presentation)

Presenters:

Biniam Gebru
Mrinmoy Sarkar

Advisor: Dr. Abdollah Homaifar

Spring, 2018

Outline

- **Part I – Background**
  - What was the Problem?
    - What are Bayesian Networks?
    - What is Structure Learning?
  - Summary of Various Approaches
  - Structure Learning using GA
  - Implementation/Simulation details

- **Part II: Results & Discussion**
  - Simulation Results
  - Discussion & Conclusions

Part I - Background

Problem Statement

- Searching the Bayesian Network (BN) that best reflects the dependence relations in a database of cases.
  - The problem of searching the BN that best reflects the dependence relations in a database of cases is a difficult one because of the large number of possible DAG structures, given even a small number of nodes to connect.
- BNs are directed acyclic graphs (DAGs), where the nodes are random variables and where the arcs specify the independence assumptions between these variables.
- After construction, a BN constitutes an efficient device for performing probabilistic inference.

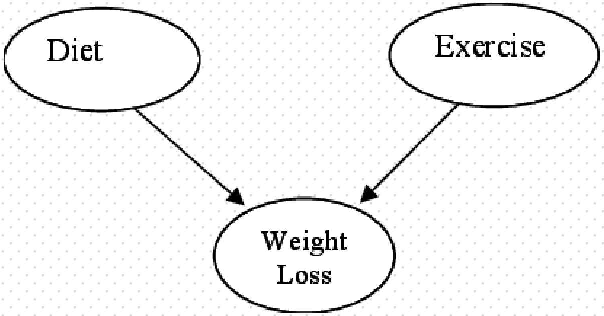
Given:

– *A set of data example, sampled from a Bayesian Network*

- *E.g.: Diet=Low, Exercise=No, Weight Loss = Low; Diet=High, Exercise=Yes, Weight Loss = High;*

Unknown:

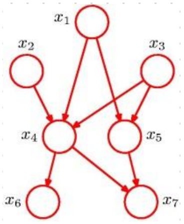
– *The Bayesian Network from which the data is sampled*



Part I …

Bayesian Networks

- Provide us with efficient representation and inference
  - ✓ Exploiting dependency **structure** makes it easier to represent and compute with probabilities
- Represent causal dependencies among variables - *causal* graphical models
  - ✓ Express underlying causal structure
  - ✓ Causality simplifies inference
    - ❑ Using a representation in which the direction of causality is correct produces sparser graphs



$$p(\mathbf{x}) = \prod_{k=1}^{K} p(x_k|\mathrm{pa}_k)$$

❖ Learning causal graphical models

- ✓ **Strength:** how strong is a relationship?
- ✓ **Structure:** does a relationship exist?

Structure Learning

The number of possible network structure for n node of a Bayesian network can be found by using the following formula:

$$f(n) = \sum_{i=1}^{n} (-1)^{i+1} \frac{n!}{(n-i)!\,n!} 2^{i(n-i)} f(n-1)$$

Nodes	Number of DAGs	Nodes	Number of DAGs
1	1	13	$1.9 \cdot 10^{31}$
2	3	14	$1.4 \cdot 10^{36}$
3	25	15	$2.4 \cdot 10^{41}$
4	543	16	$8.4 \cdot 10^{46}$
5	29281	17	$6.3 \cdot 10^{52}$
6	$3.8 \cdot 10^{6}$	18	$9.9 \cdot 10^{58}$
7	$1.1 \cdot 10^{9}$	19	$3.3 \cdot 10^{65}$
8	$7.8 \cdot 10^{11}$	20	$2.35 \cdot 10^{72}$
9	$1.2 \cdot 10^{15}$	21	$3.5 \cdot 10^{79}$
10	$4.2 \cdot 10^{18}$	22	$1.1 \cdot 10^{87}$
11	$3.2 \cdot 10^{22}$	23	$7.0 \cdot 10^{94}$
12	$5.2 \cdot 10^{26}$	24	$9.4 \cdot 10^{102}$

Part I …

Various methods

- Evolutionary algorithms(e.g.: GA)
- Neural network
- Simulated annealing
- Tabu search
- Target analysis

Structure Learning Using GA

- Representation of candidate solution
- Crossover operation
- Mutation operation

Part I …

Crossover operation

Suppose we have two candidate solution given below:

P1 = 000000|110

P2 = 001001|000

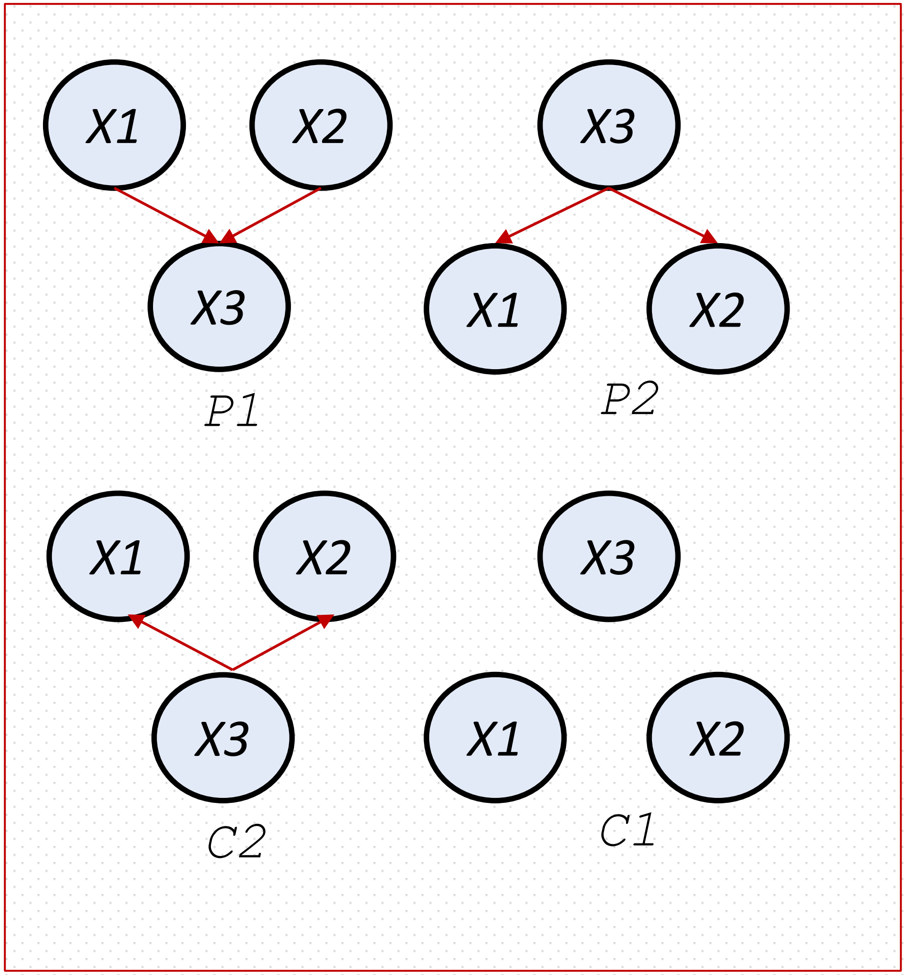
If we choose crossover point at k=6 then,

C2 = 001001110

C1 = 000000000

But C1 is not valid structure

Remark: So, crossover is not a closed operator for this problem.



Part I …

Mutation operation

Suppose we have one candidate solution as given below:

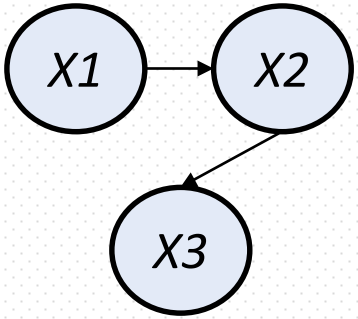
P1 = 010001000

Now if we flip the red bit then the resultant child is:

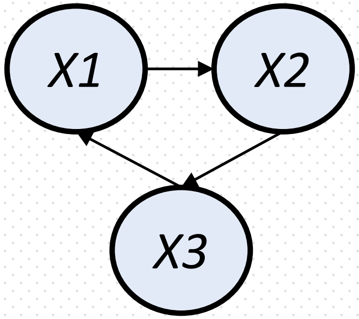
C1 = 010001100

Which is not a valid structure.

Remark: So, mutation is not a closed operator too for this problem.



P1



C1

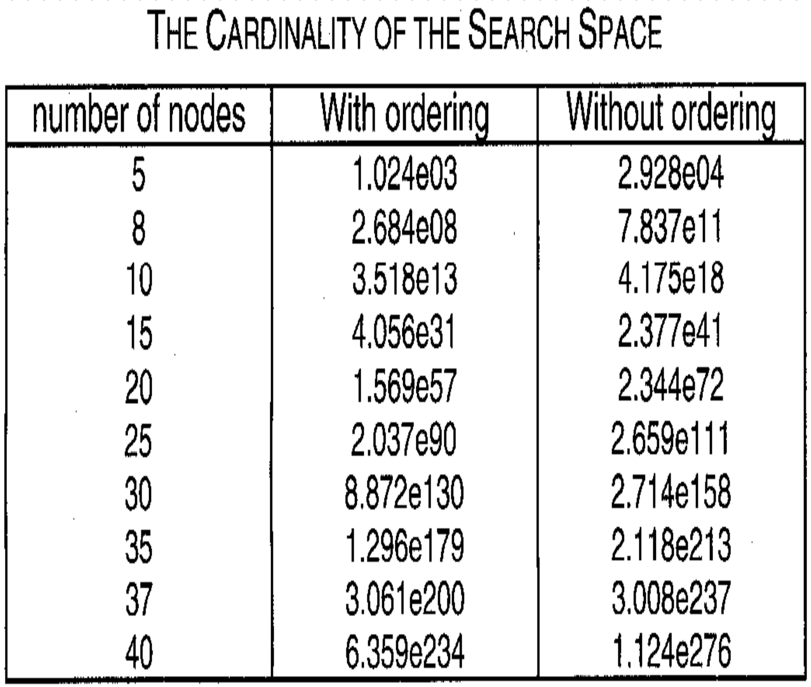
Part I …

Solution Using GA Operators

- If we consider ordering between nodes the problem we faced before can be solved.
- Ordering means node Xi can only have node Xj as a parent node if in the ordering node Xj comes before Xi.
  - e.g.: suppose we have a order for three node as X2,X1,X3 then X2 will never have any parent, X1 can have X2 as parent and X3 can have X1 and X2 as its parent
- For this solution the no. of bits for representing a candidate solution reduces to $\binom{n}{2}$ from n^2 for general case for n nodes.

Remark: Now, both crossover and mutation are closed operator for this problem.

Consequence of Ordering



THE CARDINALITY OF THE SEARCH SPACE

number of nodes	With ordering	Without ordering
5	1.024e03	2.928e04
8	2.684e08	7.837e11
10	3.518e13	4.175e18
15	4.056e31	2.377e41
20	1.569e57	2.344e72
25	2.037e90	2.659e111
30	8.872e130	2.714e158
35	1.296e179	2.118e213
37	3.061e200	3.008e237
40	6.359e234	1.124e276

Remark: The problem is still NP hard

Part II - Results

- Simulation Model
- Simulation Results
- Discussion & Conclusion

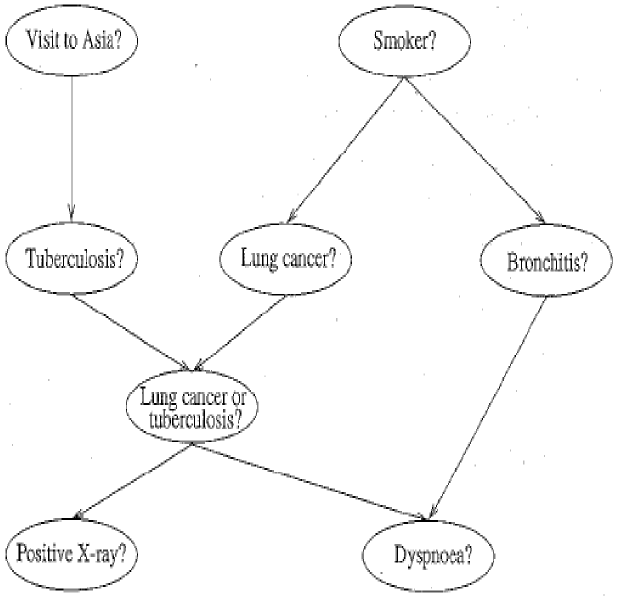
Part II - Results

Implementation/Simulation details

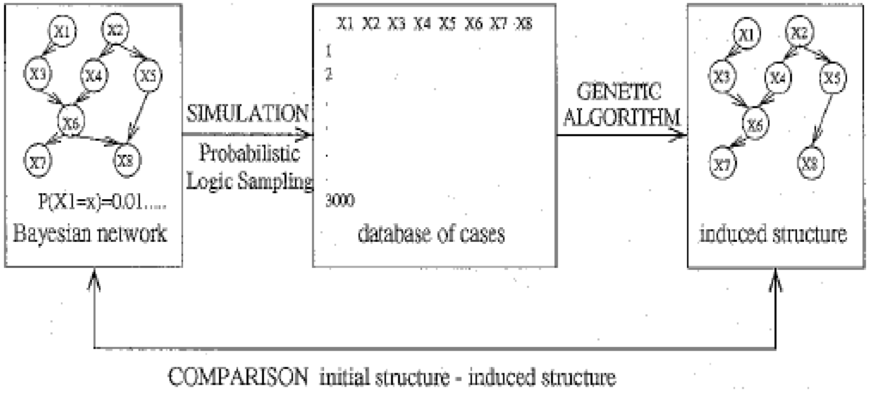
1. Determinate a BN (Structure + conditional probabilities) and simulate it, obtain a database of cases D, which must reflect the conditional independence relations between the variables.
2. Using GA find BN structure B^* , which maximizes the probability $P(D|B)$
3. Evaluate the fitness of the solutions found

Part II

ASIA Network Structure



Simulation Model



Remark: The problem is still NP hard

$$\mathrm{BIC}(S \mid \mathcal{D}) = \sum_{i=1}^{N} \log_2 P(\mathbf{d}_i \mid \hat{\theta}_S, S) - \frac{\mathrm{size}(S)}{2} \log_2(N).$$

Part II - Simulation Result(Cancer Network)

Population Size	Data Sample Size	$-logP(D\|B*)$	$-logP(D\|Bo)$	K2 score(B*)	K2 score(B°)
10	500	316.53	499.69	-743.46	-1177.92
10	3000	2074.14	2880.55	-4813.96	-6669.78
50	500	103.97	474.54	-249.20	-1119.81
50	3000	1050.17	2921.04	-2430.56	-6762.44

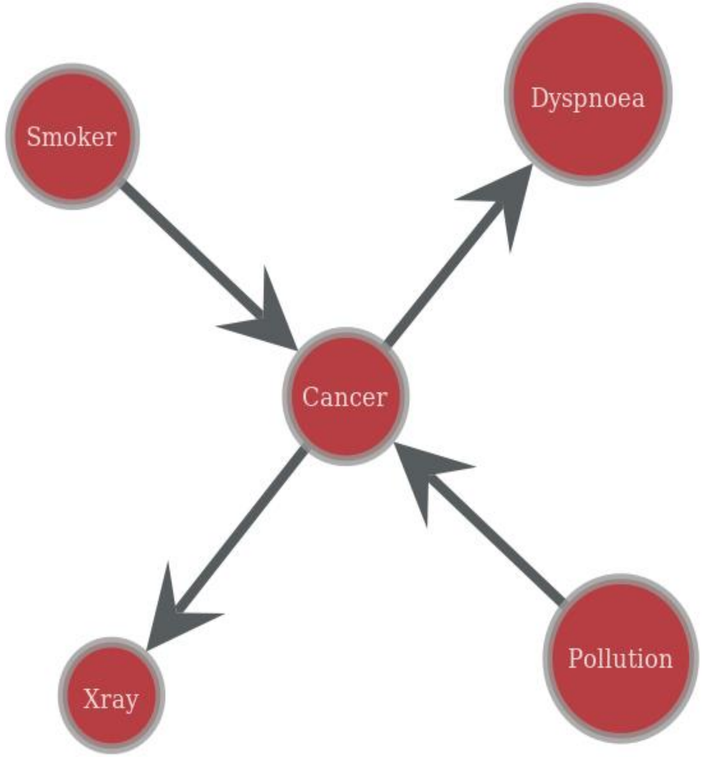
No. of generation = 100, pc = 0.5, pm=0.01,

Part II - Simulation Result(Cancer Network)

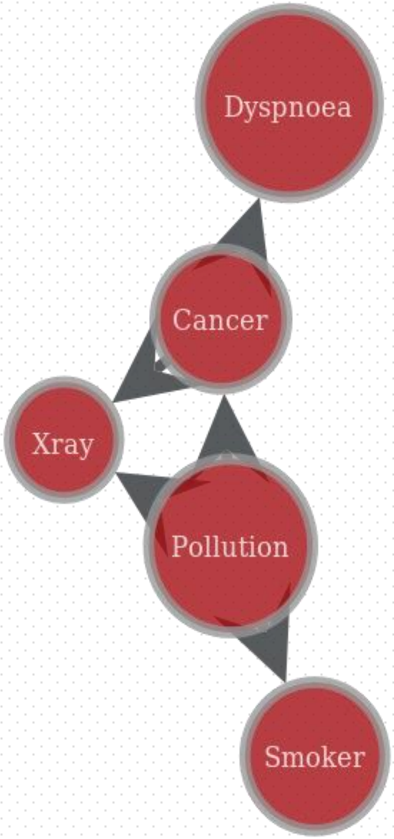
Population Size	Data Sample Size	$-logP(D\|B*)$	$-logP(D\|Bo)$	K2 score(B*)	K2 score(Bo)
10	500	235.27	462.42	-554.82	-1091.81
10	3000	2743.67	2901.87	-6339.27	-6719.01
50	500	367.89	490.68	-868.31	-1156.62
50	3000	1201.55	2859.99	-2777.47	-6622.12

No. of generation = 100, pc = 0.9, pm=0.01,

Part II – Sample Solution



True network



Found network

Simulation Result(Asia Network)

Population Size	Data Sample Size	$-logP(D\|B*)$	$-logP(D\|Bo)$	K2 score(B*)	K2 score(B^o)
10	500	592.26	513.28	-1426.52	-1230.89
10	3000	2960.26	2964.81	-6922.40	-6894.1545
50	500	398.29	469.03	-963.92	-1128.87
50	3000	2201.76	2983.63	-5161.37	-6937.31

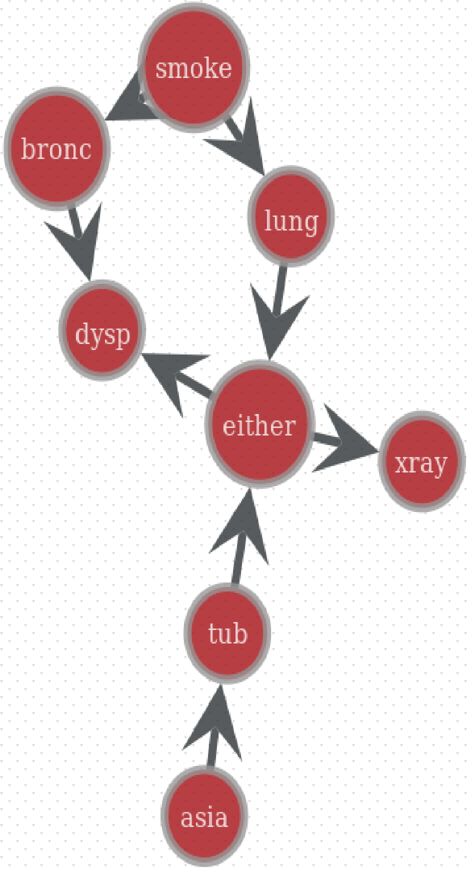
No. of generation = 100, pc = 0.5, pm=0.01,

Simulation Result(Asia Network)

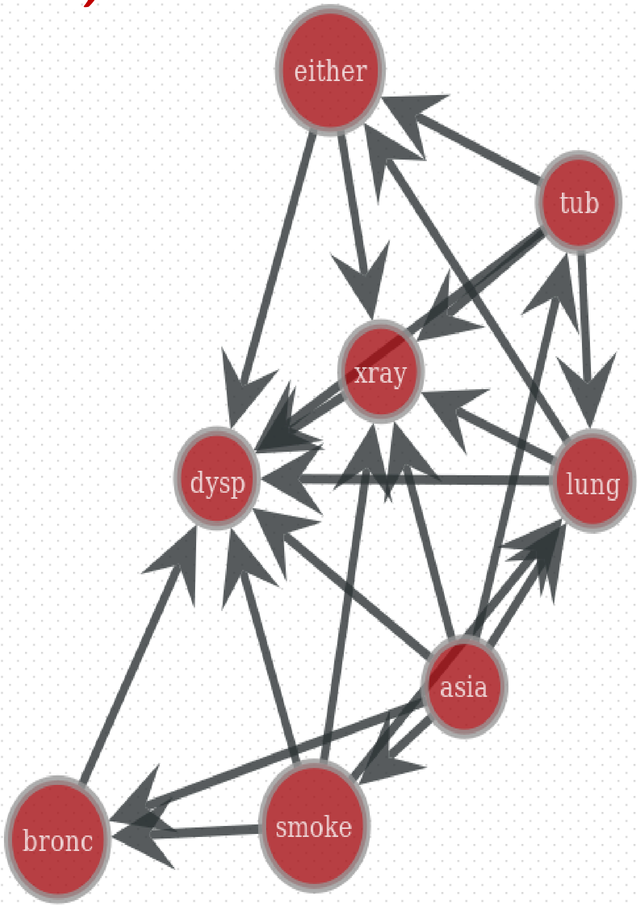
Population Size	Data Sample Size	$-logP(D\|B*)$	$-logP(D\|Bo)$	K2 score(B*)	K2 score(B^o)
10	500	528.94	490.06	-1271.78	-1177.06
10	3000	3074.21	2952.07	-7203.02	-6864.96
50	500	339.72	492.11	-813.26	-1182.66
50	3000	2454.09	3020.86	-5716.59	-7022.89

No. of generation = 100, pc = 0.9, pm=0.01,

Sample Solution(Asia Network)



True network



Found network

Observation

- Increasing the data sample size gives better result
- Increasing the no of population doesn’t result in significantly improved change in performance (mainly depends on how fit the samples are to the found structure) but the computation becomes slower.
- Increasing the crossover probability gives better result

References

1) Pedro Larranaga, et al, *Structure Learning of Bayesian Networks by Genetic Algorithms: A performance analysis of Control Parameters*, IEEE Transactions of pattern Analysis and Machine Intelligence, Vol. 18, No. 9, Sept, 1996

2) Ben-Gal I., *Bayesian Networks*, Encyclopedia of Statistics in Quality and Reliability, Wiley & Sons, 2007

3) https://www.bayesserver.com/docs/introduction/bayesian-networks Retrieved on 18th March, 2018