STATISTICS IN AI/ML PROJECT

Comparison of Different Structure Learning Algorithms for Bayesian Networks

INSTRUCTOR:

Vibhav Gogate

Arunkumar Rajappan

AXR138930

Spring 2014

1. Introduction

In this project we will be analyzing and comparing the results of different Structure learning algorithms for Bayesian networks using "bnlearn" library function available in R tool.

Package installation Instructions

Bnlearn package is available on CRAN and can be downloaded from its web page in the Packages section.

It can be installed with a simple:

```
install.packages("bnlearn")
```

The only suggested packages not hosted on CRAN are graph and Rgraphviz, which can be installed from BioConductor:

```
source("http://bioconductor.org/biocLite.R")
biocLite("Rgraphviz")
```

2. Algorithms Used

I have used four categories of structure learning algorithm in this project, which are described below:

I. Constraint-based learning algorithms

These algorithms learn the network structure by analyzing the probabilistic relations entailed by the Markov property of Bayesian networks with conditional independence tests and then constructing a graph which satisfies the corresponding d-separation statements. The resulting models are often interpreted as causal models even when learned from observational data.

a. Grow-Shrink (GS) algorithm

Based on the Grow-Shrink Markov Blanket, the simplest Markov blanket detection algorithm used in a structure learning algorithm.

b. Incremental Association Marklov Blanket (IAMB)

based on the Incremental Association Markov blanket (IAMB) algorithm (Tsamardinos et al. 2003), which is based on a two-phase selection scheme (a forward selection followed by an attempt to remove false positives).

c. Fast Incremental Association Marklov Blanket (Fast-IAMB)

A variant of IAMB which uses speculative stepwise forward selection to reduce the number of conditional independence tests

d. Interleaved Incremental Association Marklov Blanket (Inter-IAMB)

Another variant of IAMB which uses forward stepwise selection (Tsamardinos et al. 2003) to avoid false positives in the Markov blanket detection phase

II. Score-based learning algorithms

These algorithms assign a score to each candidate Bayesian network and try to maximize it with some heuristic search algorithm. Greedy search algorithms (such as hill-climbing or tabu search) are a common choice, but almost any kind of search procedure can be used.

a. Hill Climbing (HC) algorithm

III. **Hybrid learning algorithms**
   a. **Max-Min Hill Climbing (MMHC) algorithm**
   b. **Two-Phase Restricted Maximization (RSMAX2) algorithm**

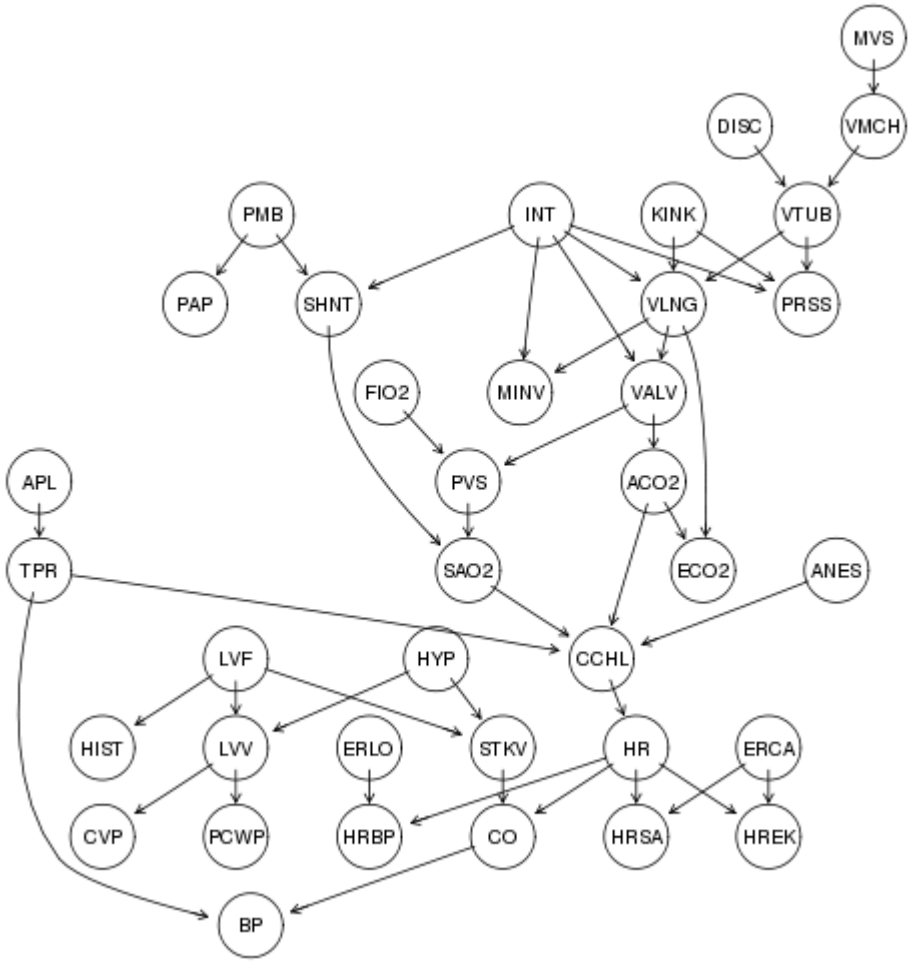
IV. **Local discovery algorithms**
   a. **Chow-Liu learning algorithm**

3. Datasets Used

For experimenting we used the aready available Bayesian networks available in "bnleearn" library for R tool.

http://www.bnlearn.com/bnrepository/

a) **Alarm** – The ALARM ("A Logical Alarm Reduction Mechanism") is a Bayesian network designed to provide an alarm message system for patient monitoring.



ALARM
Number of nodes: 37
Number of arcs: 46
Number of parameters: 509

manual
page: alarm.html

b) **Insurance** – Insurance is a network for evaluating car insurance risks.

Insurance

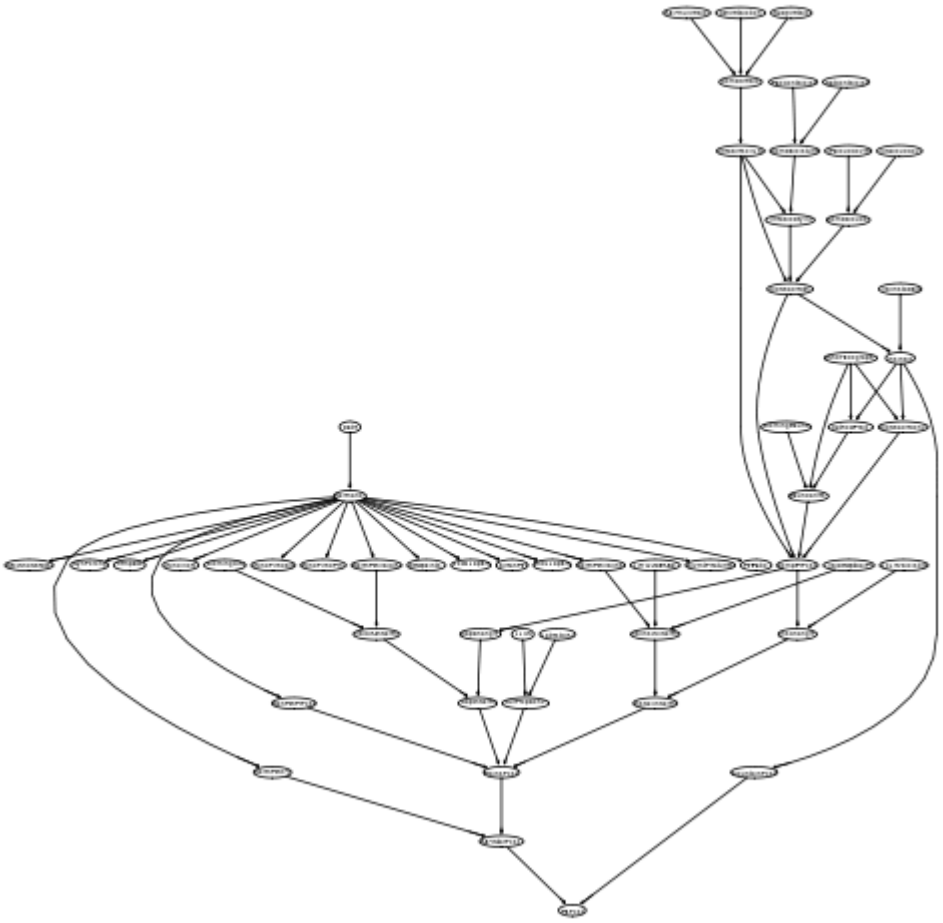
Number of nodes: 27

Number of arcs: 52

Number of parameters: 984

manual page: insurance.html

c) **Hailfinder** – Hailfinder is a Bayesian network designed to forecast severe summer hail in northeastern Colorado.



Hailfinder

Number of nodes: 56

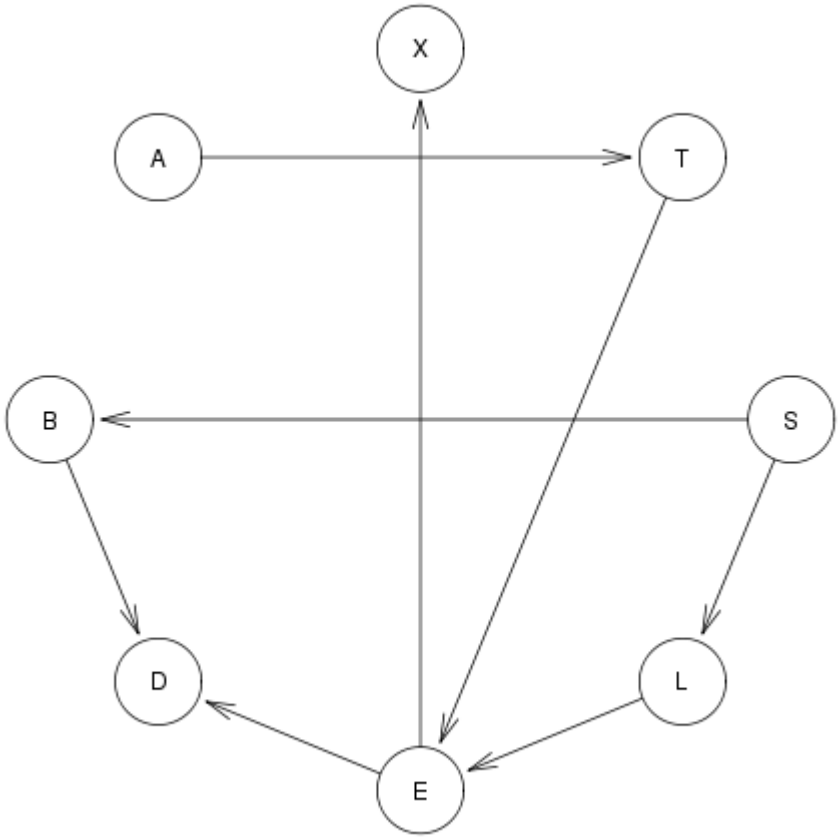
Number of arcs: 66

Number of parameters: 2656

manual

page: hailfinder.html

d) **Asia** – Small synthetic data set from Lauritzen and Spiegelhalter (1988) about lung diseases (tuberculosis, lung cancer or bronchitis) and visits to Asia.



Asia

Number of nodes: 8

Number of arcs: 8

Number of parameters: 18

manual page: asia.html

4. Results and Comparisons

We considered following parameters while comparing and evaluation the SL algorithms for different datasets:

1. **Compare two different Bayesian networks**
   We used the "compare" function of bnlearn library for comparing the skeletons/models generated by different algorithms. For this we considered the model generated by Hill-Climbing algorithm as the basis for comparing the models generated by other algorithms. We also measured the hamming distance between the two models.

   compare function returns a list containing the number of true positives (tp, the number of arcs in current also present in target), of false positives (fp, the number of arcs in current not present in target) and of false negatives (tn, the number of arcs not in current but present in target) if arcs options is FALSE; or the corresponding arc sets if arcs option is set TRUE. shd and hamming return a non-negative integer number.

For Alarm Dataset	True Positive#	False Positive#	False Negative#	Hamming Distance
HC	53	0	0	0
GS	6	18	47	50
IAMB	17	18	36	37
F-IAMB	19	16	34	32
I—IAMB	19	20	34	38
MMHC	25	10	28	33
RSMAX2	11	12	42	51
Chow-Liu	0	36	53	45

For HailFinder Dataset	True Positive#	False Positive#	False Negative#	Hamming Distance
HC	64	0	0	0
GS	-	-	-	-
IAMB	24	12	40	40
F-IAMB	24	15	40	45
I—IAMB	24	11	40	41
MMHC	35	15	29	48
RSMAX2	26	0	38	47
Chow-Liu	0	55	64	71

For Insurance Dataset	True Positive#	False Positive#	False Negative#	Hamming Distance
HC	50	0	0	0
GS	-	-	-	-
IAMB	11	19	39	38
F-IAMB	14	18	36	36
I—IAMB	9	24	41	41
MMHC	27	6	23	41
RSMAX2	18	1	32	46

Chow-Liu	0	26	50	48

For Asia Dataset	True Positive#	False Positive#	False Negative#	Hamming Distance
HC	7	0	0	0
GS	-	-	-	-
IAMB	3	1	4	5
F-IAMB	3	1	4	5
I—IAMB	3	1	4	5
MMHC	5	1	2	3
RSMAX2	4	0	3	4
Chow-Liu	0	7	7	6

2. **Execution time**

In this we calculated, CPU time utilized by each algorithm on individual dataset

For ALARM Dataset	User time	System Time	Elapsed Time
HC	0.8	0	0.78
GS	0.06	0	0.07
IAMB	1.66	0	1.66
F-IAMB	1.58	0	1.58
I—IAMB	1.70	0	1.71
MMHC	1.04	0	1.05
RSMAX2	1.66	0.00	1.66
Chow-Liu	0.08	0	0.08

For HailFinder Dataset	User time	System Time	Elapsed Time
HC	1.47	0	1.47
GS	0.06	0	0.07
IAMB	3.08	0	3.08
F-IAMB	2.81	0	2.82
I—IAMB	3	0	3

MMHC	2.18	0	2.19
RSMAX2	5.78	0.00	5.82
Chow-Liu	0.13	0	0.13

For insurance Dataset	User time	System Time	Elapsed Time
HC	0.58	0	0.58
GS	0.08	0	0.08
IAMB	0.81	0	0.81
F-IAMB	0.87	0	0.88
I—IAMB	0.86	0	0.87
MMHC	0.92	0	0.92
RSMAX2	0.81	0.0	0.81
Chow-Liu	0.05	00.0	0.05

For Asia Dataset	User time	System Time	Elapsed Time
HC	0	0	0
GS	0.06	0	0.06
IAMB	0.03	0	0.03
F-IAMB	0.03	0	0.04
I—IAMB	0.02	0.0	.02
MMHC	0.01	0	0.01
RSMAX2	0.03	0.00	0.03
Chow-Liu	0	0	0

3. **Log-Likelihood/Entropy Score**
   Calculated the log likelihood of the generated model to the original model:

Log Likelihood Values	ALARM	HAILFINDER	INSURANCE	ASIA
HC	-217488.6	-981720.1	-262349.7	-11034.9
GS	-	-	-	-
IAMB	-	-	-	-
F-IAMB	-	-1114889	-	-12466.88
I—IAMB	-	-1216013	-	-12466.88

MMHC	-253775.1	-	-278920	-12016.23
RSMAX2	-335712.1	-	-322420.3	-12231.6
Chow-Liu		-	-	-

4. **K2 Score**
   Calculated the K2 score of the generated model when compared to the original model:

K2 Score Values	ALARM	HAILFINDER	INSURANCE	ASIA
HC	-219549.7	-988692.7	-265243.8	-11109.47
GS	-	-	-	-
IAMB	-	-	-	-
F-IAMB	-	-	-	-12529.7
I—IAMB	-	-	-	-12529.7
MMHC	-254874.5	-1118860	-280453.5	-12086.05
RSMAX2	-336598.6	-1217895	-323321	-12291.37
Chow-Liu	--	-	-	-

5. Conclusion

Based on the above statistics and the generated models provided below, it is clear that Chow-Liu algorithm surpasses all other algorithms in all the possible ways, also the model generated by Chow-Liu algorithm is complete with no dangling nodes.

After Chow-Liu, Hill-climbing and Min-max hill climbing displayed promising results. But yet compared to score based algorithms. Hybrid algorithms and local discovery algorithms, constrain-based algorithms (except grow shrink) showed quite a poor performance.

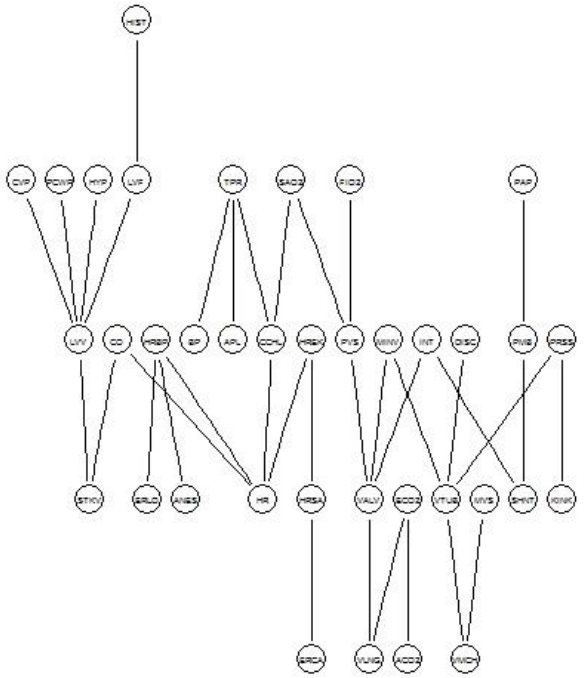
In future we plan to compare these algorithm with much larger datasets (with 100 to 1000 or more nodes) and more scoring criteria's.

http://www.bnlearn.com/bnrepository/

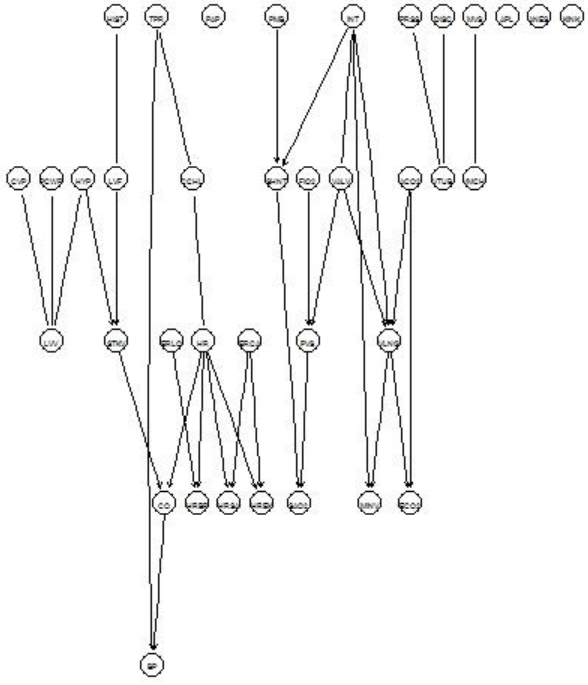
Attached below are the models generated by the different algorithms on input datasets.

a) Models generated for "ALARM" dataset

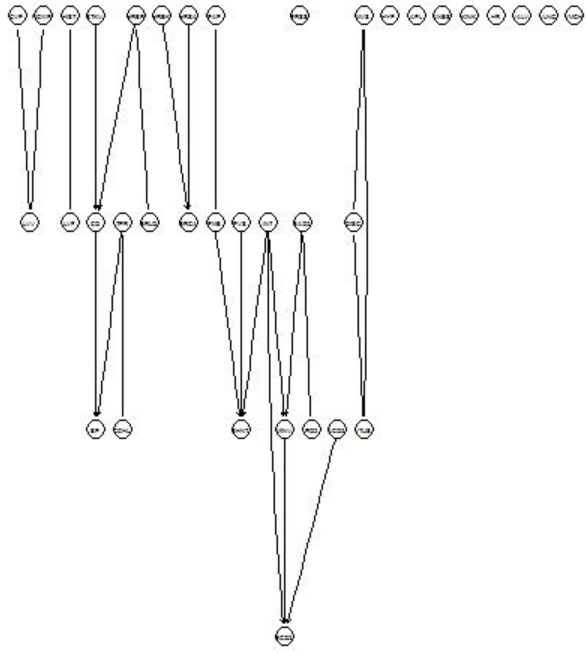
Chow-Liu algorithm



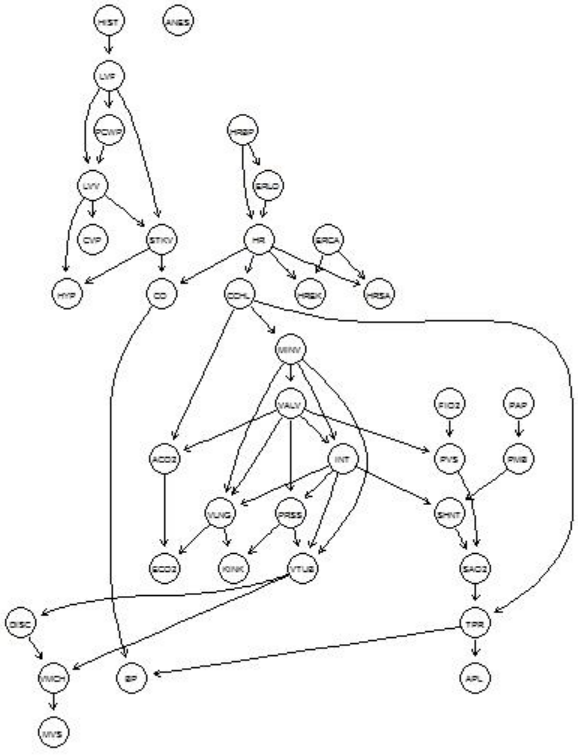
FIAMB algorithm



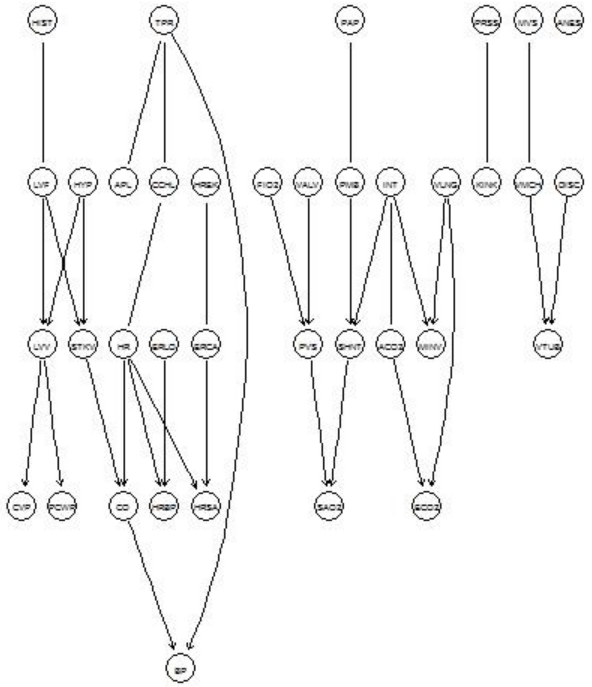
Grow-Shrink algorithm



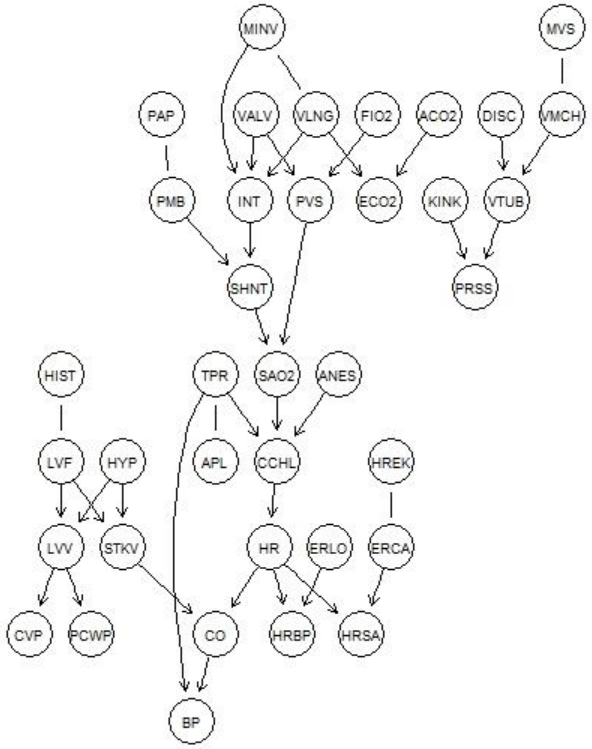
Hill-Climbing algorithm



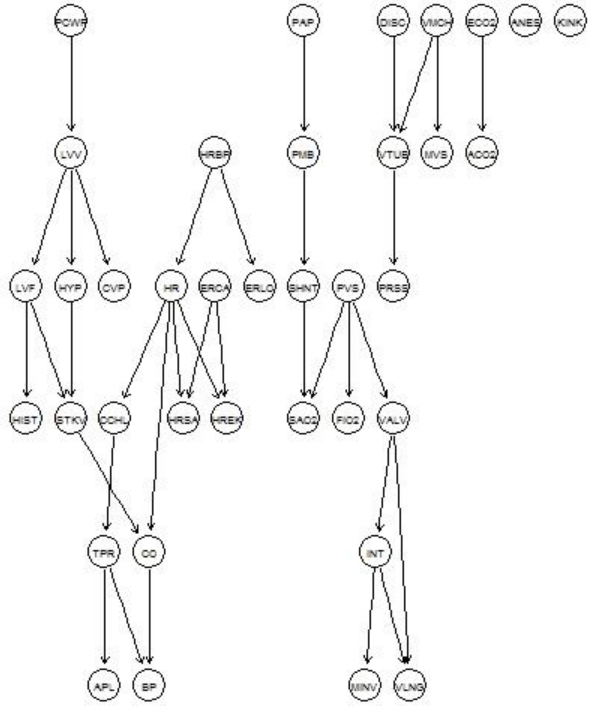
IAMB algorithm



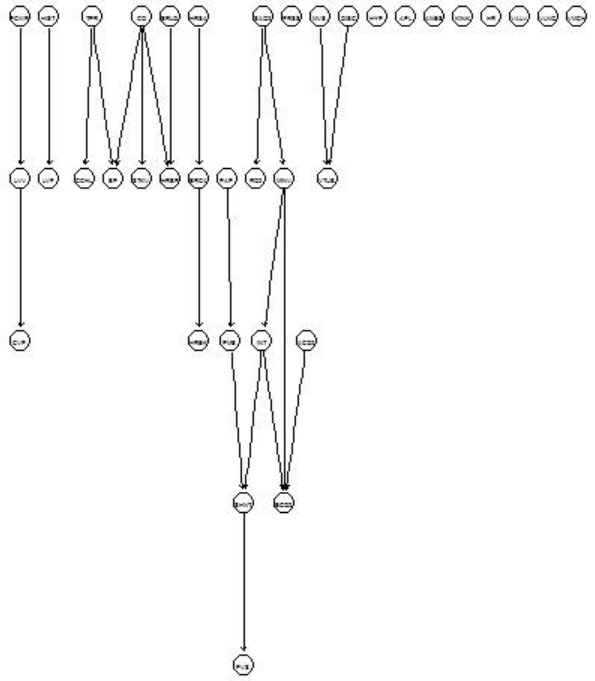
IIAMB algorithm



MMHC algorithm

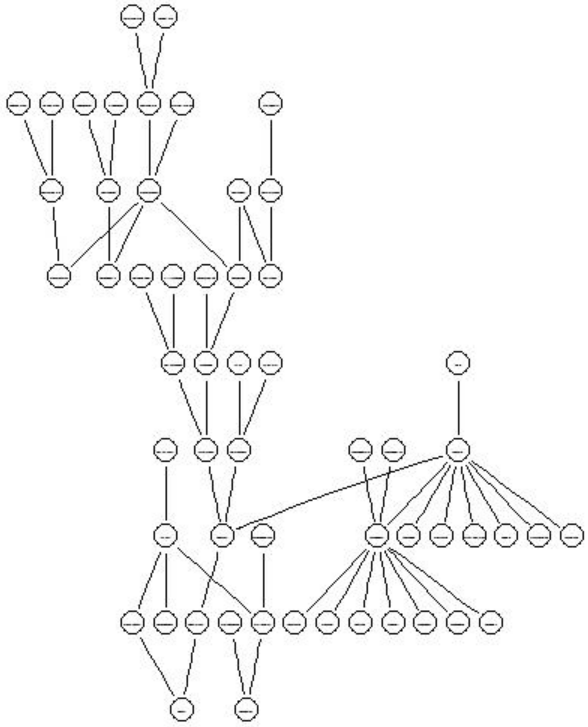


RSMAX2 algorithm

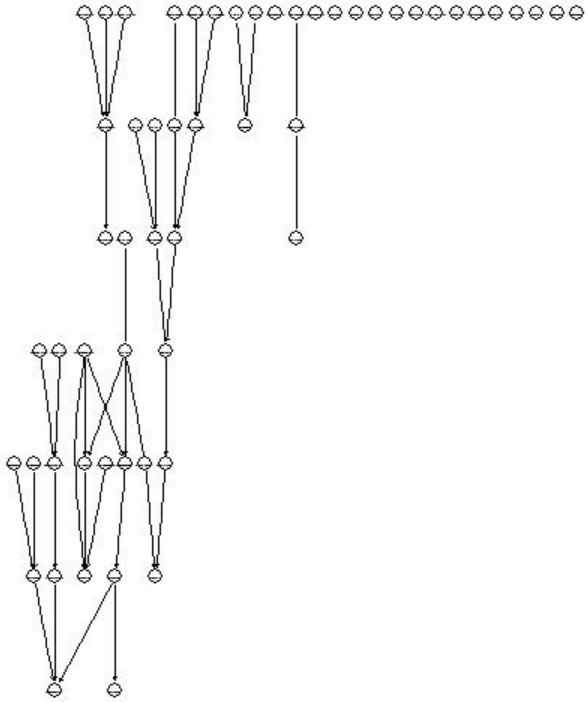


b) Models generated for "HAILFINDER" dataset

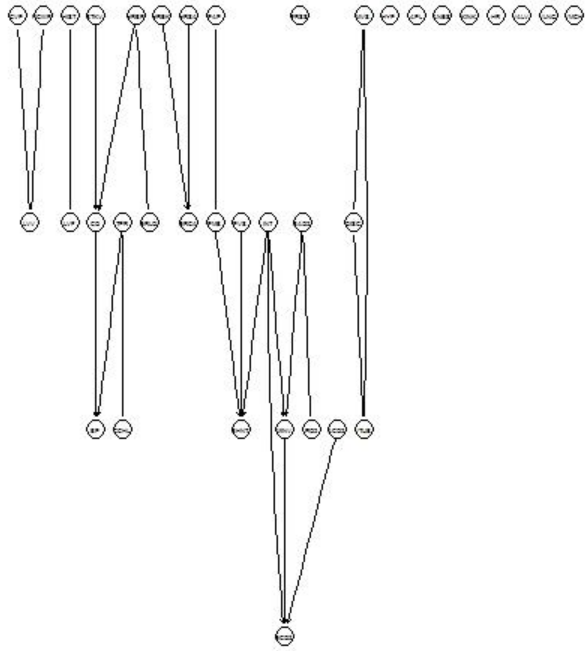
Chow-Liu algorithm



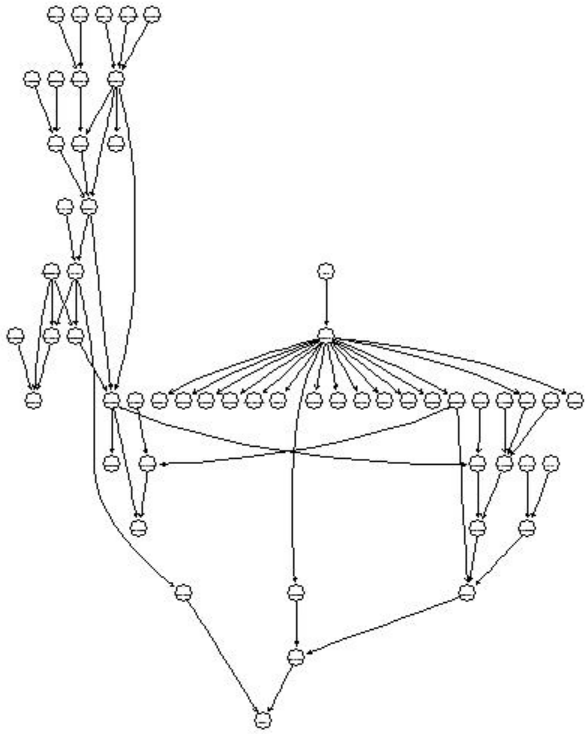
FIAMB algorithm



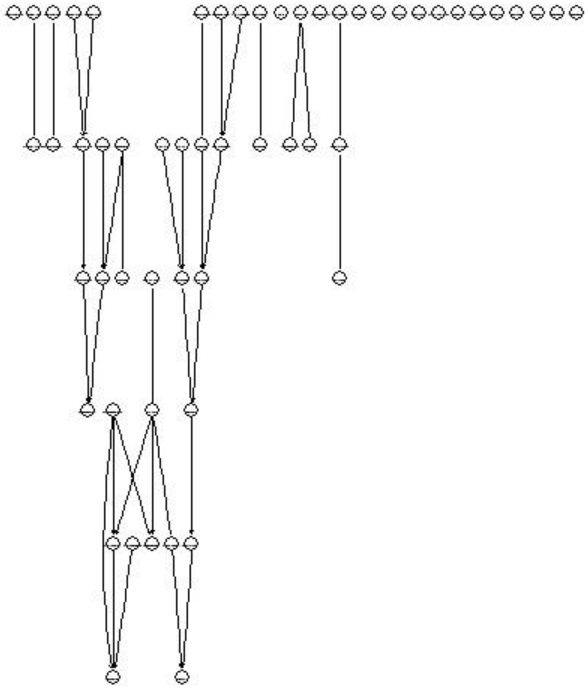
Grow-Shrink algorithm



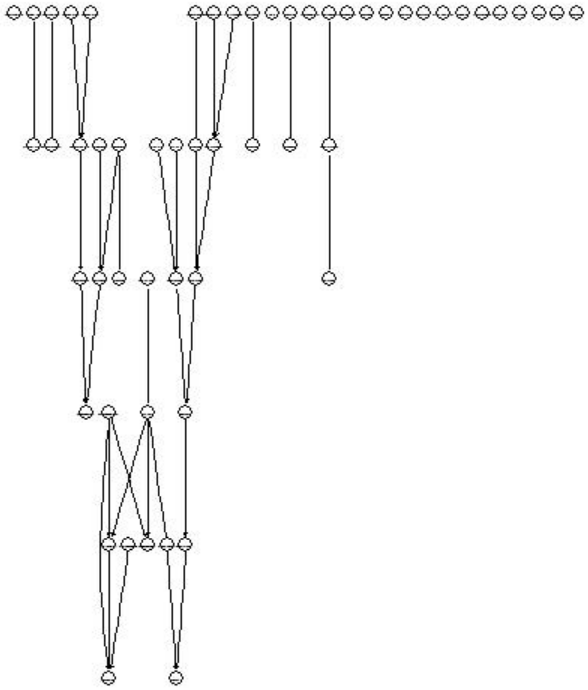
Hill-Climbing algorithm



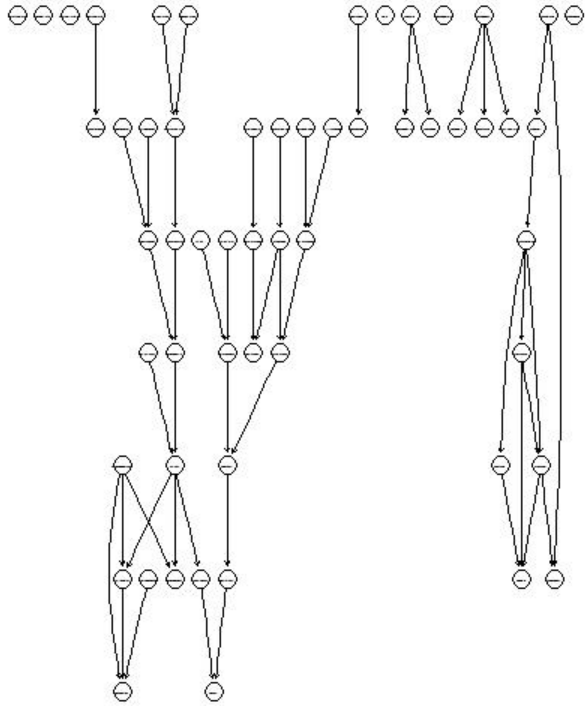
IAMB algorithm



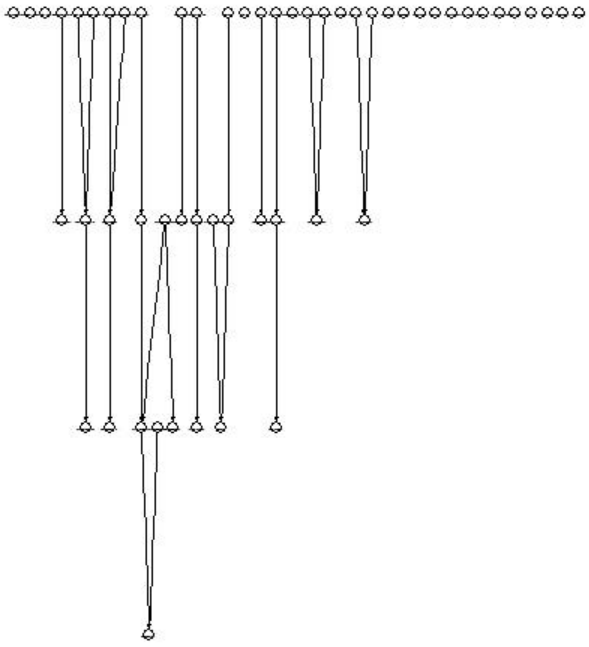
IIAMB algorithm



MMHC algorithm



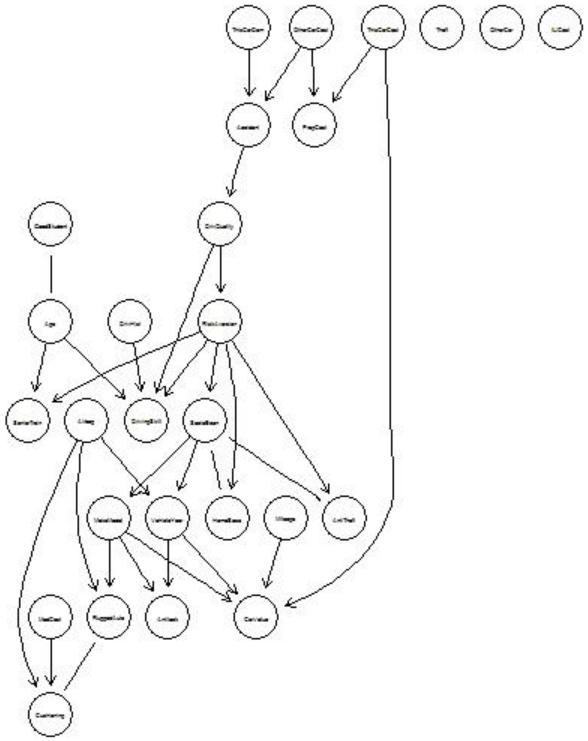
RSMAX2 algorithm



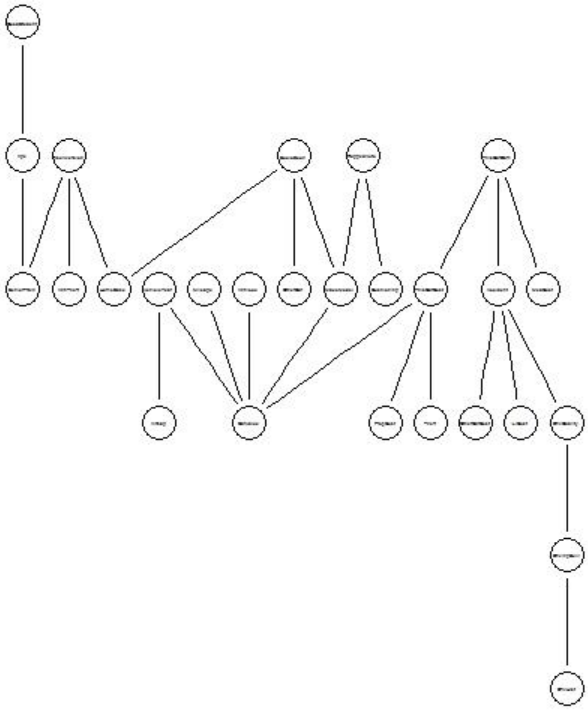
c) Models generated for "INSURANCE" dataset

d)

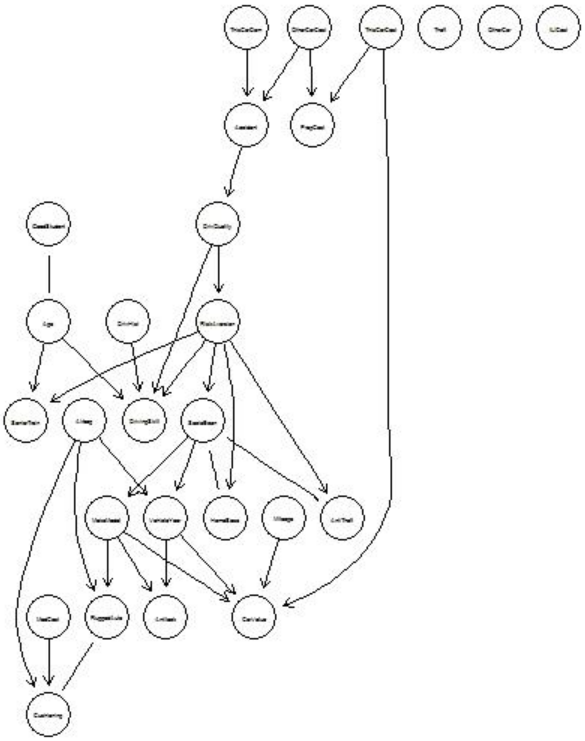
FIAMB algorithm



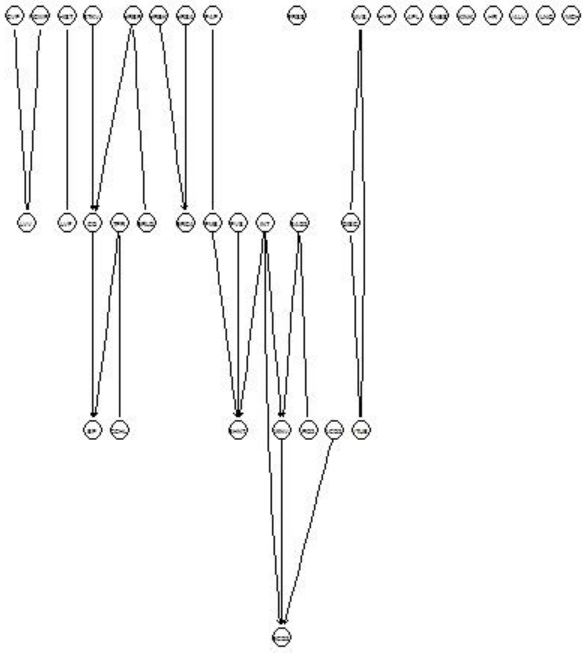
Chow-Liu algorithm



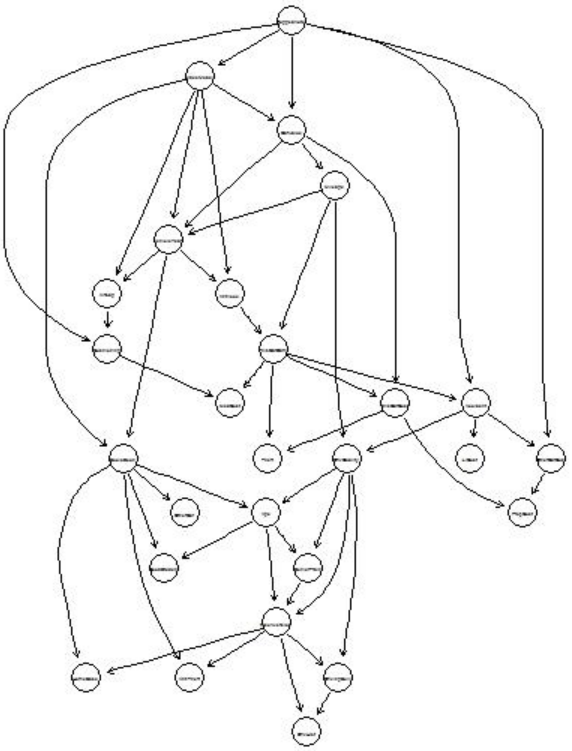
FIAMB algorithm



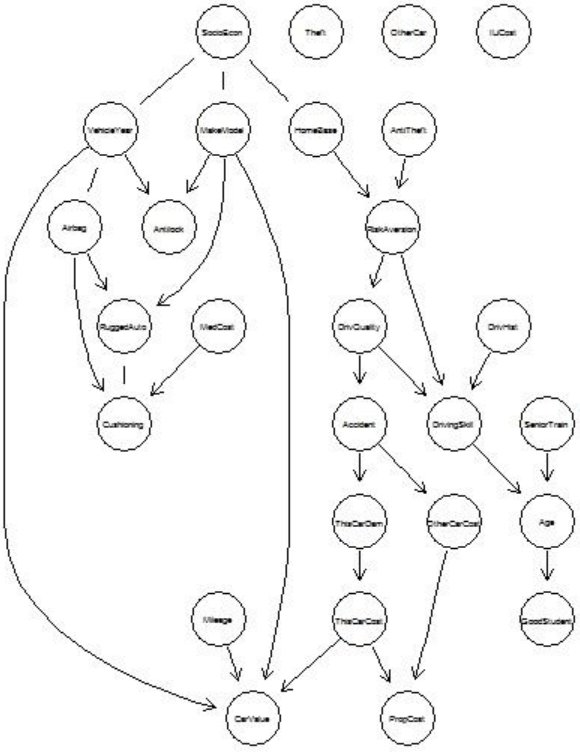
Grow-Shrink algorithm



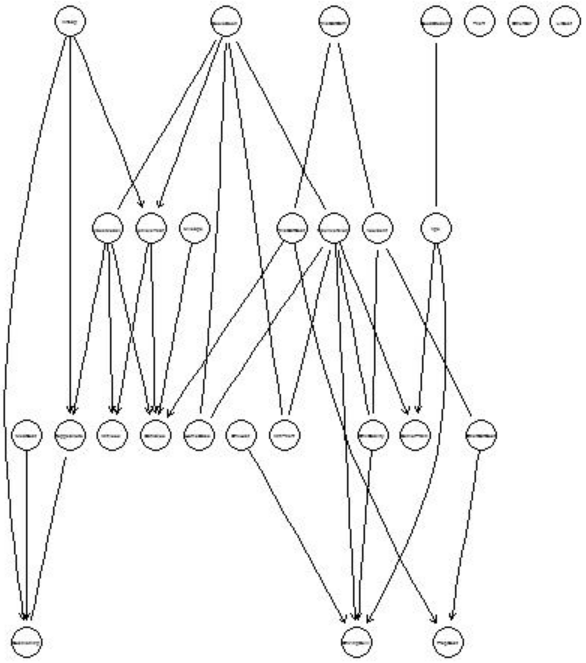
Hill-Climbing algorithm



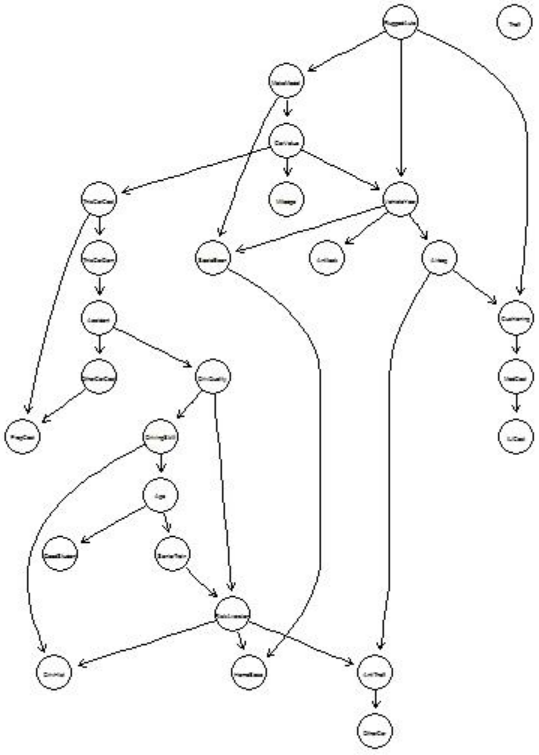
IAMB algorithm

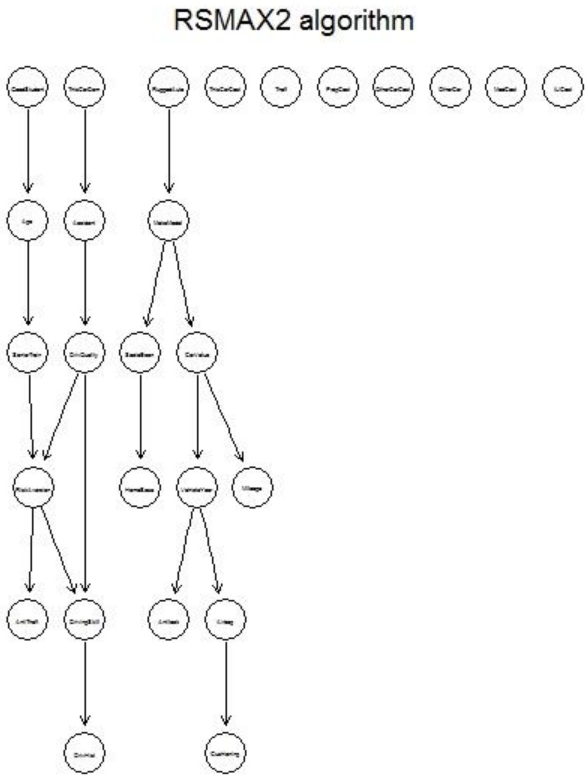


IIAMB algorithm



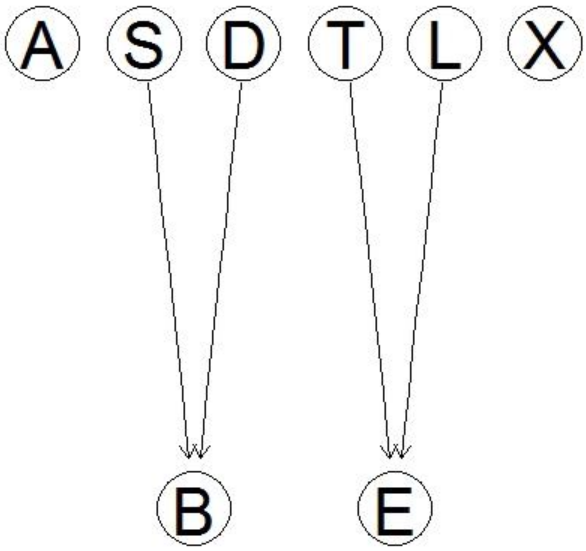
MMHC algorithm



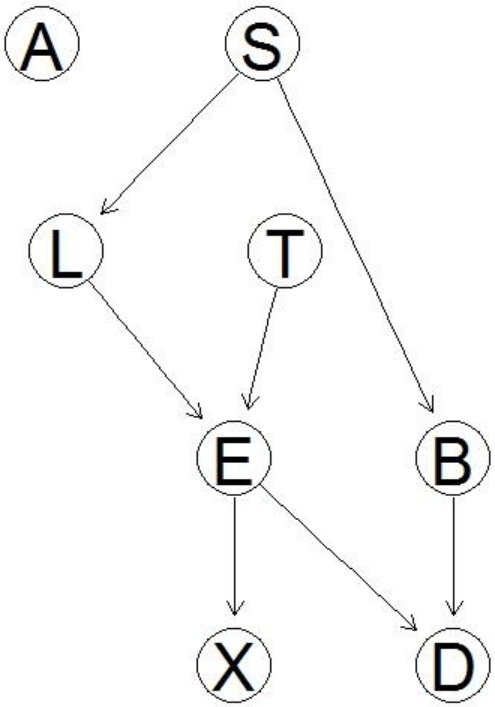


e) Models generated for "ASIA" dataset

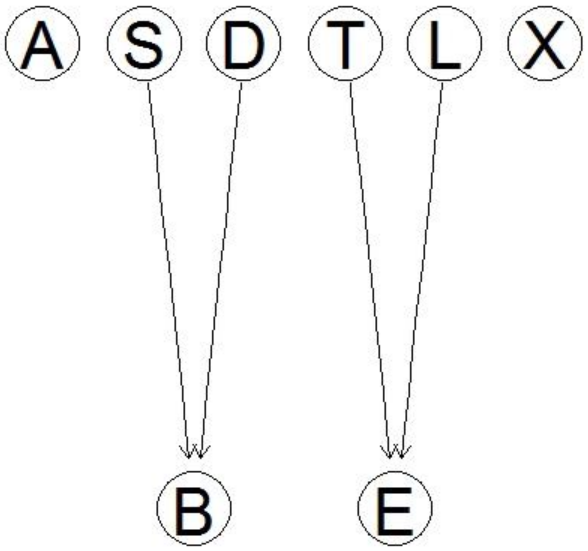
FIAMB algorithm



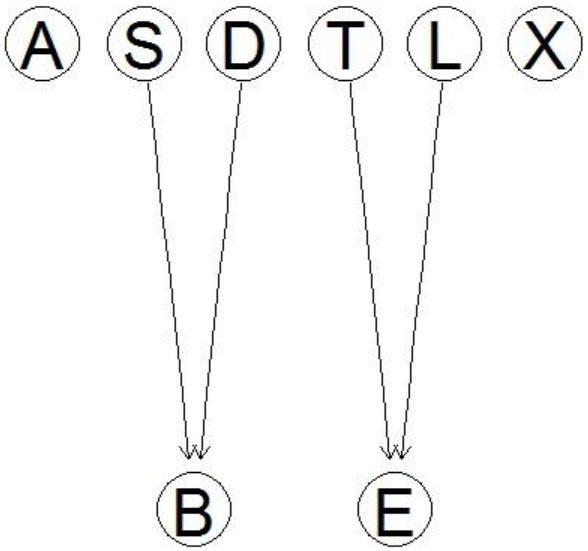
Hill-Climbing algorithm



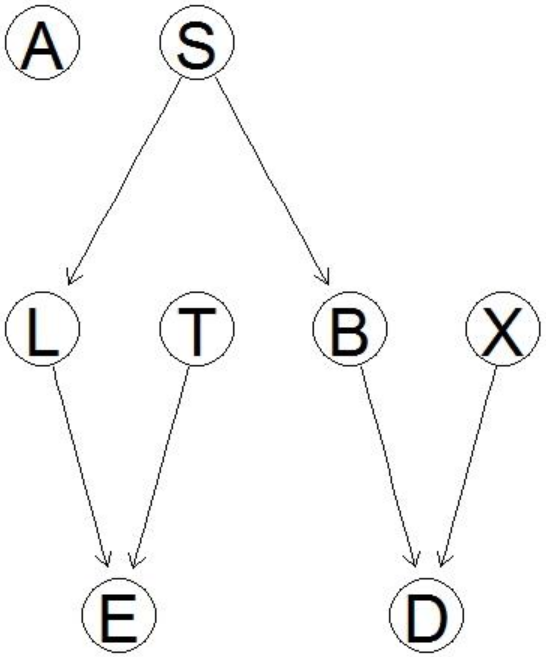
IAMB algorithm



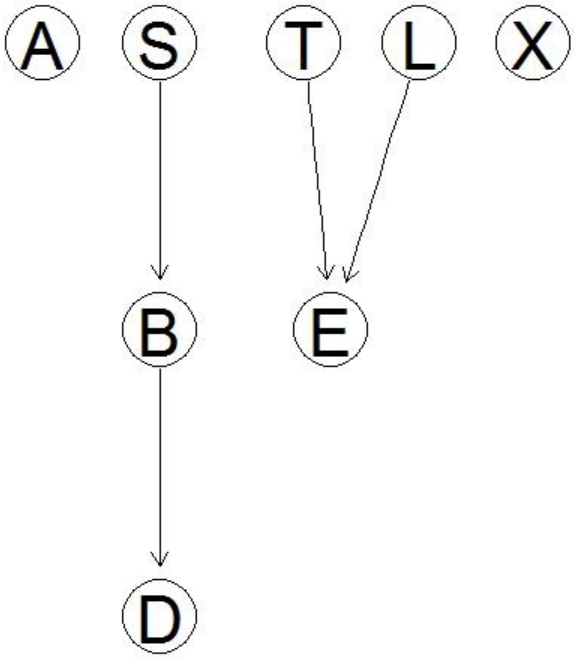
IIAMB algorithm

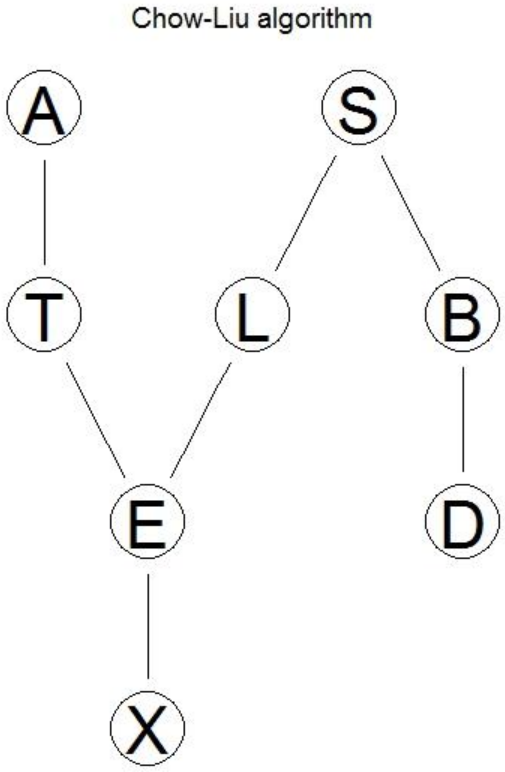


MMHC algorithm



RSMAX2 algorithm




Chow-Liu algorithm
A
S
T
L
B
E
D
X