

AN EASY WAY TO LEARN
DISCRETE BAYESIAN NETWORK
FROM DATA

Enrico Papalini

papalini@biancaneve.ing.UniFI.IT

Michele Piccini

mpiccini@biancaneve.ing.UniFI.IT

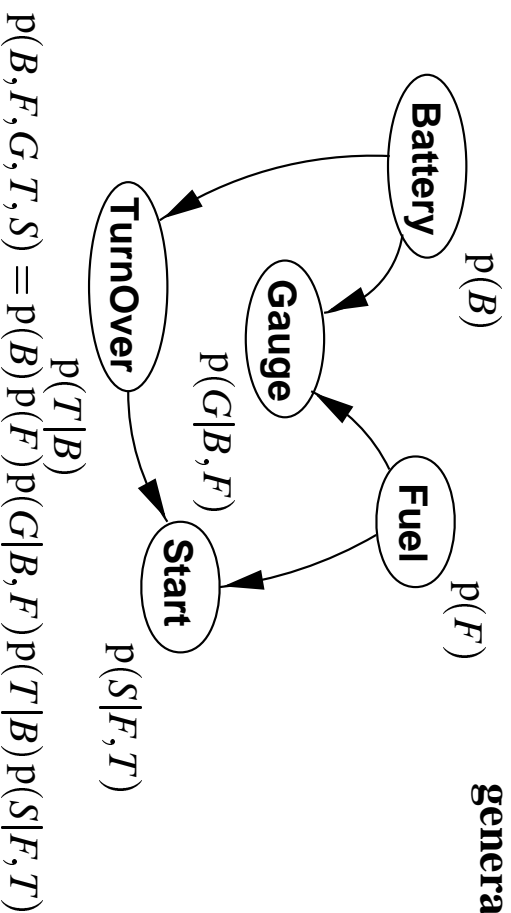
Rome, 18 Sept. 1997

INTRODUCTION TO BAYESIAN NETWORKS

definition: a random variable X is *independent* of Y given context C if $p(X, Y|C) = p(X|C)p(Y|C)$ whenever $p(C) \neq 0, \forall X, Y, C$.

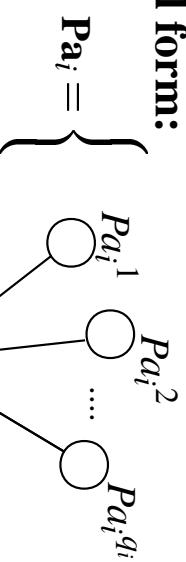
definition: a *BN* is a direct acyclic graph (DAG) that represents probabilities and independencies over a set \mathbf{X} of random variables X_i . The *DAG structure* defines the conditional independence relations. The joint probability can be factorized in marginal probabilities defined locally on nodes as set of *parameters*.

example:



$$p(B, F, G, T, S) = p(B)p(F)p(G|B, F)p(T|B)p(S|F, T)$$

general form:



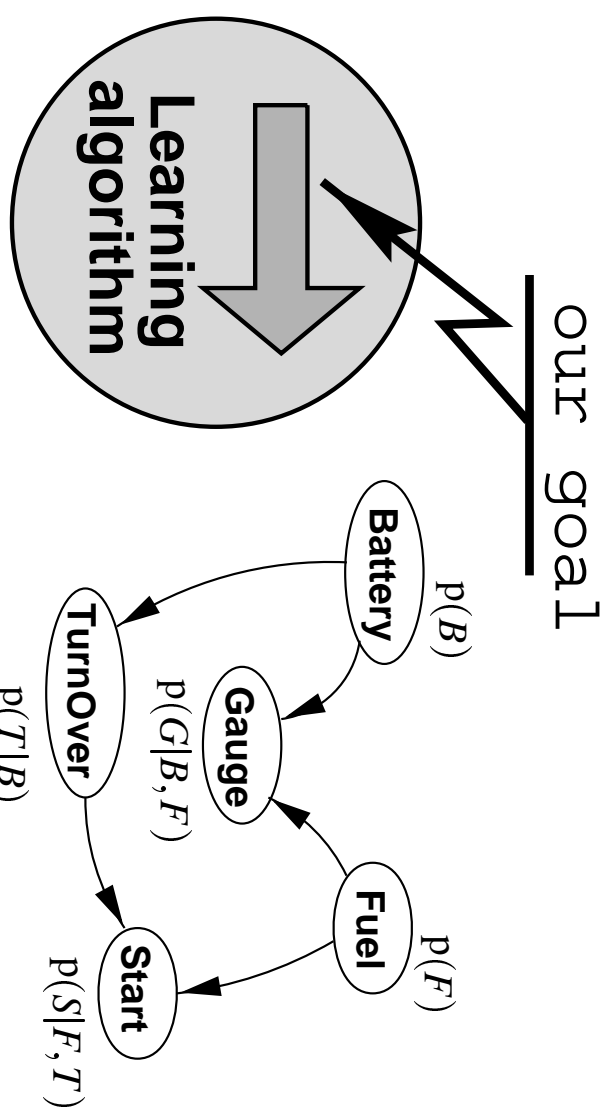
$$p(\mathbf{X}) = \prod_{i=1}^n p(X_i | \mathbf{Pa}_i)$$

OUR GOAL IS ...

our goal

prior knowledge
+
database

B	F	G	T	S
ok	full	on	ok	yes
ok	half	on	down	no
ok	half	on	ok	yes
ok	empty	off	ok	no
ok	full	on	down	no
...



We want to illustrate an easy way to learn a BN over a set of variables \mathbf{X} given a database $D = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$, where \mathbf{x}_i is instance of X_i .



We want to estimate the probability distribution on \mathbf{X} given a random sample $D = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ from the true distribution.

LEARNING WITH BAYESIAN MODEL SELECTION APPROACH

definition: \mathbf{M} is a discrete random variable (drv) whose states correspond to the possible true models. Our uncertainty about \mathbf{M} is encoded with $p(\mathbf{M} = \mathbf{m})$.

definition: \mathbf{T} is a drv whose values correspond to the possible values of the parameters of \mathbf{m} . Our uncertainty is encoded using $p(\mathbf{T} = \theta | \mathbf{m})$.

Given a prior distribution for \mathbf{X} , $p(\mathbf{x} | \theta, \mathbf{m})$, and a random sample D , we compute the posterior distribution for \mathbf{X} by averaging over all models and parameters (using posterior distribution for \mathbf{M} and \mathbf{T} estimated on D with Bayes' rule).

$$p(\mathbf{x} | D) = \sum_{\mathbf{m}} p(\mathbf{m} | D) \int p(\mathbf{x} | \theta, \mathbf{m}) p(\theta | D, \mathbf{m}) d\theta$$

This *Bayesian model averaging* approach is intractable. A good approximation is to select the “best” model $\hat{\mathbf{m}}$, and to average over its parameters.

$$p(\mathbf{x} | D, \hat{\mathbf{m}}) = \int p(\mathbf{x} | \theta, \hat{\mathbf{m}}) p(\theta | D, \hat{\mathbf{m}}) d\theta$$

This approach is known as *Bayesian model selection*.

WHICH IS THE “BEST” MODEL?

The “best” model should be the most probable given the database D . It has to maximize its posterior probability $P(\mathbf{m}|D)$.

Assuming uniform prior on models, we can define a *model score function* as

$$\text{SCORE}(\mathbf{m}) = \log P(\mathbf{m}|D) \simeq \log P(D|\mathbf{m})$$

The problem of learning a BN structure from data is equivalent to the problem of maximizing $\text{SCORE}(\mathbf{m})$ function over the space of all possible network structures, which are more than exponential in the number of nodes.

This is an \mathcal{NP} -hard problem!

Therefore, it is necessary to use heuristic search methods for model selection.

SIMPLIFYING ASSUMPTIONS

1. Every variable in \mathbf{X} is discrete \Rightarrow marginal distributions are *multinomial*:

$$P(X_i = x_i^k | \mathbf{Pa}_i = \mathbf{pa}_i^j, \theta_i, \mathbf{m}) = \theta_{ijk}$$

2. Each parameter set θ_{ij} has a *Dirichlet distribution*:

$$P(\theta_{ij} | \mathbf{m}) = \text{Dir}(\theta_{ij}, \alpha_{ij1}, \dots, \alpha_{ijr_i})$$

3. The parameters θ_{ij} are *mutually independent* \Rightarrow the problem is separable on X_i (prior distributions of parameters can be calculated locally).

4. Data D are *complete* (no missing observations, no hidden variables).

Under these assumptions, Cooper and Herskovits (1992) have shown an analytical form for *model likelihood* $P(D | \mathbf{m})$ and one for *posterior probability* $P(\mathbf{x} | \mathbf{pa}_i, D, \mathbf{m})$.

AN ALGORITHM FOR LEARNING BN

- *Initialization*
- Dirichlet parameter priors $\alpha_{ijk} = \frac{\alpha}{q_i r_i}$ where α is said *equivalent sample size*, $r_i = \#$ states of X_i , $q_i = \#$ states of parents of X_i .
- *Model Selection*

Heuristic search of DAG $\hat{\mathbf{m}}$ that maximizes the following SCORE function

$$\text{SCORE}(\mathbf{m}) = \sum_{i=1}^n \text{SCORE}_i(\mathbf{m})$$

where, for separability, the score function for a single node X_i is

$$\text{SCORE}_i(\mathbf{m}) = \sum_{j=1}^{q_i} \log \frac{\Gamma(\sum_{k=1}^{r_i} \alpha_{ijk})}{\Gamma(\sum_{k=1}^{r_i} \alpha_{ijk} + \sum_{k=1}^{r_i} N_{ijk})} + \sum_{k=1}^{r_i} \log \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})}$$

$N_{ijk} = \#$ cases where X_i is in the state x_i^k and its parents \mathbf{Pa}_i are in the state \mathbf{pa}_i^j .

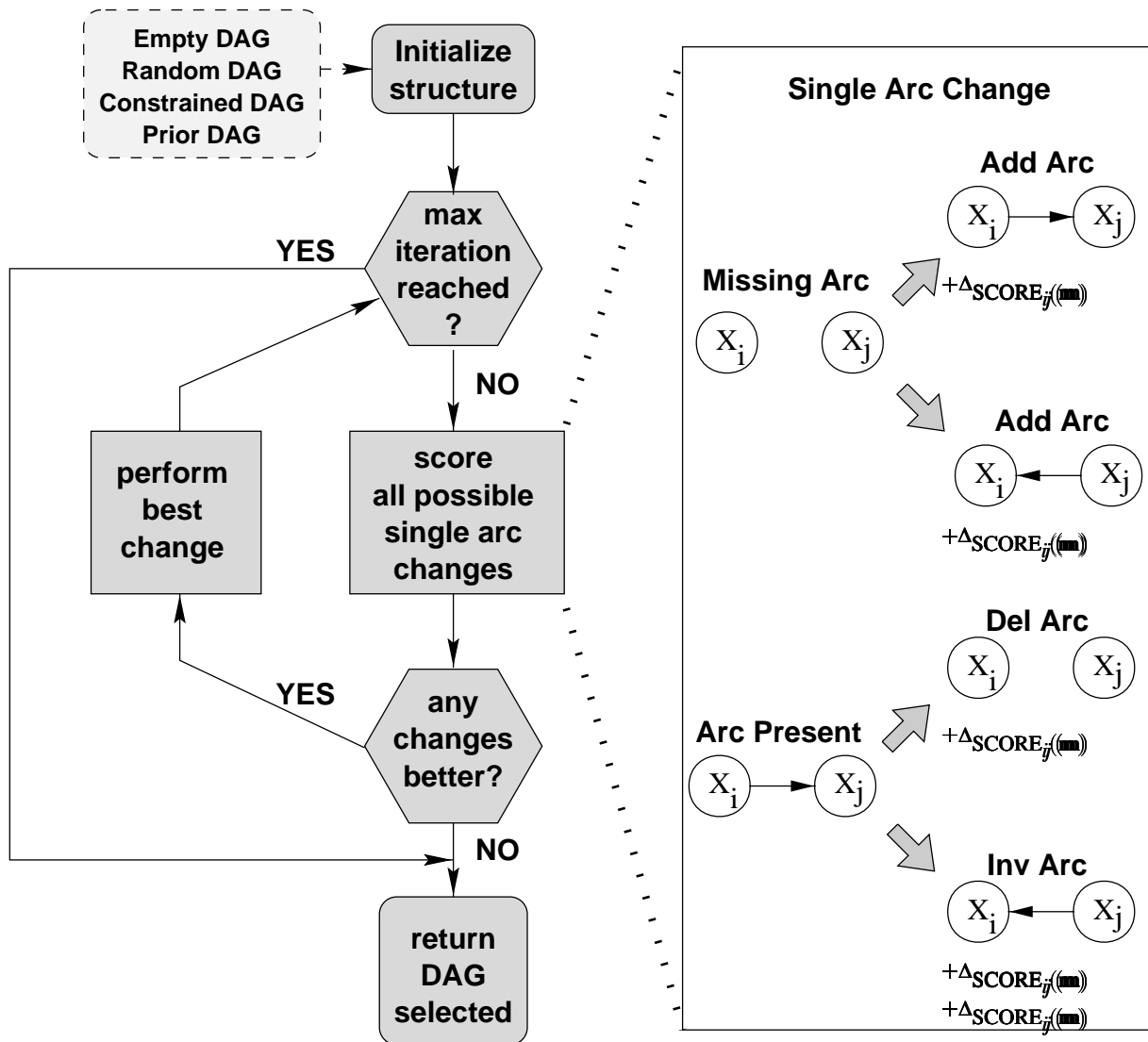
- *Parameters Evaluations*

Marginal probabilities calculation using the following formula

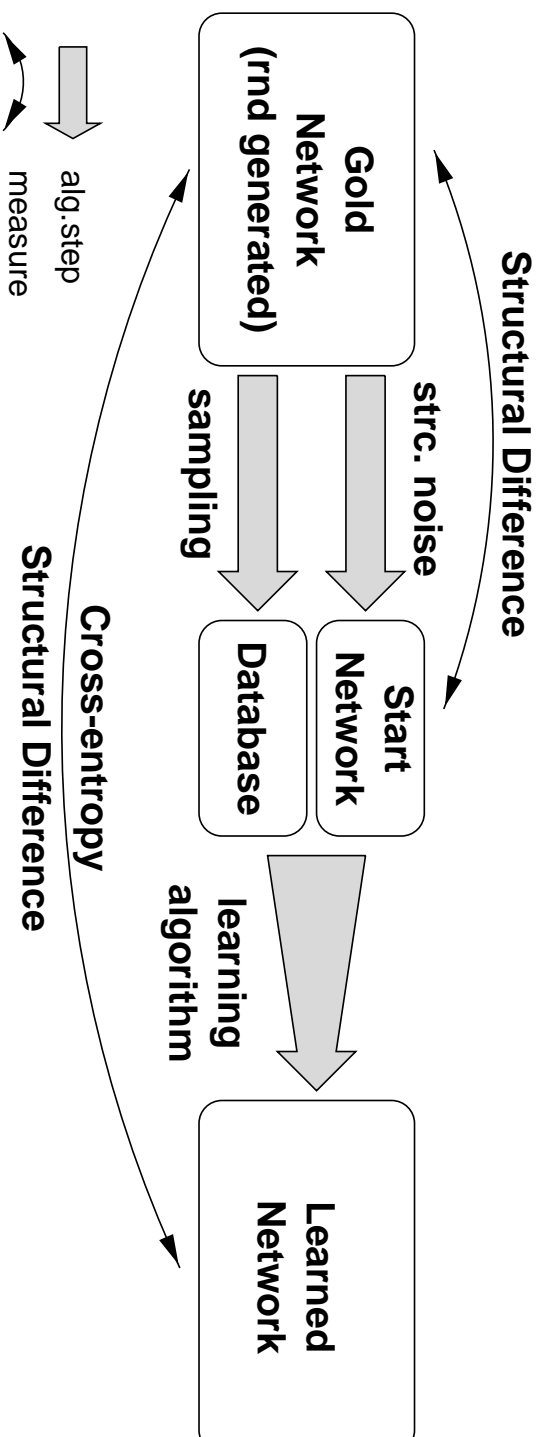
$$P(X_i = x_i^k | \mathbf{Pa}_i = \mathbf{pa}_i^j, D, \hat{\mathbf{m}}) = \frac{\alpha_{ijk} + N_{ijk}}{\alpha_{ij} + N_{ij}}$$

HEURISTIC SEARCH IN DAG SPACE

DAG space is vast and unknown. *Hill Climbing search* algorithm is good choice in order to maximize $\text{SCORE}(\mathbf{m})$ in DAG space. It makes successive arc changes to the network and employs the property of decomposability to evaluate the merit of each change.



EVALUATION METHODOLOGY

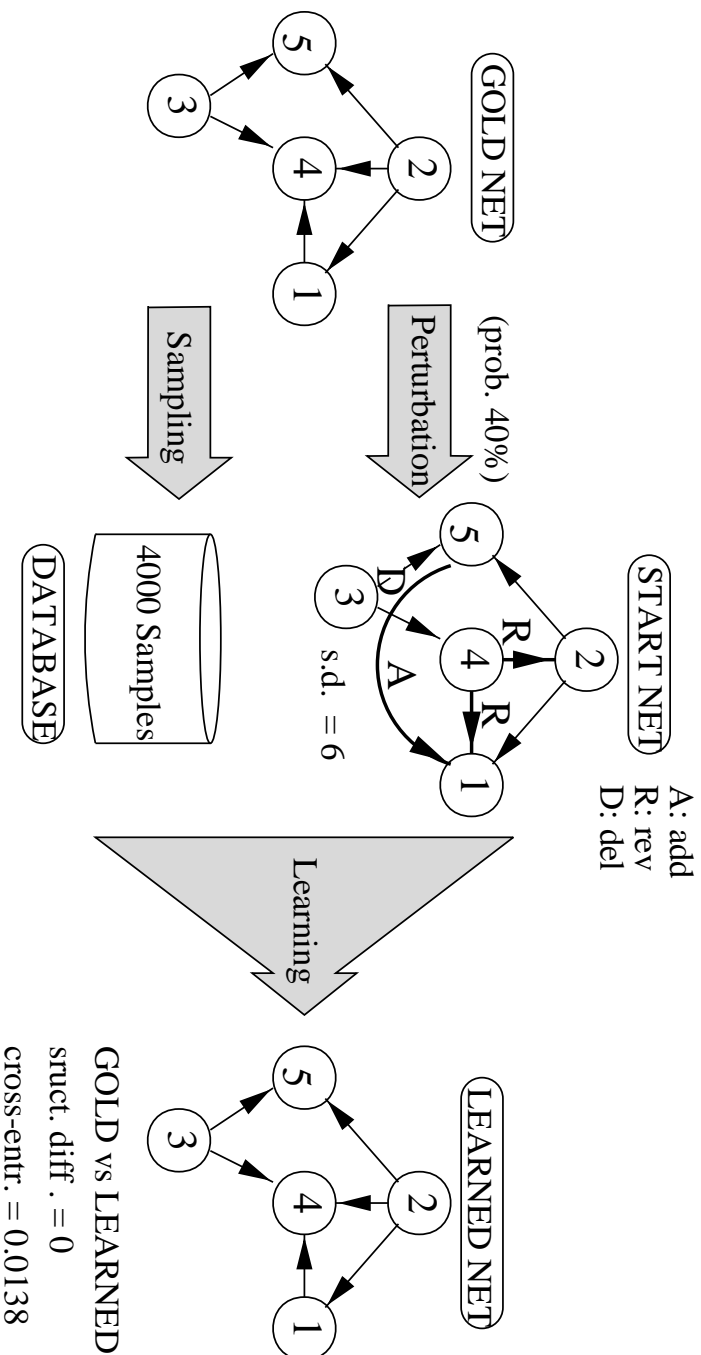


Structural Difference: number of different arcs between two BNs. This is a measure of the degree to which the learned structure have captured causal relationships. A null structural difference means equal DAGs.

Cross-Entropy: reflects how well the learned structure will predict the next case. It has derived from the information theory by Kullback and Leibler (1951). Low values correspond to a learned distribution close to the gold one.

RESULTS FROM SIMULATED BNS

example 1: a 5 variables BN, max 4 states per variable.

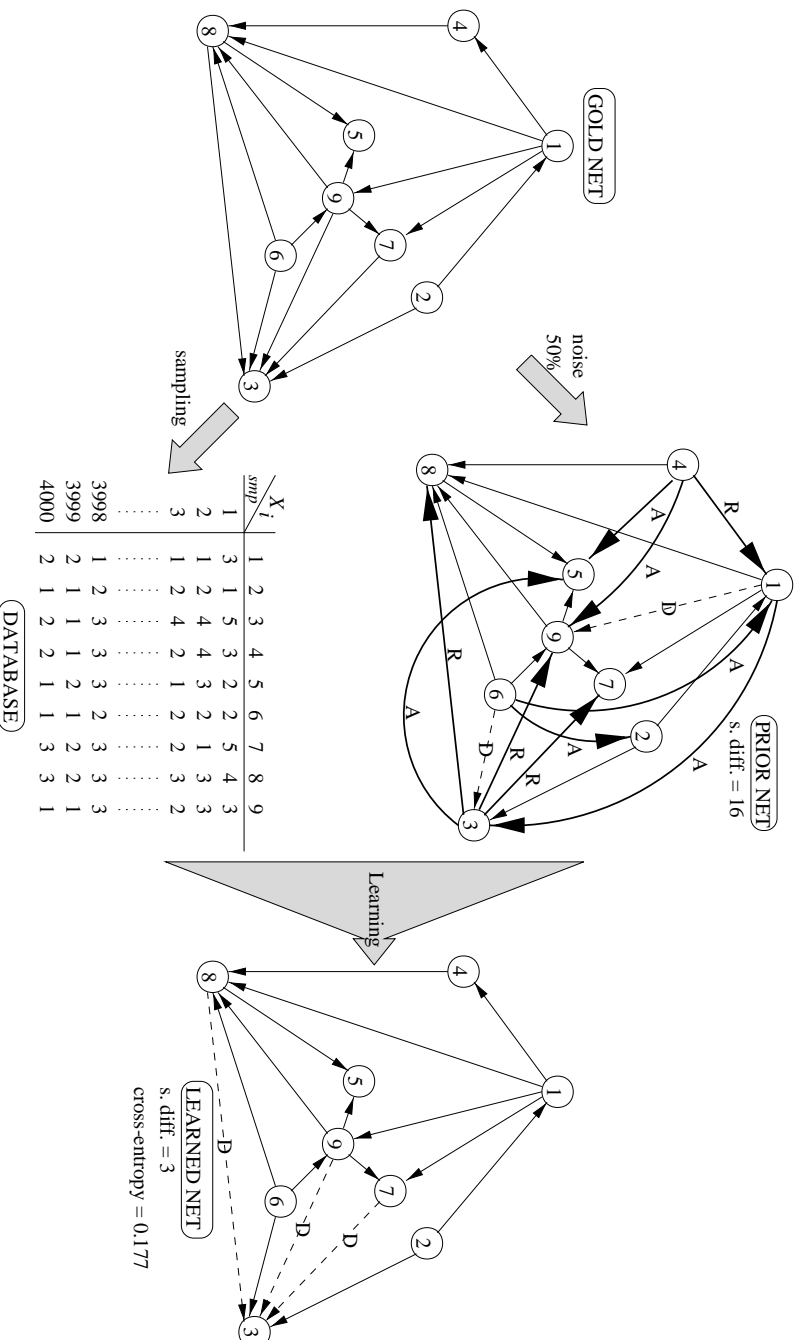


hill-climbing steps:

- 1^ search iter: current score = -26777.283203 + 310.848145 (add: 3 -> 5)
- 2^ search iter: current score = -26466.435059 + 142.760742 (rev: 2 -> 4)
- 3^ search iter: current score = -26323.674316 + 179.492676 (rev: 1 -> 4)
- 4^ search iter: current score = -26144.181641 + 24.003906 (del: 5 -X 1)
- 5^ search iter: current score = -26120.177734 No better changes.

RESULTS FROM SIMULATED BNS

example 2: a 9 variables BN, max 5 states per variable.



hill-climbing steps:

- 1^ iter: score = -40772.177979 + 869.140137 (del: 3 -x 8)
- (...)
- 13^ iter: score = -38652.027832 No better changes.

A CASE STUDY: COLLEGE PLANS

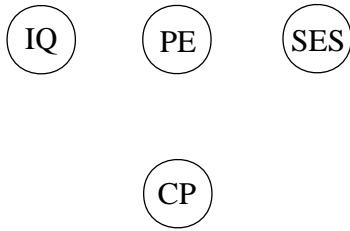
We want to investigate factors that influence the intention of high school students to attend college. Sewell and Shah (1968) measured the following variables for 10416 Wisconsin high school seniors: Sex (SEX), Socioeconomic Status (SES), Intelligence Quotient (IQ), Parental Encouragement (PE), College Plans (CP).

IQ	low		high		lower middle		high		upper middle		low		high		high		S
	low	high	low	high	low	high	low	high	low	high	low	high	low	high	low	high	
low	4	349	13	64	9	207	33	72	12	126	38	54	10	67	49	43	m
low mid	2	232	27	84	7	201	64	95	12	115	93	92	17	79	119	59	a
up mid	8	166	47	91	6	120	74	110	17	92	148	100	6	42	198	73	1
high	4	48	39	57	5	47	123	90	9	41	224	65	8	17	414	54	e
low	5	454	9	44	5	312	14	47	8	216	20	35	13	96	28	24	f
low mid	11	285	29	61	19	236	47	88	12	164	62	85	15	113	72	50	e
up mid	7	163	36	72	13	193	75	90	12	174	91	100	20	81	142	77	m.
high	6	50	36	58	5	70	110	76	12	48	230	81	13	49	360	98	
SES																	

Sufficient statistics for the Sewell and Shah study.

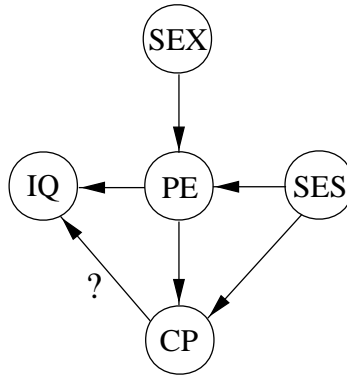
COLLEGE PLANS: OUR RESULTS

A. (SEX)



empty prior network

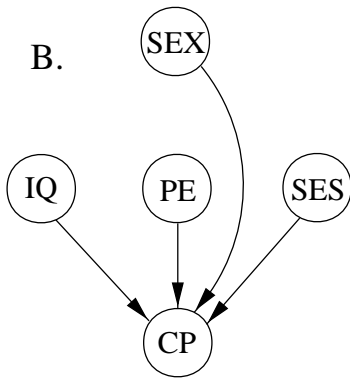
learning
 $\alpha = 8$



learned network
score = -45579

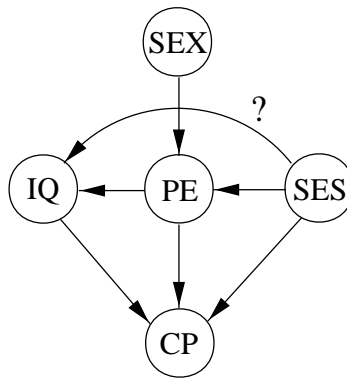
SES	PE	CP=yes
low	low	0.035
high	low	0.144
low	high	0.347
high	high	0.726

B.



prior network

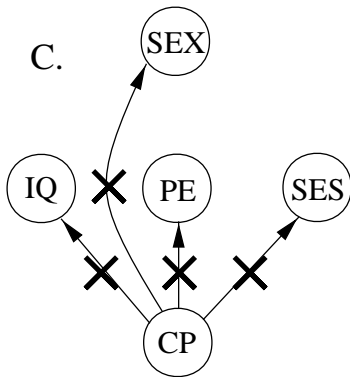
learning
 $\alpha = 256$



learned network
score = -45617

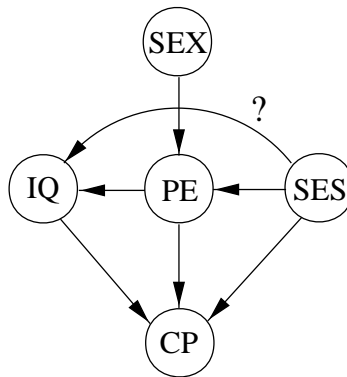
SES	IQ	PE	CP=yes
low	low	low	0.015
high	low	low	0.120
low	high	low	0.139
high	high	low	0.263
low	low	high	0.188
high	low	high	0.398
low	high	high	0.532
high	high	high	0.832

C.



empty prior network
with constraints

learning
 $\alpha = 8$



learned network
score = -45628

SES	IQ	PE	CP=yes
low	low	low	0.011
high	low	low	0.093
low	high	low	0.124
high	high	low	0.242
low	low	high	0.169
high	low	high	0.394
low	high	high	0.534
high	high	high	0.835

?: suspicious results.

SUMMARY

- We have presented a way to learn discrete Bayesian Networks from data. This is based on *Bayesian Model Selection* approach, where a “good” model is chosen using hill-climbing heuristic search on DAGs space.
- We have shown some results from simulated networks and a true case study. The algorithm has always increased our prior knowledge on data.
- This methodology can be used in early stages of resolution of a *Data Mining* problem. It picks out an independence model for variables and explains relations among variables in a causal way.
- We need to test more structures and data sets in order to investigate how to scale-up the problem size (may it be a future work ?).