

The max-min-hill-climbing algorithm

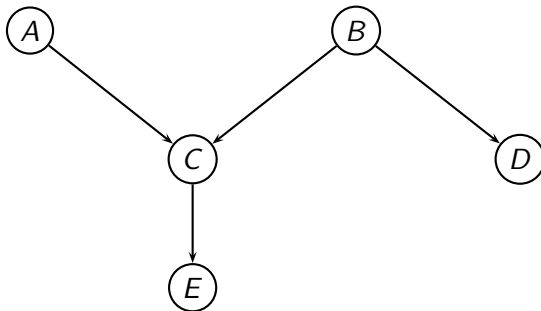
Michael Bauer

M.Sc. Comp. Science

July 15, 2014

Definition

A Bayesian Network is a directed acyclic graph (DAG) whose nodes are random variables and edges represent conditional dependencies. If two random variables are connected they are said to be dependent. If there is no connection they are said to be conditional independent.



- directed edges
- free of cycles
- random variable is represented as a node
- edges encode dependencies
- For instance: parent A, child C

Predicting the effect of missense mutations on protein function: analysis with Bayesian networks

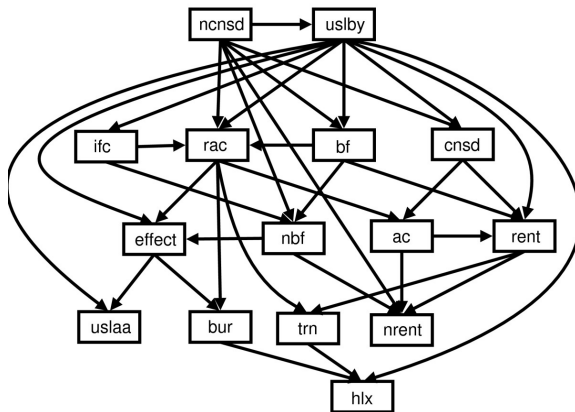
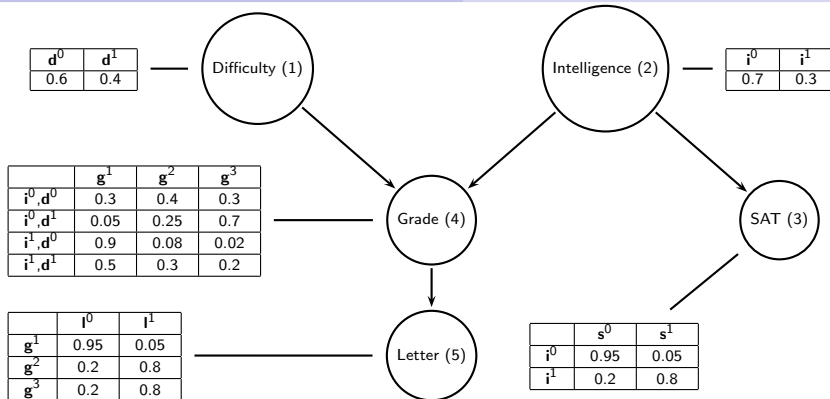


Figure: <http://www.biomedcentral.com/1471-2105/7/405/figure/F2?highres=y>
 (by Chris J Needham1, James R Bradford, Andrew J Bulpitt, Matthew A Care and David R Westhead)

Bayesian Networks in sports and medicine



Figure: http://www-ekp.physik.uni-karlsruhe.de/~zupanc/WS1011/docs/Datenanalyse2010_3.pdf



```
> head(dataFrame)
      diff int SAT gra let
[1,]    1  2  2   1  2
[2,]    1  1  1   1  2
[3,]    1  1  1   3  1
[4,]    2  2  2   2  2
[5,]    1  1  1   1  2
[6,]    2  1  1   3  1
```

Figure: The data we observe from following the rules above.

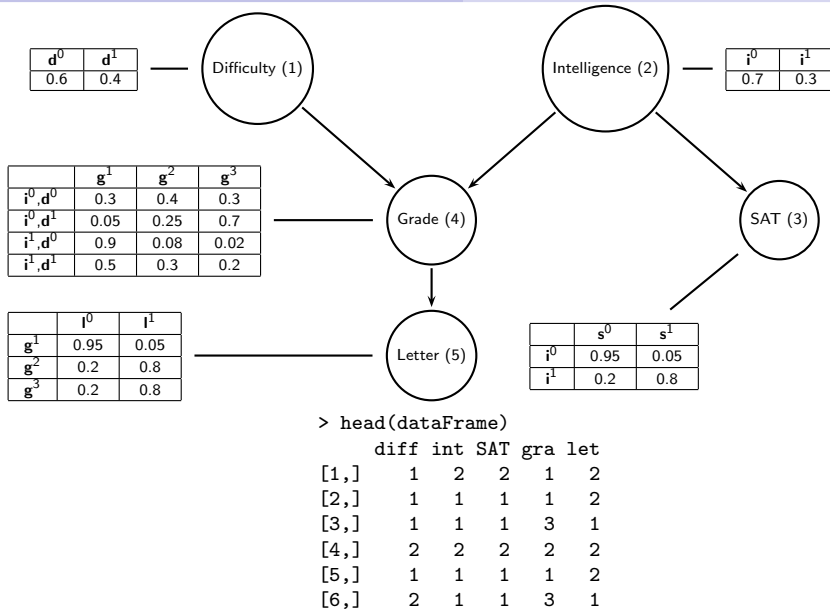


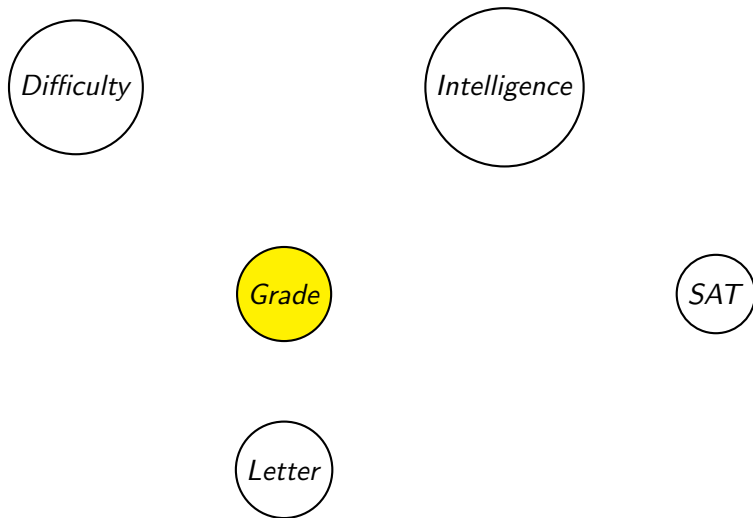
Figure: The data we observe from following the rules above.

Empty graph without any edges

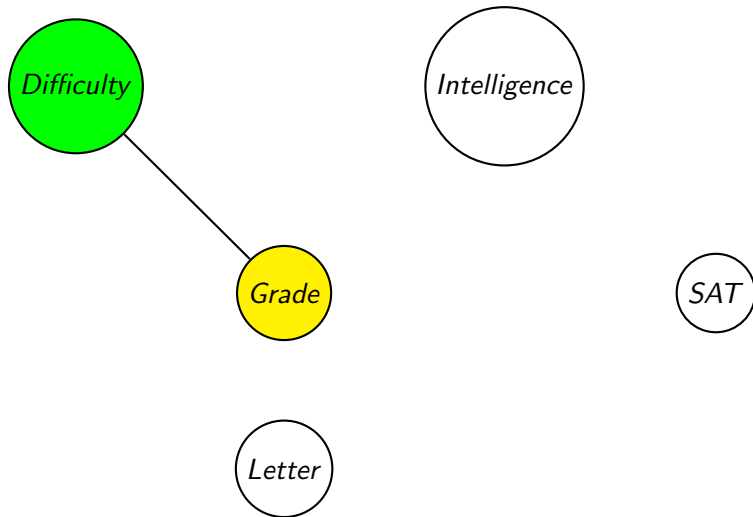


```
> head(dataFrame)
      diff int SAT gra let
[1,]     1   2   2   1   2
[2,]     1   1   1   1   2
[3,]     1   1   1   3   1
[4,]     2   2   2   2   2
[5,]     1   1   1   1   2
[6,]     2   1   1   3   1
```

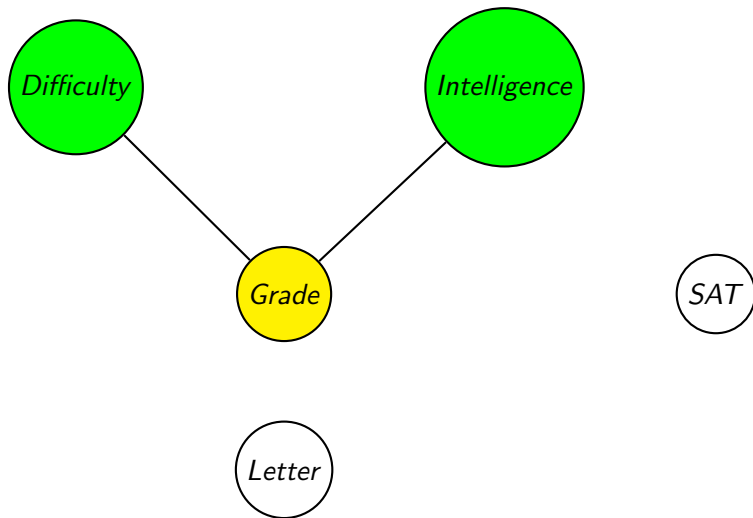

One iteration for the "Grade" node



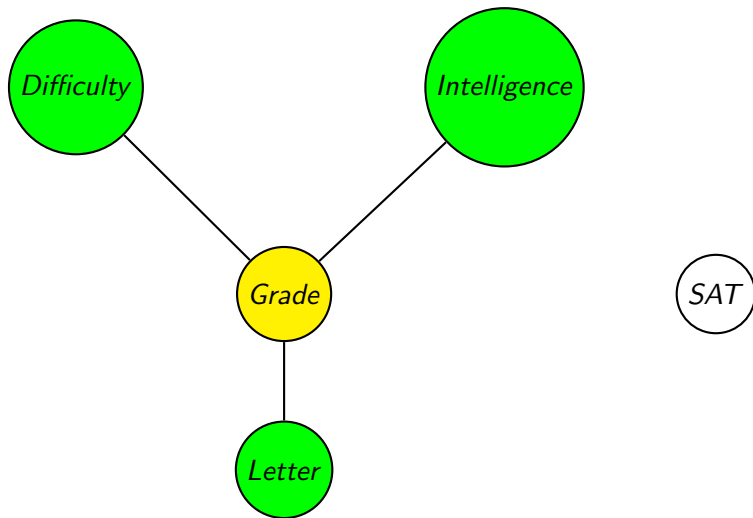
One iteration for the "Grade" node



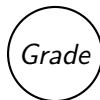
One iteration for the "Grade" node

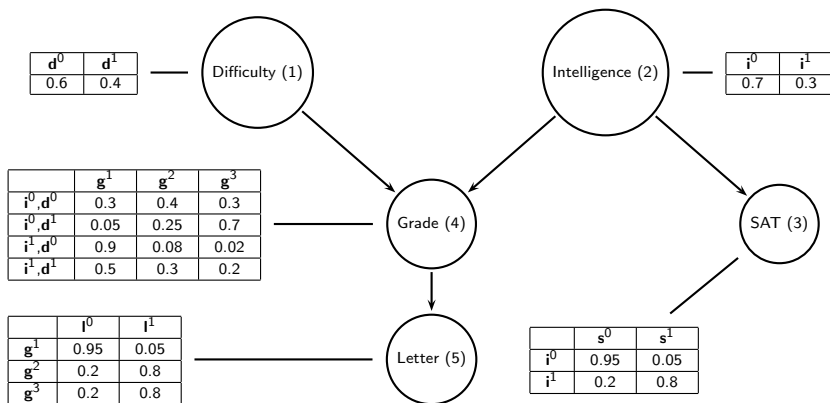


All parents or children are found



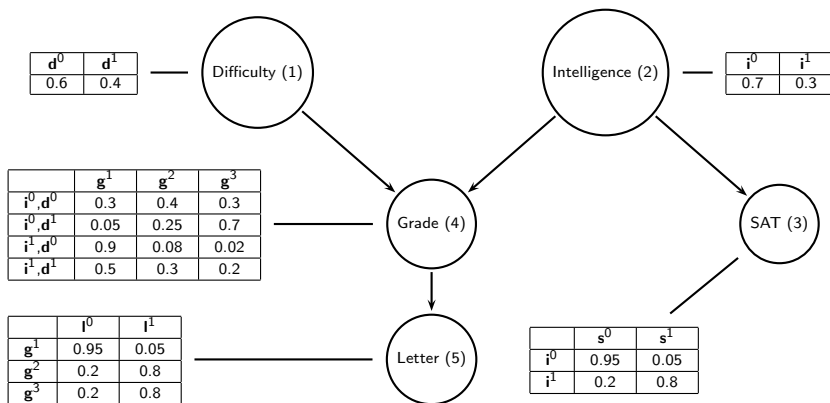
Start new iteration





```
> C_MMPC(dataSet)
```

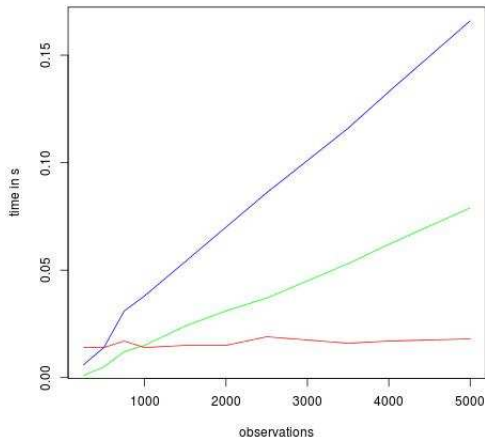
```
[[Difficulty]]  [[Intelligence]]  [[SAT]]  [[Grade]]  [[Letter]]
[1] 4           [1] 3 4          [1] 2    [1] 5 2 1  [1] 4
```



```
> C_MMPC(dataSet)
```

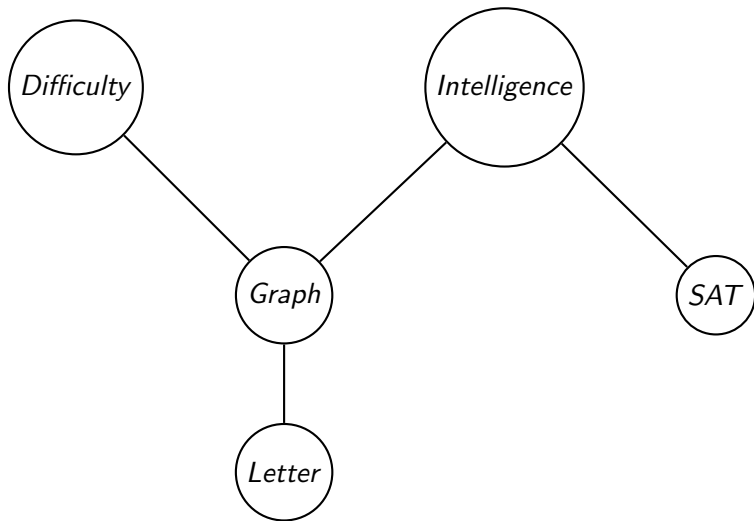
```
[[Difficulty]]  [[Intelligence]]  [[SAT]]  [[Grade]]  [[Letter]]
[1] 4           [1] 3 4          [1] 2    [1] 5 2 1  [1] 4
```

The benchmark for this algorithm



nobs	R	C	bnlearn
250	0.006	0.001	0.014
500	0.014	0.005	0.014
750	0.031	0.012	0.017
1000	0.038	0.015	0.014
1500	0.054	0.024	0.015
2500	0.086	0.037	0.019
5000	0.166	0.079	0.018

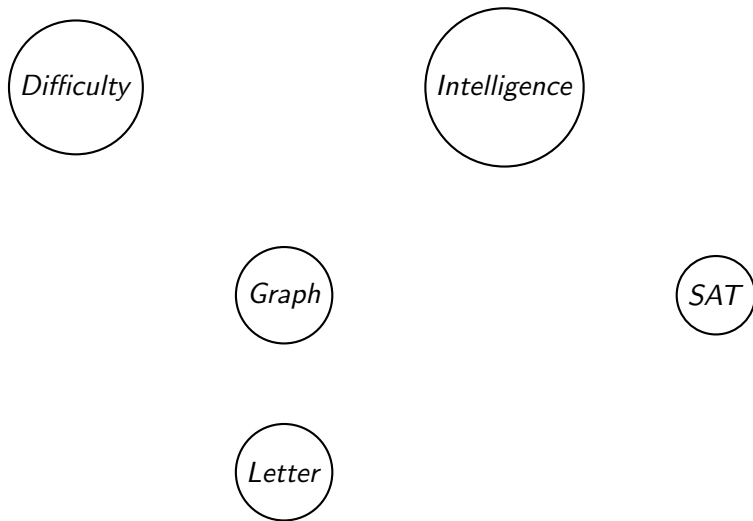
The arrows are still missing



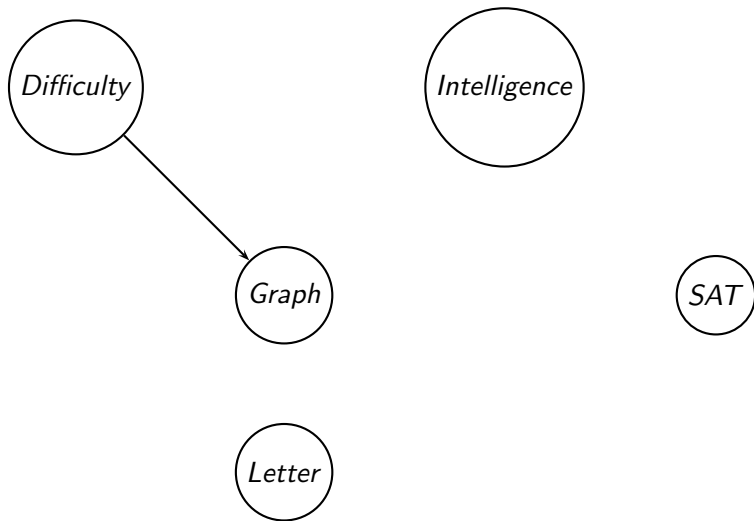
Bayesian Dirichlet equivalent uniform (BDeu) score

$$BDeu(G) = \sum_{i=1}^n \sum_{j=1}^{q_i} \left(\log \left(\frac{\Gamma(\frac{\eta}{q_i})}{\Gamma(N_{ij} + \frac{\eta}{q_i})} \right) + \sum_{k=1}^{r_i} \log \left(\frac{\Gamma(N_{ijk} + \frac{\eta}{r_i q_i})}{\Gamma(\frac{\eta}{r_i q_i})} \right) \right).$$

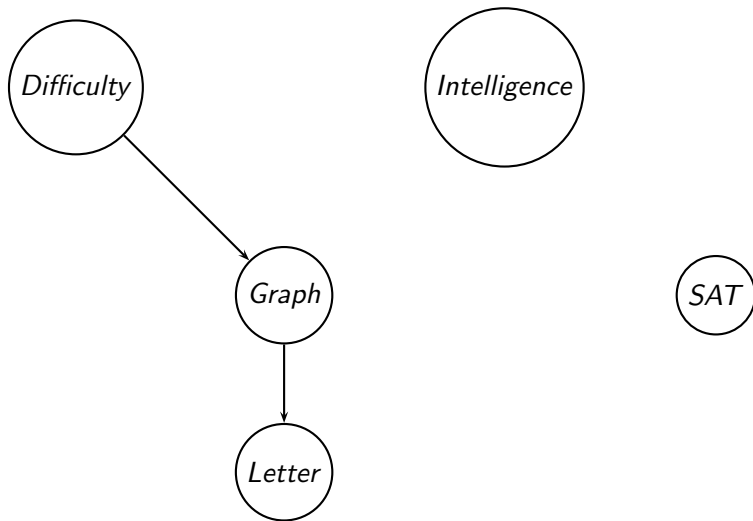
Bayesian Dirichlet equivalent uniform (BDeu) score



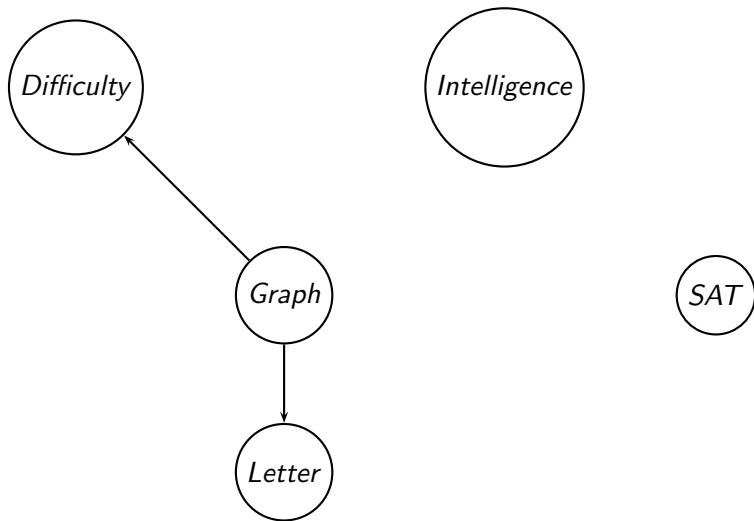
Adding an edge



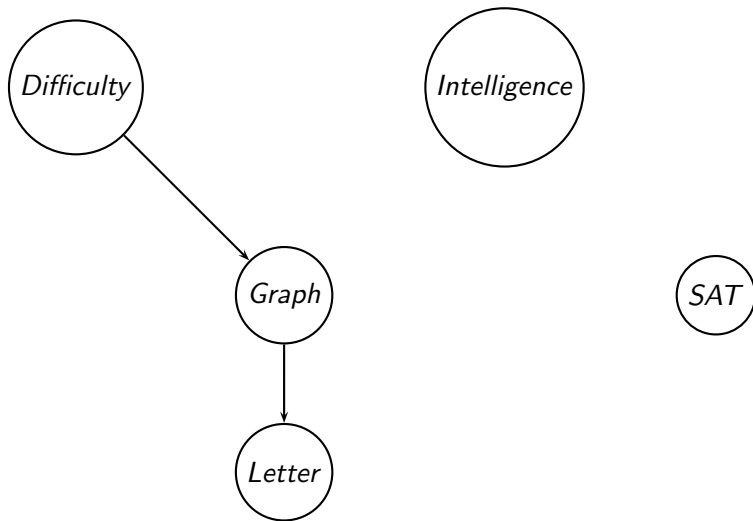
Adding an edge



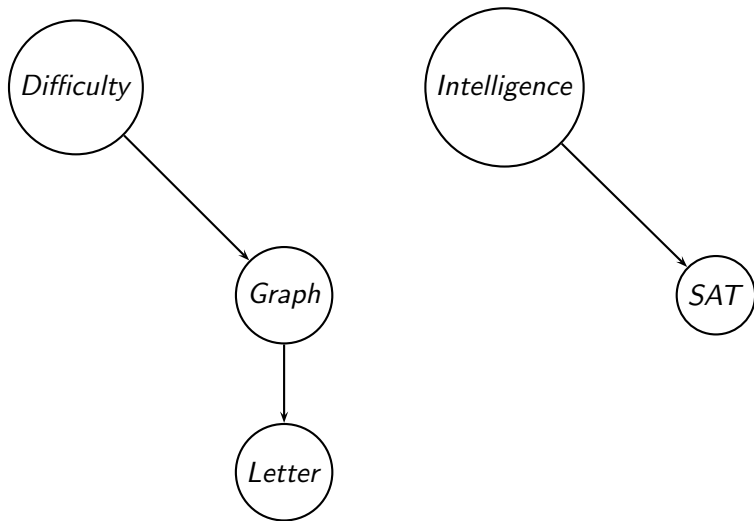
Also possible: reverse and delete



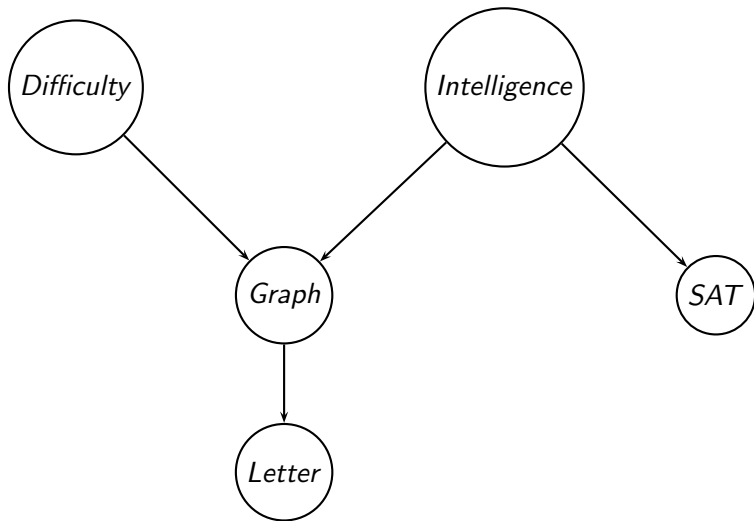
Reverse again



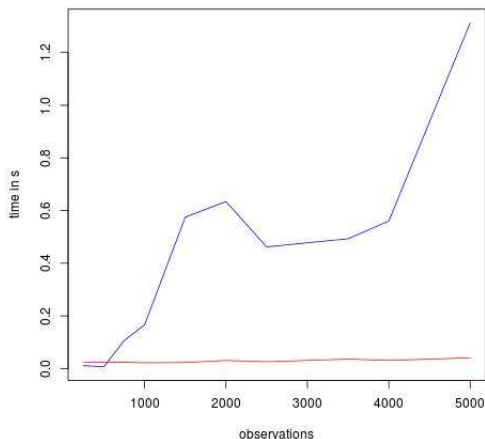
Adding an edge



Adding an edge

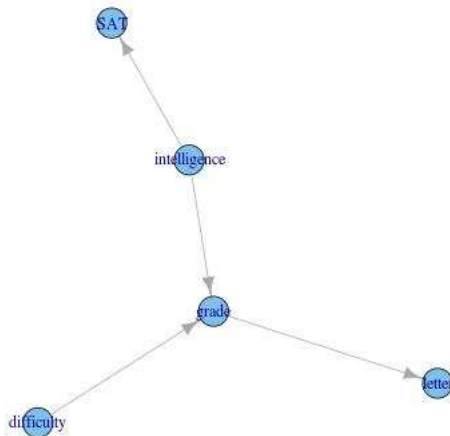


Benchmark for the whole program



nobs	C	bnlearn
250	0.011	0.023
500	0.007	0.024
750	0.107	0.024
1000	0.166	0.022
1500	0.575	0.023
2500	0.493	0.036
5000	1.313	0.041

Output of my program



Thanks for your attention!