Student Performance Bayesian Networks

Gio Formichella

Introduction

- Modelled student features and math performance from the UCI Student Performance dataset.
- Three different approaches:
  - **PC algorithm** (*Spirtes et al., 2000*, *Spirtes et al., 1991* and *Glymour et al., 2019*)
  - **Tree Search** (*Chow and Liu, 1968* and *Friedman et al., 1997*)
  - **Expert-defined**
- pgmpy (*Ankan and Textor, 2024*)
- Measured memory usage and execution time of exact inference queries using variable elimination on the three Bayesian networks.

PC algorithm

1. Form a complete undirected graph, nodes correspond to variables.
2. Remove edges based on conditional independence: iteratively test whether each pair of connected variables X and Y is conditionally independent of a subset S of other variables. If $X \perp Y|S$, remove the edge between X and Y. S starts as $\emptyset$ and iteratively increases in size.
   - Chi-squared test: if p-value $> \alpha = 0.05$ (α significance level), conclude $X \perp Y|S$.
3. Orient V-structure: For each $X - Z - Y$ variables, where X and Y are not directly connected, orient to $X \rightarrow Z \leftarrow Y$.
4. Orientation propagation: For each $X \rightarrow Y - Z$, and X and Z are not adjacent, orient the edge $Y - Z$ as $Y \rightarrow Z$.

PC algorithm

- Constrained-based structure learning algorithm
- Assumption of no hidden confounders

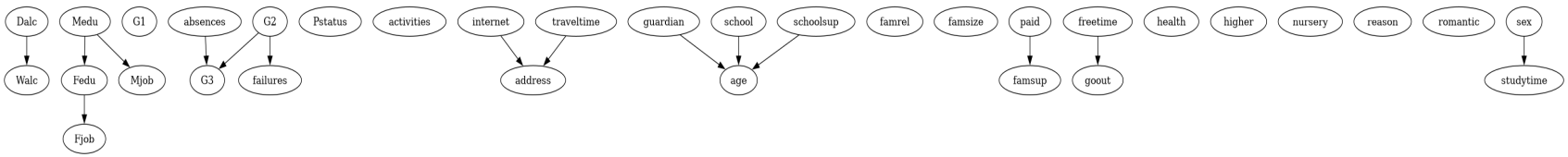


Figure: Student performance PC network

Tree Search

1. Compute the *mutual information* function between each pair of variables X and Y:
   $I_{\hat{P}_D}(X, Y) = \sum_{x,y} P(x, y) \log(\frac{P(x,y)}{P(x)P(y)})$
2. Build a complete undirected graph: nodes corresponding to variables and weights to mutual information, i.e.
   $w(i, j) = I_{\hat{P}_D}(X_i; X_j)$
3. Build a maximum weighted spanning tree (e.g. Kruskal's algorithm)
4. Transform the undirected tree to a direct one by choosing a root variable and setting the direction of all edges to be outward from it.

Tree Search

- Score-based structure learning algorithm.
- Tree-like BN: each variable, except the root, has 1 parent node.

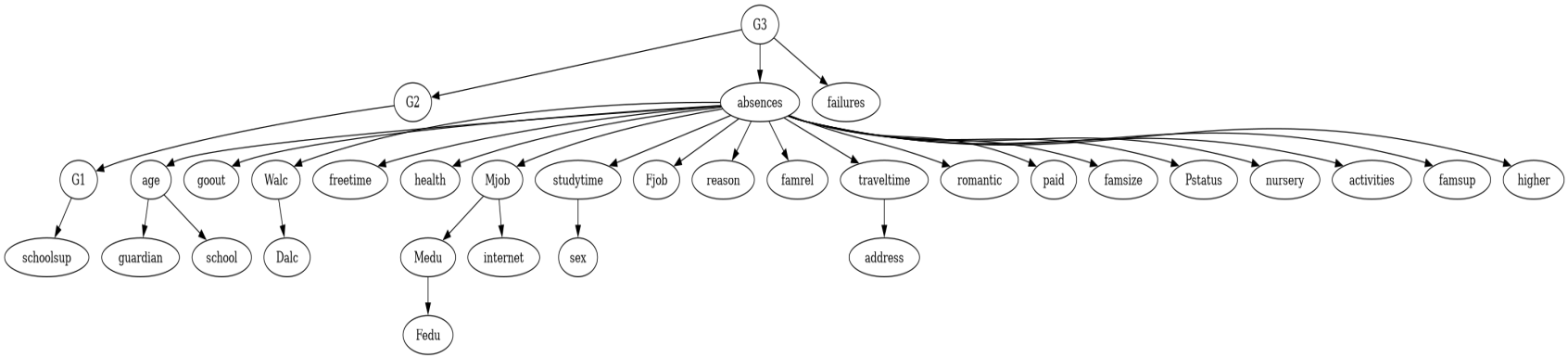


Figure: Student performance Tree network

Expert-defined

- Use of domain knowledge and prior understanding of factors influencing student performance.
- Dropped features deemed less influential on grades:
  - **school, reason**: performance proved equivalent across different schools.
  - **address**: address type not considered relevant.
  - **famsize, guardian**: family information of other features considered as more informative.
  - **nursery**: attending nursery school shown to be not relevant.
  - **G1, G2**: highly correlated with the final grade, G3. Assuming that the same causes that determine G3 also determine G1 and G2.

Expert-defined

- Feature aggregation: Created **socially_active** boolean feature from **romantic** and **goot**.
- Feature re-binning:
  - **Medu, Fedu, Mjob, Fjob**: to binary
  - **absences**: to 0, 1 and 2 corresponding to "Low", "Medium" and "High".
  - **G3**: final grade from [0, 20] to [0, 3] corresponding respectively to "Insufficient", "Sufficient", "Good", "Excellent".
- The data preprocessing simplified the model but came at the cost of contained information loss.

Expert-defined

- Use of domain knowledge and prior understanding of factors influencing student performance.

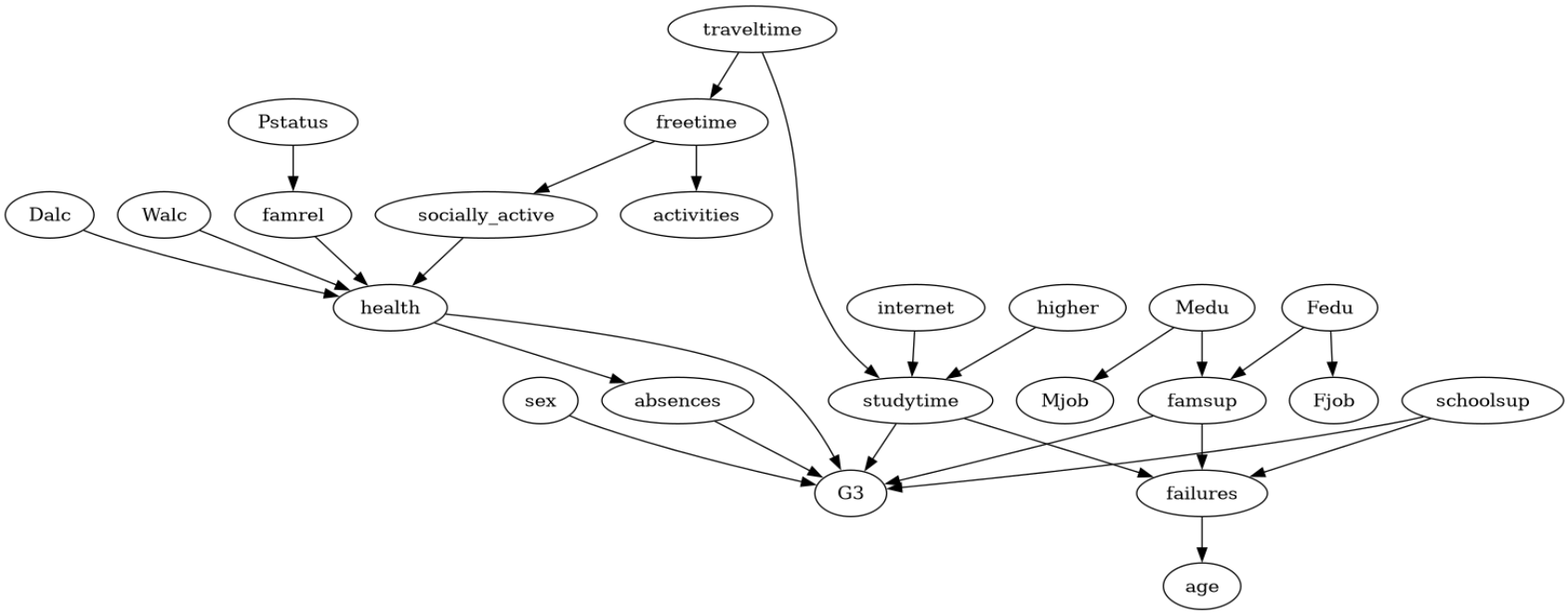
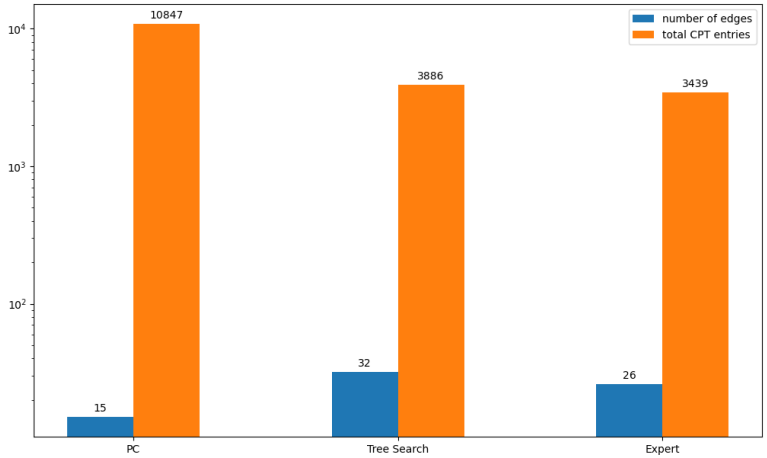


Figure: Expert-defined student performance network

Structure comparison

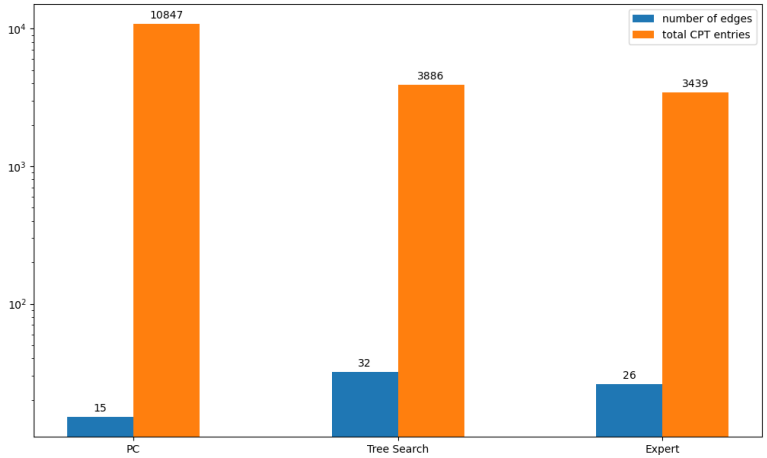
- For each network, counted the number of edges along with the number of CPT entries [1].

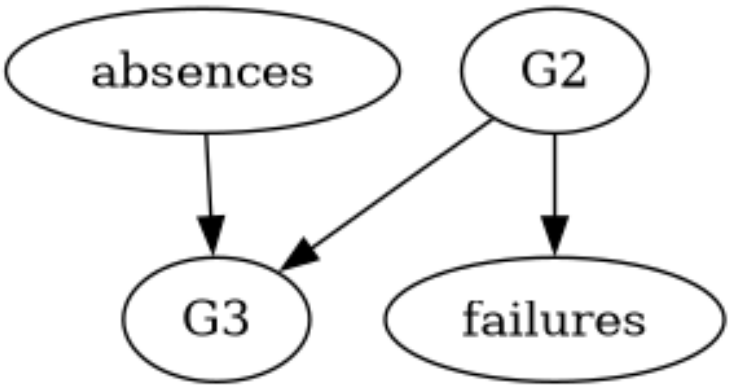


[1] pgmpy's TabularCPD stores the full conditional distribution

Structure comparison

- For each network, counted the number of edges along with the number of CPT entries [1].





In the PC network, due to $|absences| = 34$, $|G2| = 17$, and $|G3| = 17$, the CPT of G3 on its own accounts for 10404 probabilities.

- The expert-defined network is the smallest memory-wise.

[1] pgmpy's TabularCPD stores the full conditional distribution

Absences and study time

- In order to have a better chance of achieving a higher final grade, is it better to have fewer absences at the cost of less study time or is it better to have more absences but more study time ? That is to say:

$$\mathbb{P}(G3 \geq good | absences = low, studytime = average) \overset{?}{\gtrless}$$
$$\mathbb{P}(G3 \geq good | absences = medium, studytime = high)$$

Absences and study time

For the **non-re-binned networks**, the left side formula (right side is analogous) equates to:

$$\mathbb{P}(G3 \in \{15..20\}) | absences \in \{0..9\}, studytime = 3)$$

Absences and study time

Events $G3 = 15, G3 = 16, ..., G3 = 20$ are mutually exclusive:

$$\sum_{i=15}^{20} \mathbb{P}(G3 = i | absences \in \{0..9\}, studytime = 3)$$

From the conditional probability definition:

$$\sum_{i=15}^{20} \frac{\mathbb{P}(G3 = i, absences \in \{0..9\}, studytime = 3)}{\mathbb{P}(absences \in \{0..9\}, studytime = 3)}$$

Absences and study time

Events absences = j and absences = k are disjoint $\forall j, k, j \neq k$:

$$\sum_{i=15}^{20} \left(\frac{\sum_{j=0}^{9} \mathbb{P}(G3 = i, absences = j, studytime = 3)}{\sum_{j=0}^{9} \mathbb{P}(absences = j, studytime = 3)} \right)$$

Applying the chain rule:

$$\sum_{i=15}^{20} \left(\frac{\sum_{j=0}^{9} \mathbb{P}(G3{=}i|absences{=}j,studytime{=}3)\mathbb{P}(absences{=}j|studytime{=}3)\mathbb{P}(studytime{=}3)}{\sum_{j=0}^{9} \mathbb{P}(absences{=}j|studytime{=}3)\mathbb{P}(studytime{=}3)} \right)$$

Canceling out $\mathbb{P}(studytime = 3)$ & swapping summation order:

$$\frac{\sum_{j=0}^{9} \Big(\mathbb{P}(absences{=}j|studytime{=}3) \sum_{i=15}^{20} \mathbb{P}(G3{=}i|absences{=}j,studytime{=}3) \Big)}{\sum_{j=0}^{9} \mathbb{P}(absences{=}j|studytime{=}3)}$$

Absences and study time

Meanwhile, for the **re-binned network**:

$$\sum_{i=2,3} \mathbb{P}(G3 = i | absences = 0, studytime = 3)$$

Absences and study time

Meanwhile, for the **re-binned network**:

$$\sum_{i=2,3} \mathbb{P}(G3 = i | absences = 0, studytime = 3)$$

- Results:

network	prob_less	prob_more	average time (ms)
expert	0.501452	0.498276	3,142
pc	0.227796	0.145425	7,092
tree	0.227796	0.145425	6,745

- Fewer absences are more impactful than longer study time.
- Simpler query formula was faster to compute.

Travel time and health

Does having a long travel time from home to school affect students' health ? That is to say:

$$\mathbb{P}(health = good | traveltime = long) \overset{?}{\neq} \mathbb{P}(health = good)$$

$$\mathbb{P}(health = 5 | traveltime = 4) \overset{?}{\neq} \mathbb{P}(health = 5)$$

Travel time and health

network	unconditioned_prob	conditioned_prob	average time (ms)
expert	0.335300	0.336234	2,355
pc	0.369620	0.369620	0,889
tree	0.362876	0.369620	1,330

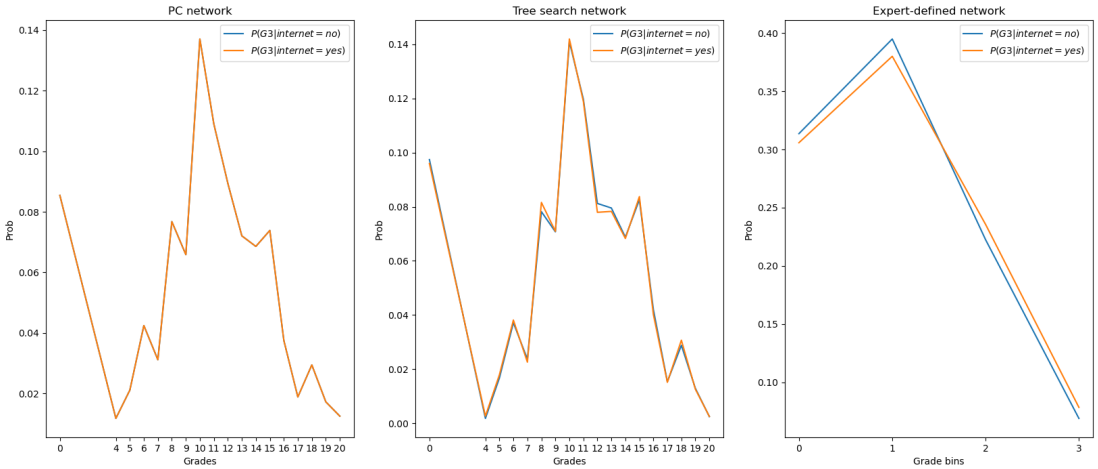
- traveltime has no significant impact on health.
- PC was the fastest due to health being independent of all other variables in this network.

Internet access

How does internet access at home affect student performance? Does it support them in their studies or does it allow for more distractions ?

- Compared distributions: $P(G3|internet = no)$ and $P(G3|internet = yes)$.

Internet access



- PC network shows no effect of internet access on performance due to $G3 \perp internet$.
- Because of the re-binning, the expert-defined network is unable to represent the distribution over the hole grade scale.
- Both Tree and expert networks show higher probabilities of better grades and lower probabilities of worse grades with internet access.

Internet access

- Unlike the previous queries, this time the fastest network on average was the Tree network.

network	**average time (ms)**
expert	4,583
pc	2,228
tree	1,566

Conclusion

- The use of domain knowledge and simplifying assumption, although they came with information loss, were pivotal to reducing the network size.
- The query efficency varied on structure, with no clear best network.
- The choice of the best network thus relies on application queries.