**Aim**

Our aim is to prove the functionality, purpose, reliability and applicability of Bayesian Networks. For that, we are performing an inference test through the likelihood weighting method using our own algorithm on a Bayes Network previously given, and an exact inference test using the open source program JavaBayes on a different Bayes Network given.We are also proving a property of a specific type of Bayes Network through mathematical calculations. Finally, we are creating an example of a Bayesian Network to address a generic problem.

This problem is important to understand better the properties and implications of a Bayesian Network, a method to determine probabilities widely used in the artificial intelligence field.

**Methods**

Bayesian networks are graphical models for representing the interaction between variables visually. The Bayesian network is a directed acyclic graph where each node corresponds to a random variable, X, and has a value corresponding to the probability of the random variable given it’s parents. The nodes and the arcs define the structure of the network. This graphical representation is visual and helps understanding. The network represents conditional independence statements and allows us to break down the problem of representing the joint distribution of many variables into local structures; this eases both analysis and computation.

Rejection sampling is a general method for producing samples of a given distribution. It can be used to compute conditional probabilities. Rejection sampling produces a consistent estimate of the true probability by repeating the heuristics many times enough to the result to converge to the actual result. The biggest problem with rejection sampling is that it rejects so many samples, that for complex problems this method is impossible to be used.

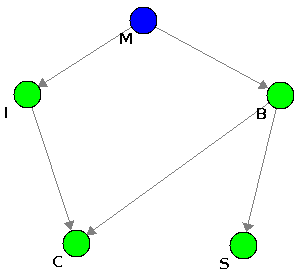
Likelihood weighting may be regarded as the optimization of the rejection sampling since it generates only events that are consistent with the evidence e to compute the conditional probability.

**Results and Discussion**

This problem is important to understand better the properties and implications of a Bayesian Network, a method to determine probabilities widely used in the artificial intelligence field.

* Task 1
* Task 2

1)



2) P(c) = 0.68

3) P(m|s,~c) = 0.2

4) Total Serum Calcium and Brain Tumor (B and I)

5) Consider

Which implies

6)

* Task 3 -1

Since there is no in the function, the independence is proven.

* Task 3 - 2
* Task 4

SD

WD

T

WO

HS

Work = 1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Sun | Mon | Tue | Wed | Thu | Fri | Sat |
| H | 0.05 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.05 |
| B | 0.05 | 0.2 | 0.8 | 0.8 | 0.8 | 0.2 | 0.05 |
| N | 0.05 | 0.95 | 0.95 | 0.95 | 0.95 | 0.95 | 0.5 |

Work = 0

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Sun | Mon | Tue | Wed | Thu | Fri | Sat |
| H | 0.95 | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.95 |
| B | 0.95 | 0.8 | 0.2 | 0.2 | 0.2 | 0.8 | 0.95 |
| N | 0.95 | 0.05 | 0.05 | 0.05 | 0.05 | 0.05 | 0.5 |

Stay Home = 1

|  |  |  |  |
| --- | --- | --- | --- |
| Work = 1 | H | N | C |
| H | 0.02 | 0.05 | 0.75 |
| B | 0.02 | 0.02 | 0.25 |
| N | 0.05 | 0.05 | 0.25 |
| Work = 0 | H | N | C |
| H | 0.05 | 0.25 | 0.95 |
| B | 0.1 | 0.25 | 0.75 |
| N | 0.4 | 0.5 | 0.9 |

Stay Home = 0

|  |  |  |  |
| --- | --- | --- | --- |
| Work = 1 | H | N | C |
| H | 0.98 | 0.95 | 0.25 |
| B | 0.98 | 0.98 | 0.75 |
| N | 0.95 | 0.95 | 0.75 |
| Work = 0 | H | N | C |
| H | 0.95 | 0.75 | 0.05 |
| B | 0.8 | 0.75 | 0.25 |
| N | 0.6 | 0.5 | 0.1 |

**Conclusions**

In this work we discovered how to sample from a Bayesian net, how to implement weighted randomness, how to specify Bayesian nets, how to design them in a open source program and how this may be helpful for visualization and querying of our data. We also observed that specific Bayesian networks may have very interesting properties, such as strong independency of the variables, and that we may find very interesting functions out of it.

We also noticed how hard it might be to design a Bayesian network and how expert knowledge is needed, since every single node probability must be defined, and also the interaction between the nodes. It was easy to notice that Bayesian networks may get very complex and hard to define it’s probability, because some data is very hard to be acquired, or even impossible.

We suggest as future work:

1. To implement likelihood weighting and variable elimination on the designed network of the task 4;
2. To implement a generic query for the Bayesian network of Task 1 such that we can observe any node and run the likelihood weighting on any other.
3. ???
4. Profit

**Reflection**

This experiment was helpful to have a better and deeper understanding of Bayesian networks since in computation the most efficient way to learn something is to implement that yourself.

In the programming part we learned how to implement Bayesian networks and to sample from them with good enough random algorithms. Then we had to learn how to use JavaBayes, which was sufficiently easy yet very interestingly because it is a very powerful tool to visualize Bayes Nets.

**Bibliography**

Introduction to Machine Learning, Ethem Alpaydin

Artificial Intelligence - A Modern Approach  
Second Edition  
Stuart J. Russel and Peter Norvig