Inferencing From Bayesian Networks
Lab 5

Aditya Gupta Milan Chaudhary

April 21, 2017

1 Introduction

In this Lab exercise, we implemented the inference from Bayesian Networks using

- Variable Elimination: which is an exact inference. It takes much space and time.
- Rejection Sampling: which is an approximated inference. It's a highly efficient algorithm, that reduces required time and space for large number of variables.

2 Variable Elimination

In this section of program, first a container with all the conditional probability tables from each node instantiated by the evidence varaibles were collected. Then for each hidden variable, all the factors with that variable in it was collected and joined together. This process was repeated until all hidden variables were joined (and summed). The result was thus the required row (after normalization) of the factor formed from joining all the remaining factors.

2.1 Reduce

For each row the index (in binary) was masked with 1 for all non-evidence variables. This leaves behind only the evidence variables and all must thus have same transformed value since evidence are given. Thus comparison was done and required rows were collected.

2.2 Join

In this operation the resultant factor will have union of all variables in both the factors, further the order of the variables is important. First the common variables were collected then the remaining variables from first then second factor. Both the tables were sorted by creating weights (for common variables) and performing a stable sort (for the remaining variables). The sort involved sorting indices instead of the whole table and then for each set of values for common variables, taken as a block, and the remaining variables's value from both factors for each block were taken and the result were calculated for each block by two nested loops.

2.3 Sum

We note that for summing over a variable's particular value, the value of the variable in the factor only changes periodically; the period was found out and then a hopping loop summed the probabilities for each value, thus finally eliminating the variable.

2.4 Normalize

The sum of the probabilities should be one, thus each row in the resultant factor was divided by the sum of probabilities of the whole factor.

3 Rejection Sampling

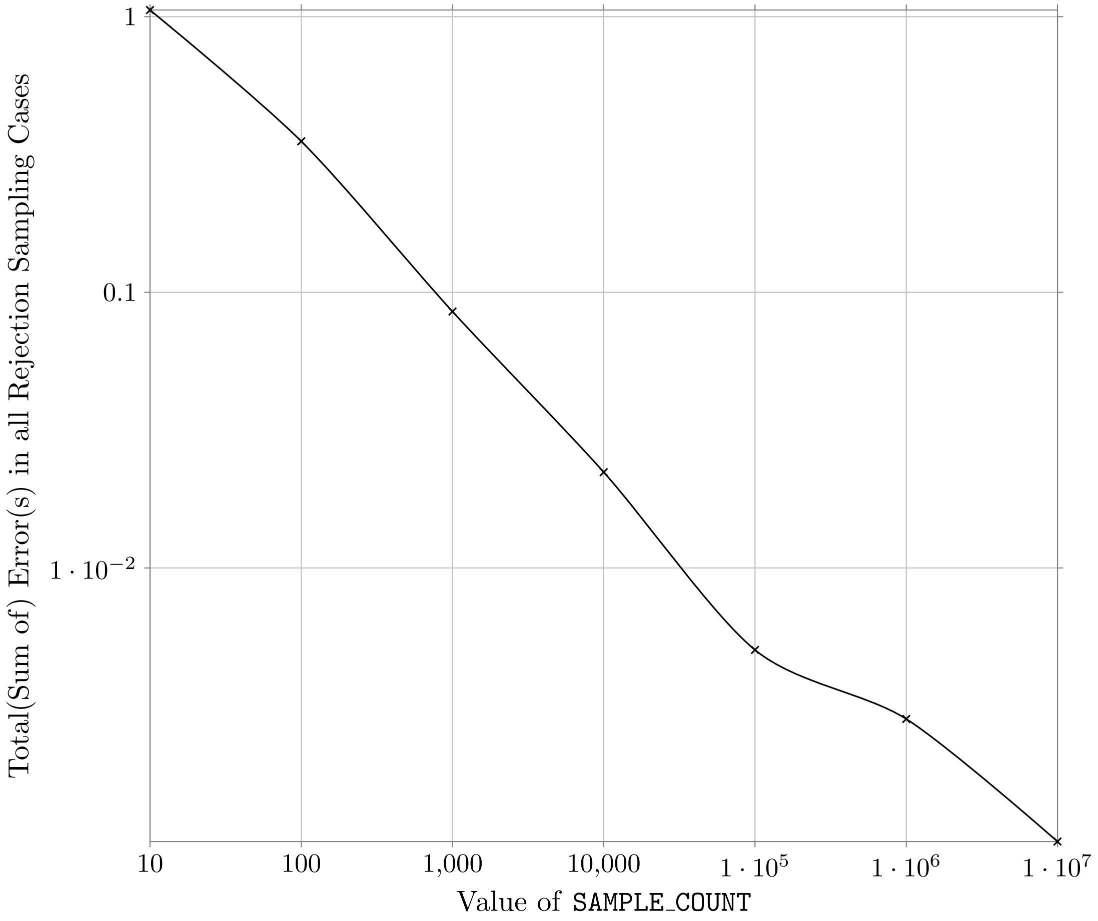
In this section of program, it will first take the topologically sorted variable list. It will sample the variable one by one from left to right in the list, which means that parents will be sampled before the child variable. This process will be done iteratively for a long finite time. In this process rejection will happen as soon as the evidences gets wrong value and will not be counted for probability calculations. The conditional probability of query variables are calculated using the non-rejected number of iterations and number of occurrences of the different values of query variables, which will be stored as a table representing probabilities of all possible value pairs of query variables. Then it will just pick the given value pair of query variables, which is the answer of input query.

Note: If the evidence value collection is not possible it will answer "These evidences don't occur at the same time, i.e. P(e) = 0, thus $P(Q|e)$ is undefined" as

$$P(Q|e) = \frac{P(Q,e)}{P(e)}$$

where $P(Q,e)$ and $P(e)$ are 0.

3.1 Convergence of the probabilities as function of number of samples generated


$1 \cdot 10^{-2}$
Total(Sum of) Error(s) in all Rejection Sampling Cases
$1 \cdot 10^{5}$
$1 \cdot 10^{6}$
$1 \cdot 10^{7}$
Value of SAMPLE_COUNT