Exact and Approximate Inference in Bayesian Networks

Jacob Regan

For this project, we were instructed to implement exact and approximate inference methods and use them to compute the posterior probability distribution for our query variables given a set of observed events. We ran these algorithms on some randomly regular directed acyclic graph (DAG) and polytree networks to investigate the performance.

The first comparison I made was a runtime analysis, the results of which are shown in the following graph. These were both done on 10 randomly generated graphs of each size, with the default options for random graph generation.

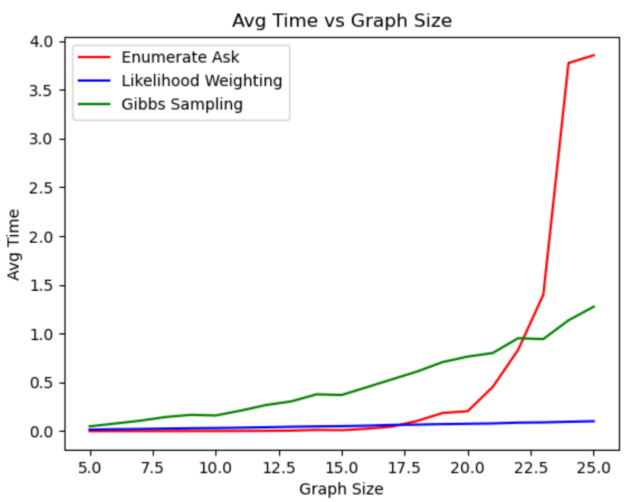


Figure 2 Regular Graph

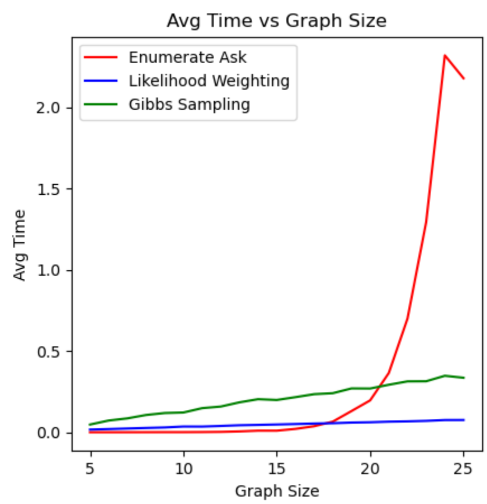


Figure 1 Polytree

As we can see, the performance of exact inference (Enumeration Ask) is initially the best and remains so until around graphs of size 17, when the exponential growth of the algorithm becomes apparent. While we can improve the runtime performance of the algorithm by structuring the larger graph as a Polytree, as show in Figure 2, the performance is still significantly worse than for the other two. I was surprised to see that Gibbs Sampling performed worst until around graph size 22 for regular graphs, and 20 for polytrees. From there, its times O(n) complexity is much more desirable, although its not as good as likelihood weighting, which compared to the other two, seemed to almost have time constant time complexity, though it is in fact O(n) as well for both cases above. This is shown in Figure 3.

Based on these results, the appeal of using one of the approximate inference algorithms (likelihood weighting or Gibbs sampling) over exact inference becomes quite apparent, especially when dealing with larger graphs.

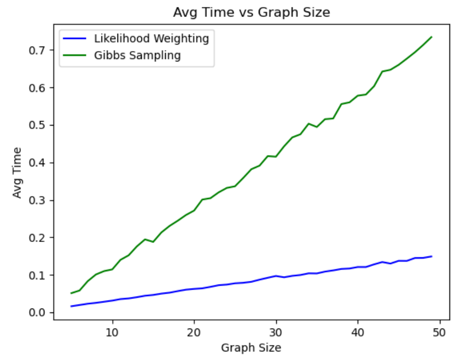


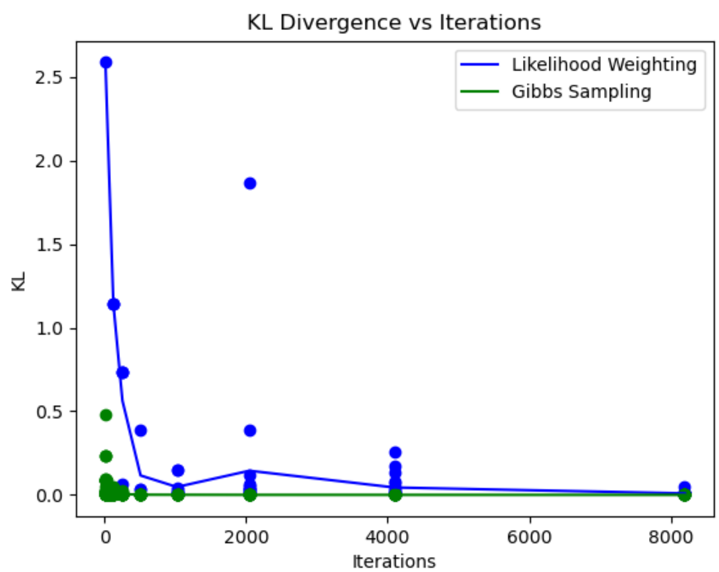
Figure 3

We also are interested in the accuracy of our results, which we can find by applying Kullback-Leibler divergence (KL divergence). KL divergence is expressed by:

$$\sum_{x \in X} P(x|e) \log \frac{P(x|e)}{\hat{P}(x|e)}$$

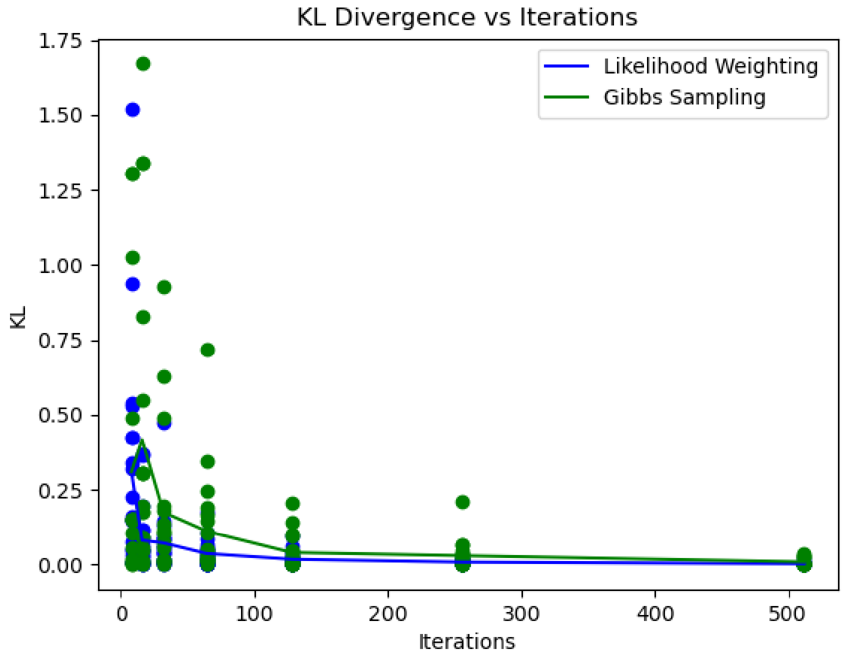
Here, $P(x|e)$ is the exact inference distribution and $\hat{P}(x|e)$ is the approximate inference distribution. The smaller the divergence, the more accurate the approximation. We apply this to compare the results of likelihood weighting and Gibbs sampling on several different data sets. Each iteration was repeated for 20 trials and averaged.

(1) The results of the Burglary example from the book for P(Burglary|JohnCalls=T, MaryCalls=T.



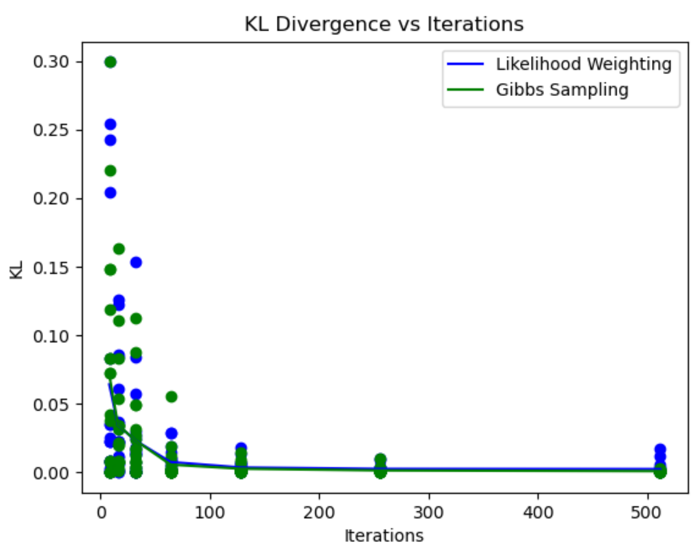
As we can see, Gibbs sampling has significant better convergence, although both were consistent by 10^4 iterations.

(2) Here are the results for a basic DAG with 20 nodes and prob=0.5. In this case, the event e is of size 5.



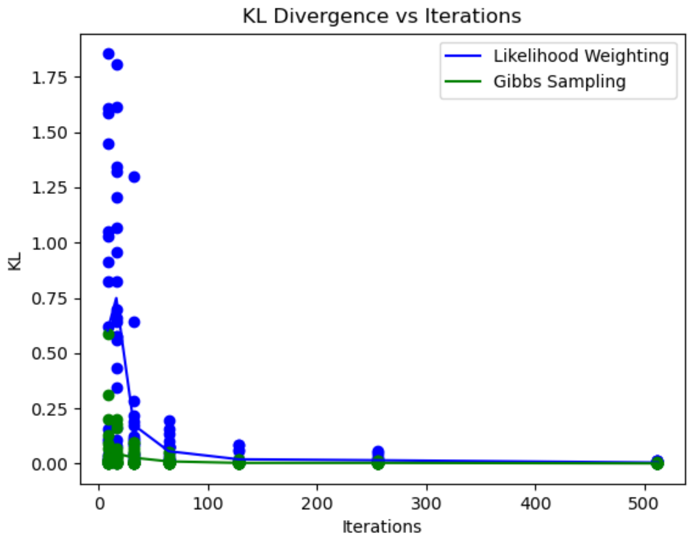
As we can see, here likelihood weighting performs better than Gibbs, though they converge much faster than before. The convergence rate was not much different with a smaller evidence set.

(3) Next, we tested the algorithms' performance on a polytree, again of size 20 with prob=0.5. The event was again of size 5.



As we can see, both algorithms had significantly better accuracy on the polytree, although they converged at a similar rate. The KL divergence was significantly better though for the polytree.

(4) Finally, we tested the performance on a larger polytree of size 30, this time with an event of size 7.



Although the KL divergence was worse, especially for likelihood weighting, it still converged just as fast as before.

In conclusion, although exact inference with variable elimination would be the ideal approach to solving Bayesian inference problems, the complexity of exact inference grows too quickly to make it viable on larger networks. Although structuring data can improve performance, ultimately an approximate inference algorithm such as Gibbs sampling or likelihood weighting will offer significantly better performance on larger networks as well as fast convergence on many networks.