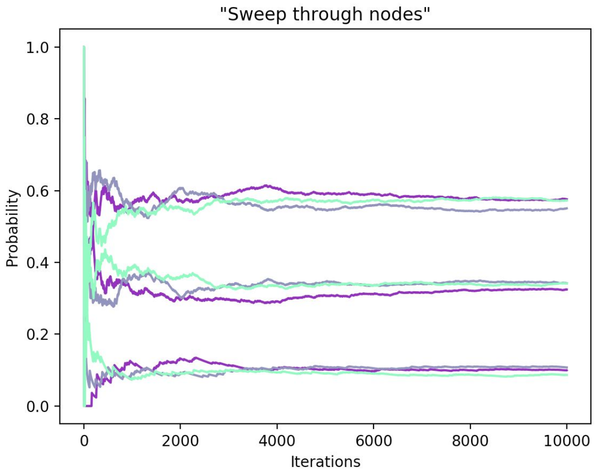
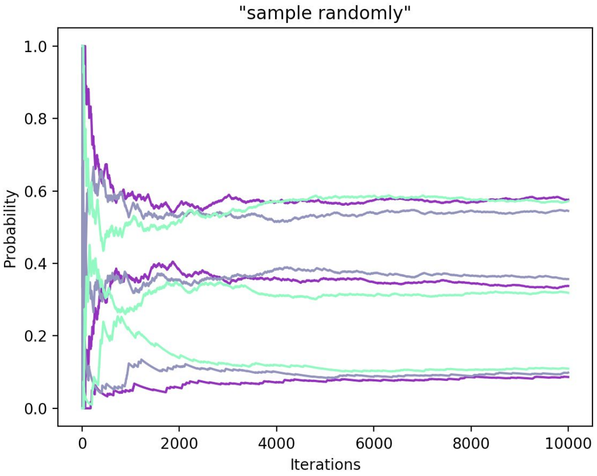
CS 534
Assignment #2: probability
Mingquan Liu, Daniel Wivagg, Mengdi Li, Ying Fang

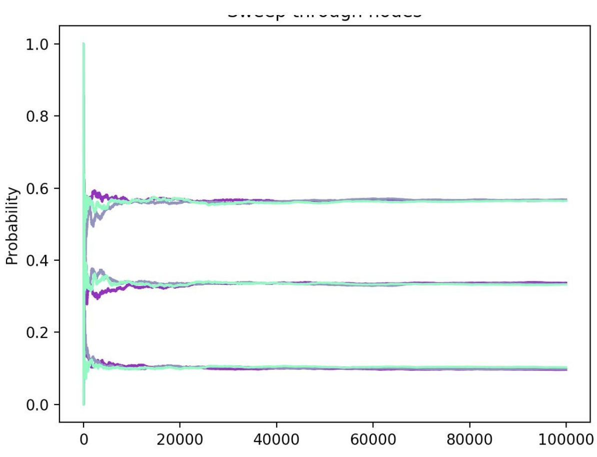
Part 1: Gibbs Sampling
The 3 queries we come up with are:
gibbs price neighborhood=good age=old -u 10000 -d 0
gibbs location age=old price=expensive -u 10000 -d 1000
gibbs schools amenities=little size=small location=ugly -u 10000 -d 1000

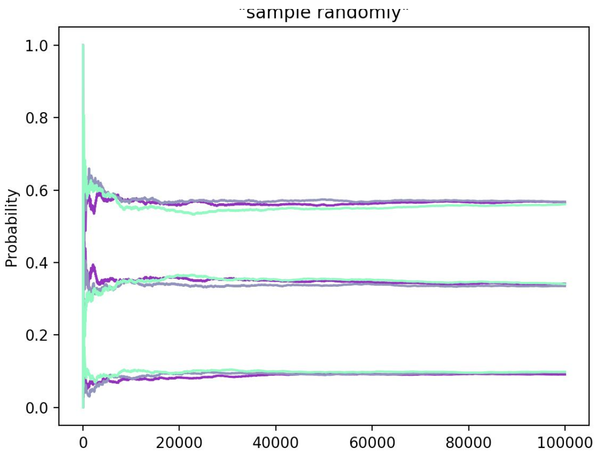
(1) Node selection:

We decided to sweep through the nodes as a batch in each sampling step. Before that, we have tried both methods of sampling, selecting node randomly and sweeping. To find out which method is better, we plotted the number of iterations vs. estimated probabilities using the first query. We applied 10,000 updates and 100,000 updates to each method three times to see which method is quicker to the convergence. As shown in graphs below, the lines in Sweep Through Nodes is smoother than the lines in Sample Randomly (which means sweeping is more stable) and takes fewer iterations to the convergence. Thus, we choose sweep through the nodes to sample.



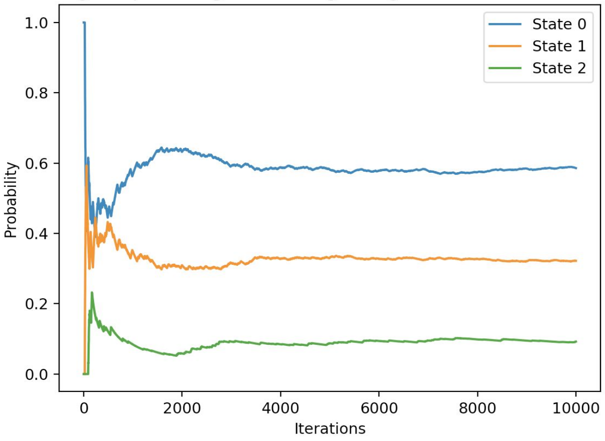


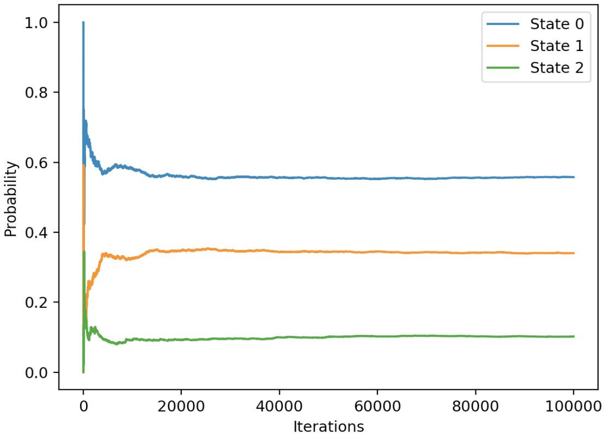




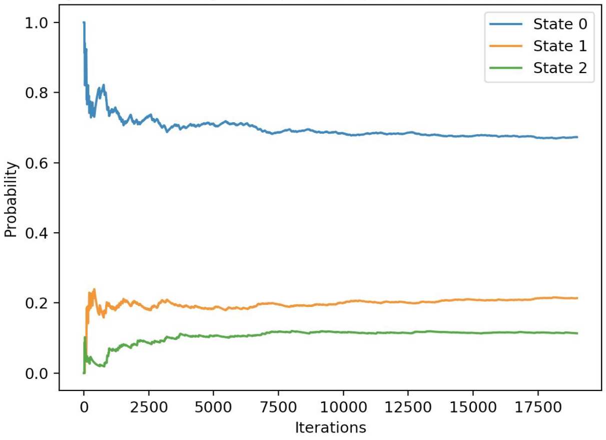
(2) Experiment:

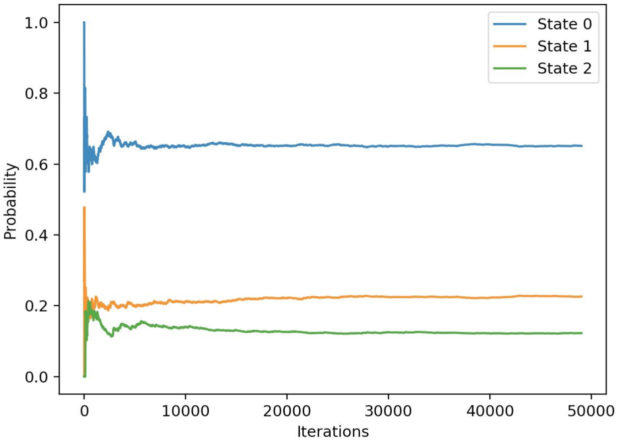
For this part, we only changed the number of samples for each query.


"gibbs price neighborhood=good age=old -u 10000 -d 0"

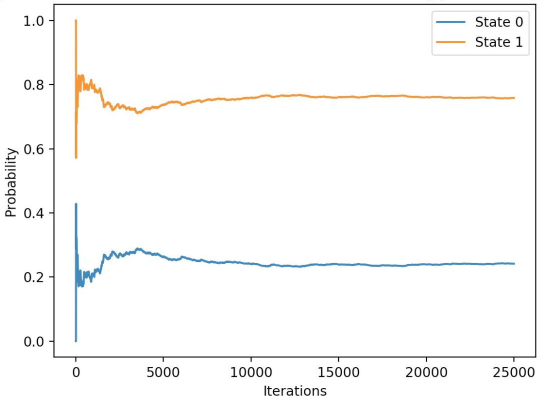

"gibbs price neighborhood=good age=old -u 100000 -d 0"

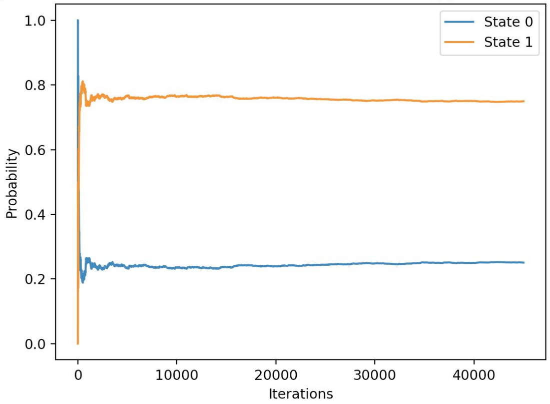
For the query of *gibbs price neighborhood=good age=old -u 10000 (100000) -d 0*, it seems like getting convergence after about 20,000 iterations.


"gibbs location age=old price=expensive -u 20000 -d 1000"


"gibbs location age=old price=expensive -u 50000 -d 1000"

For the query of *gibbs location age=old price=expensive -u 20000(50000) -d 1000*, it seems like getting convergence after about 15,000 iterations.

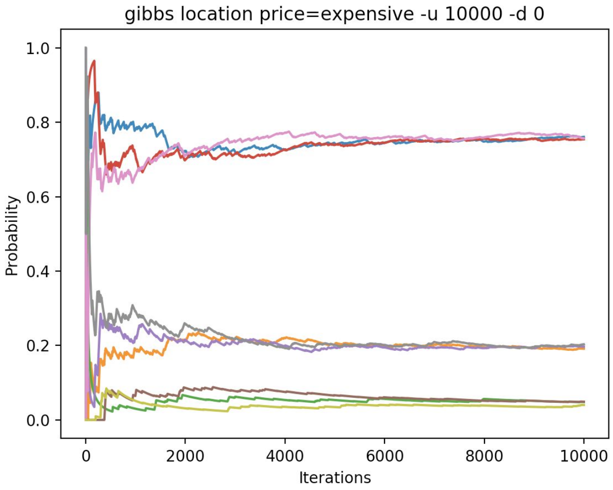

"gibbs schools amenities=little size=small location=ugly -u 30000 -d 5000

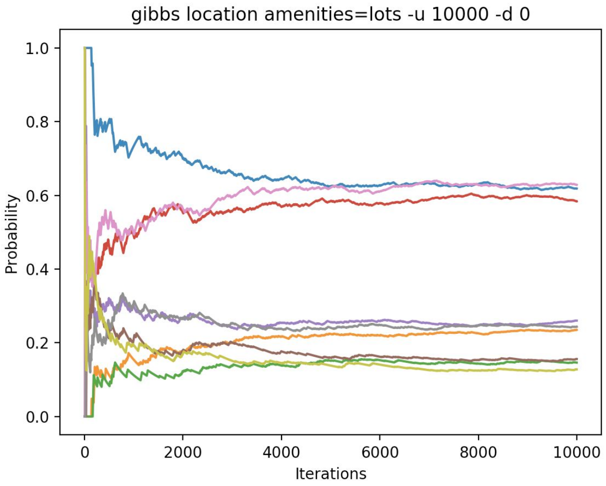

"gibbs schools amenities=little size=small location=ugly -u 50000 -d 5000

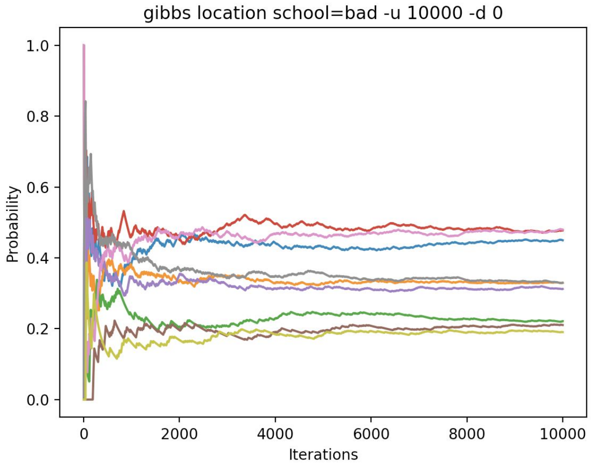
For the query of *gibbs schools amenities=little size=small location=ugly -u 30000(50000) -d 5000*, it seems like getting convergence after about 15,000 iterations.

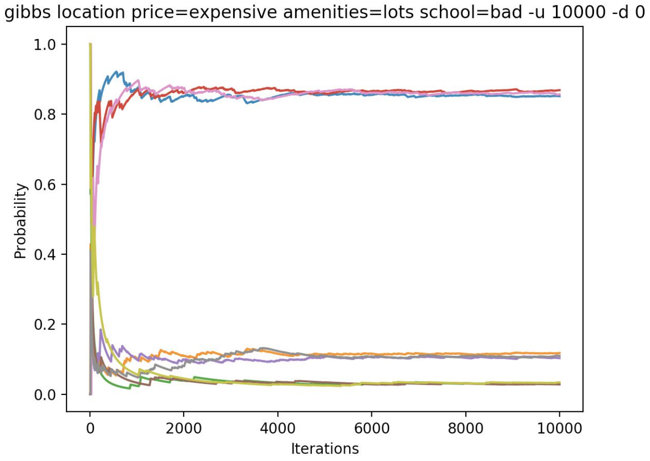
(3) Factors influenced convergence:

At first, we thought there might be four possible factors: 1. the Markov Blanket size of the query node, 2. the number of evidence nodes, 3. the Markov Blanket size of evidence node, 4. the number of dropped samples. To verify which factors have an influence on convergence, we tried to keep other factors in the same conditions and only change the scenario of the one factor. After experiments, we found that the number of evidence nodes influences the convergence a lot. We tried queries with evidence of “price=expensive”, “amenities=lots”, “school=bad” separately. Then, we set these three states as evidence at the same time. We applied each query for three times and got plots as below.





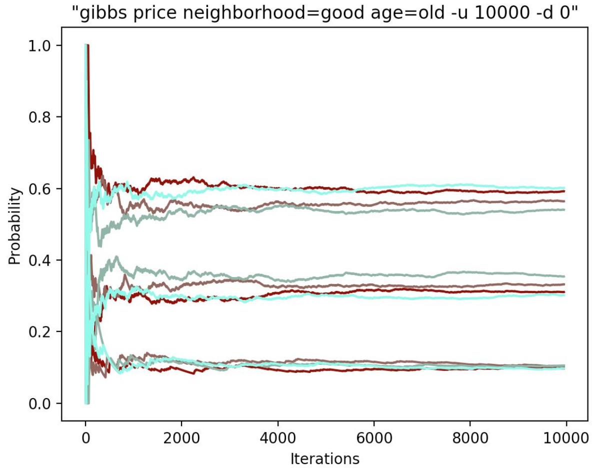


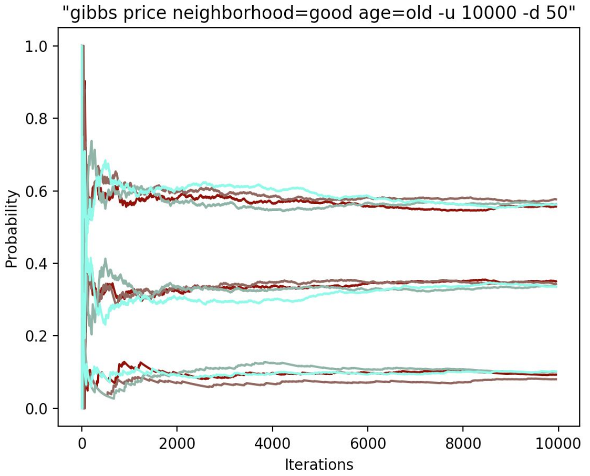


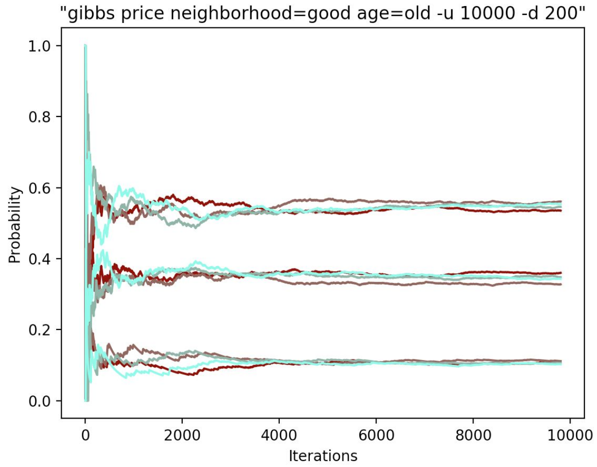
As shown in graphs above, when we only set “price=expensive” as evidence, it converges after about 7,000 iterations. When only set “amenities=lots” as evidence, it doesn’t converge after 10,000 iterations. When only set “school=bad” as evidence, it doesn’t converge after 10,000 iterations as well. When we set “price=expensive”, “amenities=lots” and “school=bad” as evidence, it converges after about 5,000 iterations. So, we can conclude that setting more evidence nodes can make convergence quicker, thus, fewer samples will be needed.

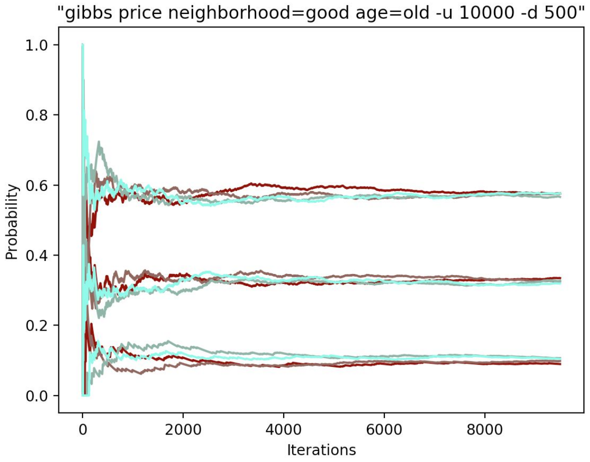
(4) Impact of drop samples on convergence:

Theoretically, we think dropping samples should have some impact on the convergence. Since the initial estimates are random, drop some samples may reduce some uncertainty and help converge. In practice, we plotted the results of a same query with different drop numbers, 0, 50, 200, 500 as shown below.









From these graphs, we think there is no significant improvement by dropping samples. It might be because the state is assigned randomly in each sample, the uncertainty still exists. Dropping samples can not help a lot with convergence.