COL776: Assignment 3

Submitted by,
Rishit Sanmukhani - 2013CS10255

Gibbs Sampling for Graphical Models

Implementation

Max. Number of samples generated = 5500
Number of Burn-in samples = 500 (10 %)
The convergence criteria is based on the marginal probability of the samples. Theoretically, the gibbs sampling must continue till the stationary distribution is obtained.
To practically implement this, I have calculated the expected assignment (which depends on marginal probability distribution). It that is stationary, I assume that the distribution is stationary.

Experiments

Following shows the comparison between the loopy belief propagation (marginal inference) and Gibbs sampling inference.
The results are almost similar (both use marginal inference)
The results of Gibbs sampling varies with number of samples (higher the samples, closer are the results) and convergence criteria.

data-tree.dat	Ch-Acc	Wd-Acc	LL	Time (ms)
Gibbs	0.667560	0.130952	-6.644602	1262.068
LBP	0.678284	0.166667	-6.629327	90.941630

data-treeWS.dat	Ch-Acc	Wd-Acc	LL	Time (ms)
Gibbs	0.636865	0.135870	-7.245039	3039.843
LBP	0.669978	0.179348	-7.248585	295.473828

data-loops.dat	Ch-Acc	Wd-Acc	LL	Time (ms)
Gibbs	0.618705	0.071429	-7.498842	569.134
LBP	0.539568	0.071429	-7.563060	60.083693

data-loopsWS.dat	Ch-Acc	Wd-Acc	LL	Time (ms)
Gibbs	0.671296	0.146154	-6.925508	2568.421
LBP	0.705247	0.207692	-6.866533	876.623621

Parameter Learning in Probabilistic Graphical Models

Bayesian Network Learning
Output extension = <name>.out.bn
Done laplacian smoothing while normalizing and counting marginal probability.

For inferrence over random variables, we only sample unknown variables. This doesn't produce samples which are inconsistent with the given data (rejection sampling).

Also for inference, we only consider CPTs in markov blanket of the sampled variable. This reduces the complexity.

Markov Network Learning
Output extension = <name>.out.mn
Done laplacian smoothing while normalizing and counting marginal probability.

Inference is same as bayesian network learning.

But learning is different, because normal counting doesn't work in case of markov network. So we do gradient descent (as discussed in class).

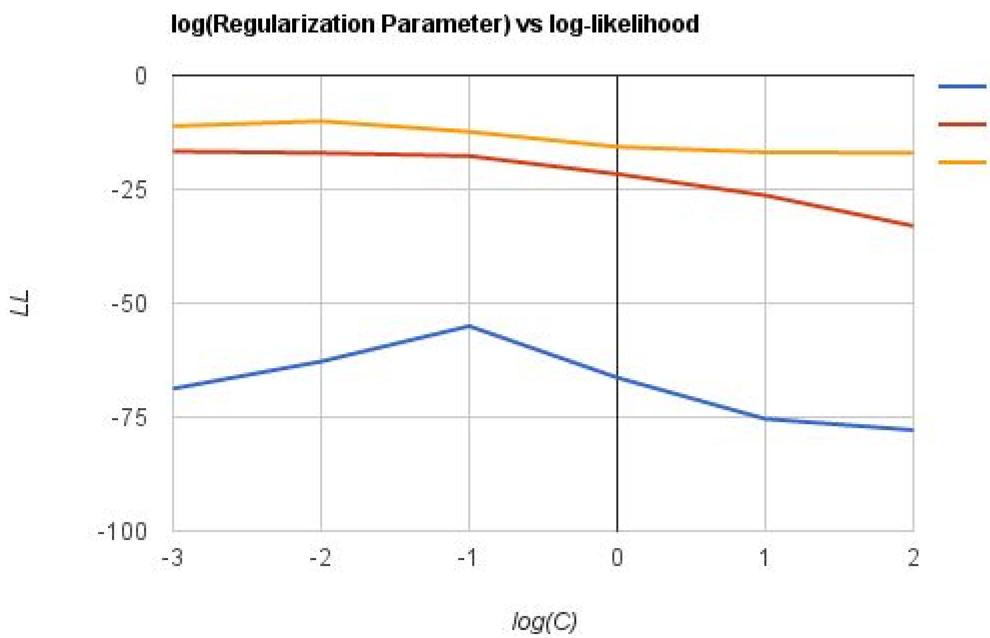
Evaluation Metric

Bayes Network	Ch-Acc	LL
andes.bif	0.777947	-60.6617
hepar2.bif	0.800176	-15.8056
insurance.bif	0.816636	-5.99603

Markov Network	Ch-Acc	LL
andes.bif	0.751186	-67.8575
hepar2.bif	0.782727	-16.6278
insurance.bif	0.807876	-9.03315

As we can observe that undirected graphical model doesn't show much improvement (rather the accuracy decreases).
The main reason for low accuracy is difficulty in learning Markov Model. In bayesian network, learning is reduced to counting, while in markov we have to rely on gradient descent.
This is difficult process and parameter tuning is difficult. So we don't learn optimal parameters, thus getting low accuracy.

Also, I have experimented with the C-parameter.
As you can see, the log-likelihood is maximum for certain value of C. And it is different for each data sets.


log(Regularization Parameter) vs log-likelihood
0
-25
-50
-75
-100
LL
-3
-2
-1
0
1
2
log(C)