Assignment 11

Machine Learning COMS 4771

Spring 2014, Itsik Pe'er

Assigned: April 21st Due: Wednesday, April 30th 1:10pm

Submission: Your submission folder on Courseworks. Submit folders for `Assignment11.Problem01`, `Assignment11.Problem02`, `Assignment11.Problem03` and `Assignment11.Problem04`

Consider the Chessboard City scenario of Assignment 10, Q3 (the 18 blocks at or adjacent to City Hall Neighborhood are observed to be uncontaminated). In addition, you have stationed 8 sensors that detect the epidemic pathogen. 4 of them are at the city corners a1,h1,a8,h8 and the other 4 are at the approximate border midpoints e1,h5,d8,a4. Assume the entire Bayes Net of Chessboard City is resampled i.i.d (wind-based infection occurs instantaneously, and every timepoint resets resets all the blocks afresh.

1) Perform exact inference using the Junction Tree Algorithm implemented by the Bayes Net Toolbox, to learn to get the marginal probabilities of each of respective your sensors to indicate contamination.

```
function [ marginal ] = GetMarginals()
%GetMarginals
% compute marginal probabilities for ChessBoardCity sensors
% returns a vector of 8 real values that have the marginal probabilities
% of respective blocks a1,e1,h1,a4,h5,a8,d8,h8 to be contaminated
```

Note that this function needs to create the Chessboard City Bayes net inside the function.
[20pt]

2) Perform exact inference using the Junction Tree Algorithm implemented by the Bayes Net Toolbox, to learn to get the marginal probabilities of each of respective your sensors to indicate contamination, given information on one of the blocks.

```
function [ marginal ] = GetMarginalsGiven1(node)
%GetMarginalsGiven1
% compute marginal probabilities for ChessBoardCity sensors
% input:
% node number to marginalize over (integer in 1..64, where 1=a1, 8=a8,
% 64=h8)
% Output:
% marginal: a 8x2 matrix of real values, the i-th column of which
% has the marginal probabilities
% of respective blocks a1,e1,h1,a4,h5,a8,d8,h8 to be contaminated,
% given that node is observed to be i
```

Note that this function also needs to create the Chessboard City Bayes net inside the function. If the input node is a sensor, the probabilities for that sensor would be 0 or 1.

[20pt]

3) Perform exact inference using the Junction Tree Algorithm implemented by the Bayes Net Toolbox, to learn to get the marginal probabilities of each of respective your sensors to indicate contamination, given information on **several** of the blocks.

```
function [ marginal ] = GetMarginalsGivenSeveral( nodes)
```

```
%GetMarginalsGivenSeveral
% compute marginal probabilities for Chessboard City sensoors
% Input: the set of nodes n1,n2,...,nk
% to be marginalized over ( a vector of integers,
% all in 1..64 , where 1=a1, 8=a8, 64=h8)
% returns a 8x2x2...x2 matrix of real values, the (i1,i2,...,ik)-th column of
% which
% has the marginal probabilities
% of respective blocks a1,e1,h1,a4,h5,a8,d8,h8 to be contaminated,
% given that node n1 is observed to be i1,
% node n2 is observed to be i2, ...
% node nk is observed to be ik
```

Note that this function also needs to create the Chessboard City Bayes net inside the function. If the input nodes include one or more sensors, the probabilities for such sensors would be 0 or 1. [20pts]

4) Use the above to figure out a Bayes net with 8 nodes corresponding to the 8 sensors, that describes the joint distribution over the sensors. This would involve running the marginal inference functions above several times to compute conditional probabilities. Submit a matlab script `run_me.m` that prints the conditional probabilities that you may need, and a description `read_me.pdf` of the Bayes net [50pt]

   Good luck!