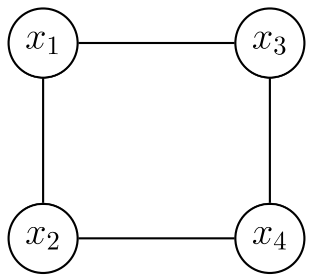
University of Illinois Spring 2018

CS 446: Machine Learning
Homework 7

Due on Tuesday, March 6, 2018, 11:59 a.m. Central Time

1. [**4 points**] Inference Methods for Discrete Markov Random Fields

   For this problem, consider the following Markov Random Field, where each node can be assigned one of the values in $\{1, 2, 3, 4, 5\}$:



   (a) To conduct MAP inference on this graph using exhaustive search, how many configurations must be checked?

Your answer: With the exhaustive search we have 5^4 choices

   (b) Can MAP inference be run on this graph using a dynamic programming algorithm? Why or why not?

Your answer: No, because dynamic programming algorithm only works with tree graphs. This is not a tree but a loop.

   (c) To run MAP inference on this graph using loopy belief propagation, how many messages must be computed?

Your answer: r = {{1},{2},{3},{4},{1,2},{1,3},{2,4},{3,4}} Hence we have 8×5 (message from each child to each parent $\times$ values of nodes)Lagrange multipliers. Therefore we have 40 messages.

2. [**7 points**] ILP Inference formulation in Discrete Markov Random Fields

   (a) Suppose we have two variables $x_1 \in \{0, 1\}$ and $x_2 \in \{0, 1\}$ and their local evidence functions $\theta_1(x_1)$ and $\theta_2(x_2)$ as well as a pairwise function $\theta_{1,2}(x_1, x_2)$. Using this setup, inference solves $\arg\max_{x_1,x_2} \theta_1(x_1) + \theta_2(x_2) + \theta_{1,2}(x_1, x_2)$. Using

$$\theta_1(x_1) = \begin{cases} 1 & \text{if } x_1 = 0 \\ 2 & \text{otherwise} \end{cases} \qquad \theta_2(x_2) = \begin{cases} 1 & \text{if } x_2 = 0 \\ 2 & \text{otherwise} \end{cases}$$

$$\theta_{1,2}(x_1, x_2) = \begin{cases} -1 & \text{otherwise} \\ 2 & \text{if } x_1 = 0 \ \& \ x_2 = 1 \end{cases}$$

what is the integer linear programming formulation of the inference task? Make the cost function and constraints explicit for the given problem, i.e., do not use a general formulation.

Your answer:

$$\max \begin{bmatrix} b_1(0) \\ b_1(1) \\ b_2(0) \\ b_2(1) \\ b_{1,1}(0,0) \\ b_{1,2}(1,0) \\ b_{1,2}(0,1) \\ b_{1,2}(1,1) \end{bmatrix}^T \begin{bmatrix} \theta_1(0) \\ \theta_1(1) \\ \theta_2(0) \\ \theta_2(1) \\ \theta_{1,1}(0,0) \\ \theta_{1,2}(1,0) \\ \theta_{1,2}(0,1) \\ \theta_{1,2}(1,1) \end{bmatrix} = \begin{bmatrix} b_1(0) \\ b_1(1) \\ b_2(0) \\ b_2(1) \\ b_{1,2}(0,0) \\ b_{1,2}(1,0) \\ b_{1,2}(0,1) \\ b_{1,2}(1,1) \end{bmatrix}^T \begin{bmatrix} 1 \\ 2 \\ 1 \\ 2 \\ -1 \\ -1 \\ 2 \\ -1 \end{bmatrix}$$

with constraints equations as:

$$b_1(0), b_1(1), b_2(0), b_2(1), b_{1,2}(1,1), b_{1,2}(1,0), b_{1,2}(0,1), b_{1,2}(0,0) \in \{0,1\}$$

and

$$b_1(0) + b_1(1) = 1$$

$$b_2(0) + b_2(1) = 1$$

$$b_{1,2}(1,1) + b_{1,2}(1,0) + b_{1,2}(0,1) + b_{1,2}(0,0) = 1$$

and

$$b_{1,2}(1,1) + b_{1,2}(1,2) = b_1(1)$$

$$b_{1,2}(2,1) + b_{1,2}(2,2) = b_1(2)$$

$$b_{1,2}(1,1) + b_{1,2}(2,1) = b_2(1)$$

$$b_{1,2}(1,2) + b_{1,2}(2,2) = b_2(2)$$

(b) What is the solution (value and argument) to the program in part (a).

Your answer: The function will be maximum when $b_1(0)$, $b_2(1)$ and thereby $b_{1,2}(0,1)$ are 1.
Then we get function value as 5 to be maximum.

(c) Why do we typically not use the integer linear program for reasonably sized MRFs?

Your answer: ILP's are run-time slow. We do not use integer linear program for reasonably sized MRF because it is repeatedly calling linear program and we could instead use linear program to save repeated calls.