MIT CSAIL

6.869 Advances in Computer Vision

Fall 2019

Problem Set 5: Belief Propagation

Posted: Thursday, Oct 10, 2019 **Due:** Thursday, Oct 17, 2019

Note: 6.819 students are expected to complete problems 1 and 2; 6.869 students are expected to complete problems 1, 2, and 3.

We provide a python notebook with the code to be completed. You can run it locally or in Colab (upload it to Google Drive and select 'open in colab') to avoid setting up your own environment. Once you finish, run the cells and download the notebook to be submitted.

Submission Instructions: Please submit a .zip file named <your kerberos>.zip containing 1) report named report.pdf including your answers to all required questions with images and/or plots showing your results, and 2) the python notebook provided, with the cells run and the relevant source code. If you include other source code files for a given exercise, please indicate it in the report.

Late Submission Policy: If your pset is submitted within 7 days (rounding up) of the original deadline, you will receive partial credit. Such submissions will be penalized by a multiplicative coefficient that linearly decreases from 1 to 0.5.

Problem 1 [4 points] *Markov Network*

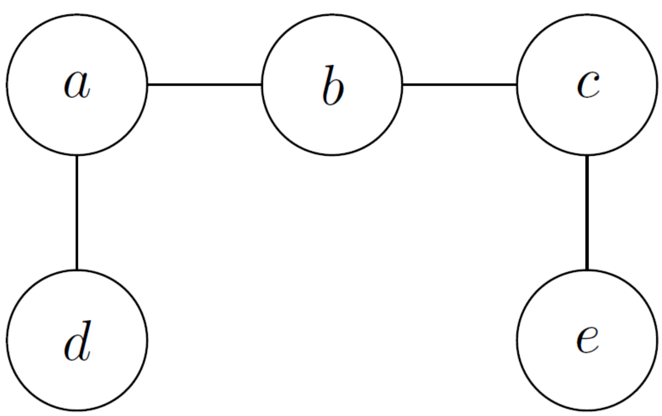

Figure 1: A Markov network

Consider the Markov network in Figure 1. Each variable is binary and can be in state 0 or state 1. d is observed to be in state 1 and e is observed to be in state 0. Additionally, the compatibility matrices Φ and Ψ are given by

$$\Phi\left(a,d\right)=\begin{pmatrix}0.9 & 0.1\\ 0.1 & 0.9\end{pmatrix}=\Phi\left(c,e\right)$$

$$\Psi\left(a,b\right)=\begin{pmatrix}\alpha & 1-\alpha\\ 1-\alpha & \alpha\end{pmatrix}=\Psi\left(b,c\right)$$

(a) [**2 points**] For $\alpha = 0.99$ find P(a), the marginal probabilities of variable a being in either of its two possible states, 0 and 1.

(b) [**2 points**] Do the same for $\alpha = 0.6$. Discuss why the result is different in these two cases.

Problem 2 [6 points] *Belief Propagation* Many vision problems consist of measuring local evidence, then propagating it across space. Belief propagation is often useful for such tasks. This homework problem was presented as a belief propagation example by Yair Weiss in a NIPS paper [1].

A task of early vision is to make a figure/ground assignment: which side of a contour is the foreground object, and which side is the background? A good cue for that assessment is convexity. Contours typically encircle the object, rather than form holes within it, so the foreground side is often on the inside of a contour's curve.

Locally, a complex contour may bend both ways and only a global assessment of convexity can tell us the right answer. We define a Markov chain of points along a contour in an image (we assume this contour has already been detected). The hidden states are the side of the foreground assignment for the contour (+1 means to the right as you traverse the contour, incrementing the node index; -1 is to the left). The local evidence at each node is based on the local curvature, defined by the angle θ_j based on the local three adjacent points on the curve (nodes $j-1$, j, and $j+1$). Let $\theta_j = 0$ correspond to a straight line, and $\theta_j = \frac{\pi}{2}$ correspond to a 90° right bend, and $\theta_j = -\frac{\pi}{2}$ correspond to a 90° left bend.

Let the local evidence for a positive or negative curvature curve be:

$$\phi(x_j, y_j) = \begin{pmatrix}\frac{\pi+\theta_j}{2\pi}\\ \frac{\pi-\theta_j}{2\pi}\end{pmatrix}$$

This favors figure/ground evidence in proportion to the acuteness of the local angle of bending.

The hidden state compatibility requires that hidden states have the same value as that of the neighboring node: $\psi\left(x_j, x_{j+1}\right) = I_2$ (i.e. the identity matrix).

The joint probability of figure/ground estimates conditioned on the observed curve is given by the product of the local evidence and the node capabilities:

$$P\left(\vec{x}|\vec{y}\right)=\prod_j \phi\left(x_j,y_j\right)\psi\left(x_j,x_{j+1}\right)$$

In the python notebook, you will find three images and their curves, which are arrays containing x and y coordinates of points along the boundary. In general, we can use computer vision algorithms (some of which you have already encountered) for extracting such contours from images, however here we produced them manually for you to use.

(a) [**1 point**] Load and plot the supplied curves in the 2D plane for the three included images. Note that the coordinate system has its origin as the top left corner of the image. Include your plots and images in your write-up.

(b) [**2 points**] Complete the code to indicate the local direction of figure (plot a small arrow to the foreground side at every node) based on local evidence alone, before running belief propagation. Include your plots and images in your write-up.

(c) [**3 points**] Implement belief propagation and show the final estimated direction of the arrows after running the algorithm. Include your plots and images in your write-up.

Problem 3 (6.869 only) [3 points] *Restricted Boltzmann Machines*

Restricted Boltzmann Machines (RBMs) are a class of probabilistic graphical models that have applications in several areas of computer vision, including image feature extraction and image classification. An RBM is a bipartite Markov network consisting of a hidden layer and an observed layer, where each node is a random variable taking on one of only two values. More specifically, it models latent factors that may be learned from some input features. Let us consider an RBM used in image classification. Specifically, suppose we are trying to discern between a human (H_1) and a tree (H_2). These make up our latent, or hidden, nodes. Additionally, suppose that we have five different binary, visible observations: "has a face" (V_1), "has leaves" (V_2), "has clothes" (V_3), "has bark" (V_4), and "has arms and legs" (V_5) [2].

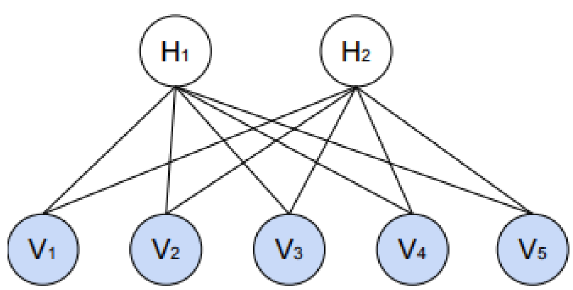

Figure 2: An example of RBM with 5 visible nodes and 2 hidden nodes.

In the following questions, let $V = (V_1, \ldots, V_5)$ be a vector of observations (e.g. the observation $v = (1, 0, 1, 0, 1)$ implies that someone observed an object that has a face, is wearing clothes, and has arms and legs). Similarly, let $H = (H_1, H_2)$ be a vector of latent factors. Again, note that all the random variables are binary and take on states in $\{0, 1\}$. The joint distribution of such a configuration can be modeled by:

$$p_{V,H}(v, h) = \frac{1}{Z} \exp(-E(v, h)) \tag{1}$$

where

$$E(v,h) = -\sum_{i,j} w_{ij} v_i h_j - \sum_i a_i v_i - \sum_j b_j h_j$$

is the associated energy function, $\{w_{ij}\}$, $\{a_i\}$, $\{b_i\}$ are model parameters, and

$$Z = Z(\{w_{ij}\}, \{a_i\}, \{b_i\}) = \sum_{v,h} \exp(-E(v,h))$$

is the partition function, where the summation runs over all joint assignments to V and H.

(a) [**1 point**] Using the above equations, show that $p_{H|V}(h|v)$ (the distribution of the hidden nodes conditioned on all of the observed nodes) can be factorized as:

$$p_{H|V}(h|v) = \prod_j p_{H_j|V}(h_j|v)$$

where

$$p_{H_j|V}(1|v) = \sigma\Big(b_j + \sum_i w_{ij} v_i\Big)$$

and $\sigma(s) = \frac{\exp(s)}{1+\exp(s)}$ is the sigmoid function. Note that $p_{H_j|V}(0|v) = 1 - p_{H_j|V}(1|v)$.

(b) [**1 point**] Give the factorized form of $p_{V|H}(v|h)$, the distribution of the visible units conditioned on all of the hidden units (This should be similar to what was given in part (a) and so you may omit the derivation).

(c) [**1 point**] Based on your answers so far, does the distribution in Equation (1) respect the conditional independencies of the graph in Figure (2)? Explain why or why not. Are there any independencies in Figure (2) that are not captured in Equation (1)?

References

[1] Yair Weiss. "Interpreting images by propagating Bayesian beliefs". In: Advances in Neural Information Processing Systems 9. `<http://www.cs.huji.ac.il/~yweiss/nips96.pdf>`. 1996, pp. 908-915.

[2] Probabilistic Graphical Models, 10-708, Carnegie Mellon University as taught by Prof. Eric Xing.