

CS 228, Winter 2008

Problem Set #4

1. Parameter Estimation in Template-Based Models [20 points]

In class, we talked about parameter learning in the case of partially observed data for general Bayesian networks. Here, we apply these methods to the special case of plate models. Consider the plate model in Figure 1 that has two plates, $P1$ and $P2$, with M and N copies, respectively. Let \mathcal{X} be the variables that lie only in $P1$, let \mathcal{Y} be the variables that lie only in $P2$, and let \mathcal{Z} be the variables that lie in both plates. Each variable in \mathcal{Z} must either have parents in both \mathcal{X} and \mathcal{Y} , have parents in \mathcal{Z} , or both these must be true. (For a concrete example, see Figure 6.6 in the book.) Assume all variables are binary variables.

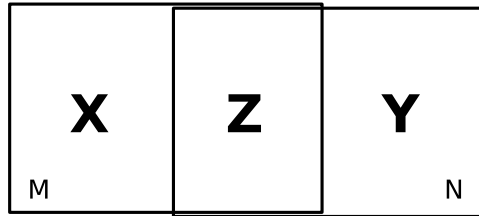


Figure 1: Plate Model

- (a) [10 points] Suppose you are given the plate model as above, and a data set \mathcal{D} where *each training sample* is a full assignment to *all* copies of the variables in \mathcal{X} , *all* copies of the variables in \mathcal{Y} , and *all* copies of the variables in \mathcal{Z} . Let $\theta_{\mathcal{X}}$ denote the parameter vector for the CPDs of \mathcal{X} , and similarly for $\theta_{\mathcal{Y}}$ and $\theta_{\mathcal{Z}}$. Thus, for example, for a variable $X_i \in \mathcal{X}$, we have a parameter $\theta_{x_i|\mathbf{u}}$ for each assignment x_i to X_i and \mathbf{u} to \mathbf{Pa}_{X_i} (note that the parents of X_i must lie in $P1$ and therefore be a subset of \mathcal{X}). Define appropriate sufficient statistics for this likelihood function, and write down the explicit form of the likelihood function in terms of the parameters and your sufficient statistics. (Note that it is not enough to simply use the term “counts” or $M[\cdot]$ when defining sufficient statistics, because the definition in the book does not precisely correspond to this particular setting. You may have to define an analogous quantity for the case of plate models.)
- (b) [4 points] Using the likelihood function you obtained for part (a), derive maximum likelihood estimates for the different parameters in $\theta_{\mathcal{X}}$, $\theta_{\mathcal{Y}}$, and $\theta_{\mathcal{Z}}$.
- (c) [6 points] Now, consider the case of partially observed data. In particular, let $\mathbf{H} \subset \mathcal{X} \cup \mathcal{Y} \cup \mathcal{Z}$ be variables that are hidden (unobserved) for all instances. How would you execute the EM algorithm to learn parameters? Specifically, you should describe how to perform the E-step and M-step of the algorithm. Your computations for each step must be tractable. It is fine to use a description that calls other algorithms as a subroutine, as long as it is clear what precisely their input is, and how their output is used.

2. Learning CRFs [20 points]

In class and in the course notes, we discussed the problem of estimating the parameters of both Markov Nets and Bayesian networks using Maximum Likelihood Estimation (MLE). In this problem, we consider MLE for the parameter estimation of the Conditional Random Fields (CRFs).

Recall that a CRF encodes the following distribution:

$$P(\mathbf{Y}|\mathbf{X}) = \frac{1}{Z(\mathbf{X})} P'(\mathbf{Y}, \mathbf{X})$$

In this problem, we will consider the log-linear parameterization of a CRF, so that the network is annotated with a set of n features $f_i[\mathbf{X}_i, \mathbf{Y}_i]$, where $\mathbf{Y}_i \neq \emptyset$, and weights w_i . Thus we have

$$\begin{aligned} P'_{\mathbf{w}}(\mathbf{Y}, \mathbf{X}) &= \prod_{i=1}^n \exp(w_i f_i(\mathbf{X}_i, \mathbf{Y}_i)) \\ Z(\mathbf{X}) &= \sum_{\mathbf{Y}} P'_{\mathbf{w}}(\mathbf{Y}, \mathbf{X}). \end{aligned}$$

In class and in the course notes posted online, we showed that the derivative of the log-likelihood for a standard log-linear Markov Network was

$$\frac{\partial}{\partial w_i} \ell(\mathcal{D} : \mathbf{w}) = (\mathbf{E}_{\hat{P}}[f_i] - \mathbf{E}_{P_{\mathbf{w}}}[f_i])$$

where $\mathbf{E}_{\hat{P}}[f_i]$ is the empirical expectation of f_i in the dataset and $\mathbf{E}_{P_{\mathbf{w}}}[f_i]$ is the expectation of f_i in our model parameterized by \mathbf{w} .

Now we come to the questions. In the following, you should assume you are given a dataset $\mathcal{D} = \{\langle \mathbf{x}[1], \mathbf{y}[1] \rangle, \dots, \langle \mathbf{x}[m], \mathbf{y}[m] \rangle\}$.

- (a) [3 points] Write the log-likelihood $\ell(\mathbf{w}, \mathcal{C})$ for a log-linear CRF \mathcal{C} .
 (b) [12 points] Prove that the derivative of $\ell(\mathbf{w}, \mathcal{C})$ with respect to w_i is the following:

$$\frac{\partial}{\partial w_i} \ell(\mathbf{w}, \mathcal{C}) = \sum_{j=1}^m f_i(\mathbf{x}_i[j], \mathbf{y}_i[j]) - \mathbf{E}_{(\mathbf{Y}_i|\mathbf{x}[j]) \sim P_{\mathbf{w}}} [f_i(\mathbf{x}_i[j], \mathbf{Y}_i)]$$

where $\mathbf{E}_{(\mathbf{Y}_i|\mathbf{x}[j]) \sim P_{\mathbf{w}}} [f_i(\mathbf{x}_i[j], \mathbf{Y}_i)]$ is the expectation of f_i given $\mathbf{x}[j]$ in our CRF with distribution $P_{\mathbf{w}}$. That is,

$$\mathbf{E}_{(\mathbf{Y}_i|\mathbf{x}[j]) \sim P_{\mathbf{w}}} [f_i(\mathbf{x}_i[j], \mathbf{Y}_i)] = \sum_{\mathbf{Y}_i} f_i(\mathbf{x}_i[j], \mathbf{y}_i) P_{\mathbf{w}}(\mathbf{y}_i|\mathbf{x}[j]).$$

- (c) [5 points] Given the above derivative, why is learning a CRF computationally more expensive than learning a standard (generatively trained) Markov network?

3. **Structural EM [20 points]** Recall that structural EM (SEM) uses the existing candidate Bayesian network to complete the data, and then uses the completed data to evaluate each candidate successor network. One possible scheme to discovering hidden variables in a Bayesian network is to introduce a disconnected hidden variable H into our current candidate, then use structural EM to “hook it up” to the remaining network.

Specifically, assume that our current candidate BN structure in the structure search contains some set of known variables X_1, \dots, X_n (ones that are observed at least sometimes) and a hidden variable H (one which is never observed). Our current structure G has the X_i 's connected to each other in some way, but the variable H is not connected to any other variable. As usual, SEM completes the data (including for the hidden variable) using the network G , and uses the completed data to evaluate the potential successor BNs. It then greedily chooses the successor network G' that most improves the score (we are using the BIC (equivalently, MDL) score, so there may be no network improving the score). Show that, in the chosen successor G' , H will still necessarily be disconnected from the X_i 's.

4. **Causality [20 points]** For probabilistic queries, we have that

$$\min_x P(y | x) \leq P(y) \leq \max_x P(y | x).$$

Show that the same property does not hold for intervention queries. Specifically, provide an example where it is not the case that:

$$\min_x P(y | do(x)) \leq P(y) \leq \max_x P(y | do(x)).$$