

Learning Undirected Models with Missing Data

Sargur Srihari
srihari@cedar.buffalo.edu

Topics

- Log-linear form of Markov Network
- The missing data parameter estimation problem
- Methods for missing data:
 1. Gradient Ascent
 - Log-likelihood for missing data
 - Expression for Gradient
 - Cost of gradient ascent
 2. EM
 - In E-step we compute for each f_i a sufficient statistic
 - In M-step we run inference multiple times
- Trade-offs between the two

Log-linear form of Markov Network

- Log-linear form of MN with parameters θ is

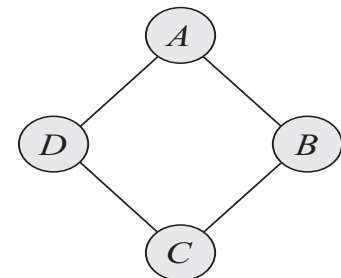
$$P(X_1, \dots, X_n; \theta) = \frac{1}{Z(\theta)} \exp \left\{ \sum_{i=1}^k \theta_i f_i(D_i) \right\}$$

where $\theta = \{ \theta_1, \dots, \theta_k \}$ are k parameters, each associated with a feature f_i defined over instances of D_i

Where the partition function defined as:

$$Z(\theta) = \sum_{\xi} \exp \left\{ \sum_i \theta_i f_i(\xi) \right\}$$

- Features are typically formed of indicator functions, e.g., $f_{a^0 b^0}(a, b) = I\{a = a^0\} I\{b = b^0\}$



The missing data problem

- How to use data when some data fields are missing, e.g.,
 - Some data fields omitted or not collected
 - Some hidden variables
- A simple approach is to “fill-in” the missing values arbitrarily
 - Default values, say false
 - Randomly choose a value
 - They are called data imputation methods
 - Problem is that they introduce “bias”

The “chicken and egg problem”

- We are trying to solve two problems at once
 - Learning the parameters
 - Hypothesizing values of unobserved variables
- Given complete data we can estimate parameters using MLE formulas
- Given a choice of parameters we can infer likely values for unobserved variables
- Since we have neither, problem is difficult

Expectation Maximization approach

- EM solves this problem by bootstrapping
 - Start with some arbitrary starting point
 - Either choice of parameters or initial assignment of hidden variables
 - These assignments are either random or selected using a heuristic approach
- Assume we start with parameter assignment
- The algorithm then repeats the two steps
 - Use current parameters to complete the data using probabilistic inference
 - Then treat the completed data as if it were observed and learn a new set of parameters

Methods for Parameter Estimation with Missing Data

- We look at estimating θ from data \mathcal{D}
- Difficulties in Learning problem:
 - Parameters θ may not be identifiable
 - Coupling between different parameters
 - Likelihood is not concave (has local maxima)
- Two alternative methods
 1. Gradient Ascent (assume missing data is random)
 2. Expectation-Maximization

Gradient Ascent Method for Missing Data:

- Assume data is missing at random in data \mathcal{D}
- In the m^{th} instance, let $\mathbf{o}[m]$ be observed entries and $\mathcal{H}[m]$ random variables that are missing entries in that instance
 - So that for any $\mathbf{h}[m] \in \text{Val}(\mathcal{H}[m])$,
 - $(\mathbf{o}[m], \mathbf{h}[m])$ is a complete assignment to χ

Log-likelihood for Missing Data:

- The average log-likelihood has the form

$$\frac{1}{M} \ln P(D | \theta) = \frac{1}{M} \sum_{m=1}^M \ln \left(\sum_{\mathbf{h}^{(m)}} P(\mathbf{o}^{(m)}, \mathbf{h}^{(m)} | \theta) \right)$$

$$= \frac{1}{M} \sum_{m=1}^M \ln \left(\sum_{\mathbf{h}^{(m)}} \tilde{P}(\mathbf{o}^{(m)}, \mathbf{h}^{(m)} | \theta) \right) - \ln Z$$

where the partition function is explicit and P is replaced by its unnormalized form

- Now consider a single term within the sum

$$\sum_{\mathbf{h}^{(m)}} \tilde{P}(\mathbf{o}^{(m)}, \mathbf{h}^{(m)} | \theta)$$

- This has the same form as a partition function;
- it is precisely the partition function for the MN we would obtain by reducing the original MN with the observation $\mathbf{o}^{(m)}$ to obtain the conditional distribution $P(\mathcal{H}^{(m)} | \mathbf{o}^{(m)})$

Expression for Gradient

- Since $\sum_{\mathbf{h}[m]} \tilde{P}(\mathbf{o}[m], \mathbf{h}[m] | \boldsymbol{\theta})$ has form of partition function, we can apply the proposition $\frac{\partial}{\partial \theta_i} \ln Z(\boldsymbol{\theta}) = E_{\boldsymbol{\theta}}[f_i]$ and conclude that $\frac{\partial}{\partial \theta_i} \ln \sum_{\mathbf{h}[m]} P(\mathbf{o}[m], \mathbf{h}[m] | \boldsymbol{\theta}) = E_{\mathbf{h}[m] \sim P(\mathbf{h}[m] | \mathbf{o}[m], \boldsymbol{\theta})} [f_i]$ i.e., gradient of this term is the conditional expectation of the feature given the observations of this instance. Thus:
- Proposition: For a data set \mathcal{D}

$$\frac{\partial}{\partial \theta_i} \frac{1}{M} \ell(\boldsymbol{\theta} : \mathcal{D}) = \frac{1}{M} \left[\sum_{m=1}^M E_{\mathbf{h}[m] \sim P(\mathbf{h}[m] | \mathbf{o}[m], \boldsymbol{\theta})} [f_i] \right] - E_{\boldsymbol{\theta}} [f_i]$$
 - i.e., gradient for feature f_i with missing data is the difference between two expectations
 - Expectation over the *data and hidden variables* minus the feature expectation over all variables

Gradient Ascent Complexity: Full vs Missing

1. With full data, gradient of log-likelihood is

$$\frac{\partial}{\partial \theta_i} \frac{1}{M} \ell(\boldsymbol{\theta} : D) = E_D[f_i(\chi)] - E_{\boldsymbol{\theta}}[f_i]$$

- For second term we need inference over current distribution $P(\chi | \boldsymbol{\theta})$
- First term is aggregate over data \mathcal{D} .

2. With missing data we have

$$\frac{\partial}{\partial \theta_i} \frac{1}{M} \ell(\boldsymbol{\theta} : D) = \frac{1}{M} \left[\sum_{m=1}^M E_{h[m] \sim P(h[m] | o[m], \boldsymbol{\theta})} [f_i] \right] - E_{\boldsymbol{\theta}}[f_i]$$

- We have to run inference separately for every instance m conditioning on $o[m]$
- Cost is much higher than with full data

EM for MN: Missing Data Param Estimation

- As for any probabilistic model an alternative method for parameter estimation in context of missing data is via Expectation Maximization
 - E step: use current parameters to estimate missing values
 - M step is used to re-estimate the parameters
- For BN it has significant advantages
 1. Can we define a variant of EM for MNs?
 2. Does it have the same benefits?

EM for MN parameter learning

- E-step

- Use current parameters θ^t to compute expected sufficient statistics, i.e., expected feature counts
 - At iteration t expected sufficient statistic for feature f_i is

$$M_{\theta^{(t)}}[f_i] = \frac{1}{M} \left[E_{h[m] \sim P(H[m] \mid o[m], \theta)}[f_i] \right]$$

- M-step

- Critical difference: EM for BNs has closed form
- EM for MN requires running inference multiple times, once for each iteration of gradient ascent
 - At step k of this “inner loop” optimization, we have a gradient of the form

$$M_{\theta^{(t)}}[f_i] - E_{\theta^{(t,k)}}[f_i]$$