10-708: Probabilistic Graphical Models 10-708, Spring 2014

24: Spectral Algorithms for Graphical Models

Lecturer: Eric P. Xing *Scribes: Yuan Xie, Yulong Pei, Junier Oliva*

1 Introduction

Modern machine learning tasks often deal with high-dimensional data. One typically makes some assumption on structure, like sparsity, to make learning tractable over high-dimensional instances. Another common assumption on structure is that of latent variables in the generative model. In latent variable models, one attempts to perform inference not only on observed variables, but also on unobserved latent variable.

Although such models lend a great deal of flexibility and power, it is often the case that exact inference is intractable. Thus, one typically employs approximate inference methods like Markov chain Monte Carlo (MCMC) sampling or heuristic methods like Expectation Maximization (EM). Such methods have several drawbacks, however. For example, EM can get stuck in local minima, and can be slow to converge. Furthermore, there are no statistical guarantees about the quality of EM estimates in general. Spectral methods attempt to mitigate EM's short-comings in several models.

Spectral methods attempt only to estimate probabilities over the observable variables. While some applications may be interested in latent variables themselves, many applications, like forward prediction in sequence models, only deal with observed variables; hence, spectral methods provide useful estimates in a wide range of domains and models. Spectral methods reparameterize the model in such a way that the probabilities of observable variables depend only on statistics over the observable variables themselves.

By avoiding latent variables, spectral methods can bypass difficult optimizations. Indeed, as one will see, spectral methods do not explicitly solve any optimization problem per se. Hence, by making some tradeoffs, one may develop efficient, consistent, and local-minima free estimation procedures using spectral methods.

First, we explore the close connection between latent variables, independence, and rank.

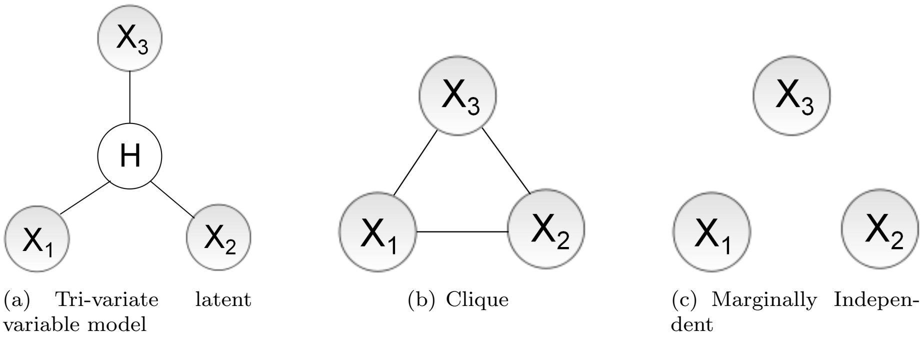
2 Graphical Models and Rank

As previously mentioned, spectral methods attempt to reparameterize a latent variable model in such a way that yields a direct observable model. The obvious way of doing this is via mariginalization of latent variables. For example, consider mariginalizing a latent variable H that is at the center of a star graph (Figure 1(a)). This amounts to the following summation:

$$\Pr[X] = \sum_h \Pr[X_1|H=h]\Pr[X_2|H=h]\Pr[X_3|H=h].$$

Thus, one can see that the sum does not factor and one is left with a full clique on the observed variables (Figure 1(b)).

However, if the latent variable H has exactly one state $H = h_0$ then one no longer has a clique. In fact, our



(a) Tri-variate latent variable model (b) Clique (c) Marginally Independent

Figure 1: Latent variable model and resulting observable model depending on H.

observed variables would be marginally independent (Figure 1(c)). This is because:

$$\begin{aligned}\Pr[X] &= \sum_h \Pr[X_1|H=h]\Pr[X_2|H=h]\Pr[X_3|H=h] \\ &= \Pr[X_1|H=h_0]\Pr[X_2|H=h_0]\Pr[X_3|H=h_0] \\ &= \Pr[X_1]\Pr[X_2]\Pr[X_3].\end{aligned}$$

In terms of conditional probability tables (CPTs), it is clear to see that having exactly one hidden state greatly simplifies things, since such a model would only have $3M$ parameters (if observable variables could each take M different values). On the other hand, if H had M^3 different states, then we are doing no better than a full clique, since a full CPT would have M^3 parameters.

These simple observations about latent variable models beg the question: ”Is there some structure that can be assumed that is somewhere between a marginally independent model and a full clique?” To see how spectral assumptions can provide an answer we explore the relationship between rank and independence.

2.1 Independence and Rank

Let us begin by examining the joint distribution of two random variables X_1, X_2 each taking values in $\{1, \ldots, M\}$. The joint probability of X_1, X_2 can be fully represented as $M \times M$ matrix $\mathcal{P}$ with elements $\mathcal{P}_{ij} = \Pr[X_1 = i, X_2 = j]$. Furthermore, let $\vec{P}[X_1]$ be the $M \times 1$ vector of marginal probabilities for X_1; i.e. $\vec{P}[X_1]_i = \Pr[X_1 = i]$. Likewise, let $\vec{P}[X_2]$ be the vector of marginal probabilities for X_2. An interesting thing occurs if X_1, X_2 are marginally independent:

$$\begin{aligned}\mathcal{P}_{ij} = \Pr[X_1 = i, X_2 = j] &= \Pr[X_1 = i, X_2 = j] \\ &= \Pr[X_1 = i]\Pr[X_2 = j] \\ &= \vec{P}[X_1]_i \vec{P}[X_2]_j.\end{aligned}$$

Thus, we have:

$$\mathcal{P} = \vec{P}[X_1]\vec{P}[X_2]^T,$$

that is, $\mathcal{P}$ is a rank one matrix when X_1, X_2 are marginally independent.

Let us revisit our simple latent variable model, this time over just the two observed variables X_1, X_2 (Figure 2). Suppose that H takes values in $\{1, \ldots, m\}$. Let $\vec{P}[H]$ be the $m \times 1$ vector of marginal probabilities of H: $\vec{P}[H]_k = \Pr[H = k]$. Moreover, define $\vec{P}[X_1|H = k]$ to be the $M \times 1$ vector of marginal probabilities of

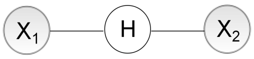


Figure 2: Bivariate latent variable model.

X_1 conditioned on $H = k$: $\vec{P}[X_1|H = k]_i = \Pr[X_1 = i|H = k]$; similarly for $\vec{P}[X_2|H = k]$. Now, our joint probability matrix would be such that:

$$\begin{aligned}
\mathcal{P}_{ij} &= \Pr[X_1 = i, X_2 = j] \\
&= \sum_{k=1}^{m} \vec{P}[H]_k \Pr[X_1 = i, X_2 = j|H = k] \\
&= \sum_{k=1}^{m} \vec{P}[H]_k \Pr[X_1 = i|H = k] Pr[X_2 = j|H = k] \\
&= \sum_{k=1}^{m} \vec{P}[H]_k \vec{P}[X_1|H = k]_i \vec{P}[X_2|H = k]_j .
\end{aligned}$$

Thus, we have:

$$\mathcal{P} = \sum_{k=1}^{m} \vec{P}[H]_k \vec{P}[X_1|H = k] \vec{P}[X_2|H = k]^T,$$

that is, $\mathcal{P}$ is a rank m matrix. Note that this is stemming directly from the fact that our model is the mixture of m marginally independent models: $\Pr[X_1, X_2] = \sum_{k=1}^{m} \Pr[H = k] \Pr[X_1|H = k] Pr[X_2|H = k]$. Using matrix notation:

$$\mathcal{P}(X_1, X_2) = \mathcal{P}(X_1|H) \times \mathcal{P}(\oslash H) \times \mathcal{P}(X_2|H)^T$$

where $\mathcal{P}(\oslash H) = \text{diag}(\vec{P}[H])$, $\mathcal{P}(X_1|H) = [\vec{P}[X_1|H = 1] \cdots \vec{P}[X_1|H = m]]$, and $\mathcal{P}(X_2|H) = [\vec{P}[X_2|H = 1] \cdots \vec{P}[X_2|H = m]]$.

2.2 HMM Example

In this section, we use HMM as the example and the graphical modes for HMM is shown in Figure 3.

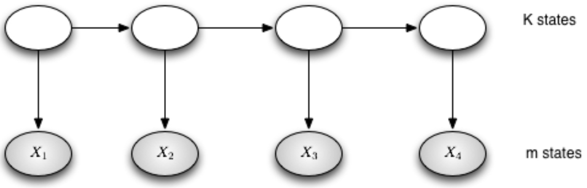


Figure 3: Graphical model for HMM.

We know that low rank matrix can be factorized in the following way:

$$\underset{m\ by\ n}{M} = \underset{m\ by\ k}{L} \times \underset{k\ by\ n}{R} --------\text{if M has rank k}$$

In this case, one factorization would be:

$$P[X_{1,2}, X_{3,4}] = P[X_{1,2}|H]P[H]P[X_{3,4}|H]^T$$

We can notice that this factorization is not unique: by inserting invertible transformations in between, we could have many different factorizations while the insight of spectral learning is that there exists an factorization that only depends on observed variables.

$$\underset{m\ by\ n}{M} = \underset{m\ by\ k}{L} \times S \times S^{-1} \underset{k\ by\ n}{R} - - - - - - - - - \text{if M has rank k}$$

So in our case, we can rewrite the factorization to obtain the following form:

$$\underset{\text{factor of four variables}}{P[X_{1,2}, X_{3,4}]} = \underset{\text{factor of three variables}}{P[X_{1,2}, X_3]} \underset{\text{factor of three variables}}{P[X_2, X_3]^{-1}P[X_2, X_{3,4}]}$$

- One salient advantage of this form is that all factors can be directly computed without using EM since they only involve observed data.
- One potential drawback is that some factors may no longer be probability tables, thus not necessarily non-negative.

Also from the above factorization we can see that it can reduce the size of factors. Along with this, there is another nice property: Every latent variable tree of V variables will have such a factorization where :

- All factors are of size 3
- All factors are only functions of observed data.

As to the learning of this new factorization, we compute the ML estimate for each factor during training and in testing we compute the joint probability with evidence below.

$$\hat{P}[x_1, x_2, x_3, x_4] = P_{MLE}[x_{1,2}, X_3]P_{MLE}[X_2, X_3]^{-1}P_{MLE}[X_2, x_{3,4}]^T$$

where lower cases stand for evidence.

This estimator is consistent and computationally tractable compared to EM but suffers a loss of statistical efficiency due to the dependence on the inverse. This approach could be generalized to situations where you have more variables.

2.3 Spectral Learning with Features

A general requirement that we can see from the factorization above is that all the factors(matrices) must be invertible or have full ranks. Even if some factors are not of full ranks, we can easily fix it by singular vectors decomposition. Another difficulty for this type of factorization is that the intermediate factor may has the size that is much larger than other factors. That is, $k \gg m$. Intuitively, this means long range dependencies.

Previously we only use marginals of pairs or triples of variables to construct the full marginal among the observed variables. So it only works when the size of the intermediate factor is smaller. That is $k \leq m$. But now we need to capture long range dependencies and we do this by introducing long range features. We can user more complex feature, such as $\mathbb{E}[\delta_L \otimes \delta_R]$, instead of the original variables, then we have

$$\begin{aligned} \mathcal{P}[X_{1,2}, X_{3,4}] &= \mathbb{E}[\delta_{1\otimes 2}, \delta_{3\otimes 4}] \\ &= \mathbb{E}[\delta_{1\otimes 2}, \phi_R]V(U^T\mathbb{E}[\delta_L \otimes \delta_R]V)^{-1}U^T\mathcal{P}[\phi_L, X_{3,4}] \end{aligned}$$

Therefore, features can be incorporated into spectral learning.

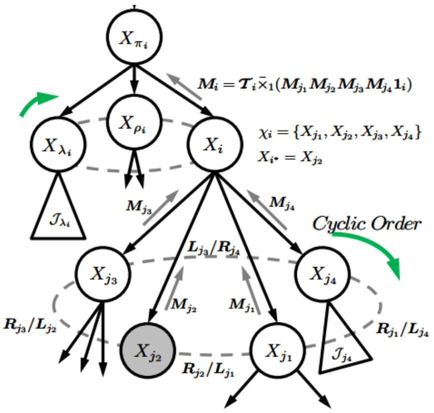


Figure 4: Graphical model for latent tree models.

2.4 Continuous Variables and Hilbert Space Embeddings

It is difficult for EM methods to deal with nonparametric continuous variables. However, by combining with Hilbert Space Embeddings, spectral methods can generalized to learn nonparametric latent models. As mentioned above, using more complex features to take place of variables, features can be incorporated into spectral methods. Moreover, we can introduce infinite dimensional features instead. In particular, we replace

$$\mathcal{P}[X_2, X_3] = \mathbb{E}[\delta_2 \otimes \delta 3] = \mathbb{E}[\delta_2 \delta 3^T]$$

with

$$\mathcal{C}[X_2; X_3] = \mathbb{E}[\phi_2 \otimes \phi 3]$$

Therefore, for discrete case, we have

$$\mathcal{P}[X_{1,2}, X_{3,4}] = \mathcal{P}[X_{1,2}, X_3] V (U^T \mathcal{P}[X_2, X_3] V)^{-1} U^T \mathcal{P}[X_2, X_{3,4}]$$

and for continuous case, we have

$$\mathcal{C}[X_{1,2}; X_{3,4}] = \mathcal{C}[X_{1,2}; X_3] V (U^T \mathcal{C}[X_2, X_3] V)^{-1} U^T \mathcal{C}[X_2; X_{3,4}]$$

3 Applications

3.1 Latent Tree Graphical Models

Parameter learning algorithms for latent variable models have predominantly relied on local search heuristics such as EM methods. In the latent variable models with arbitrary tree topologies where the number of hidden states is smaller than or equal to that of the observed states, a spectral learning method is proposed. The joint distribution of the observed variables is directly compute without explicitly recovering the model parameters. The graphical model for latent tree models is shown in Figure 4 (This figure is from the original paper *A Spectral Algorithm for Latent Tree Graphical Models*). The experiments demonstrate the spectral method can achieve comparable results to EM but are more efficient in both synthetic and real datasets.

3.2 Latent-Variable PCFGs

Latent variable HMM is widely used in sequence labeling and latent-variable PCFG is one example. In order to avoid local optima in EM methods and give consistent parameter estimates, a spectral learning method is proposed. The parameter estimation algorithm, which is simple and efficient, consists of two steps:

- The first step is to take an SVD of the training examples, followed by a projection of the training examples down to a low-dimensional space.
- The second step is calculate empirical averages on the training example which is followed by standard matrix operations.

The experimental studies show the efficiency and effectiveness of the spectral method in latent variable PCFGs.

4 Summary

In this section, we summarize the advantages and disadvantages of EM methods and spectral methods.

- From the perspective of the aim, EM aims to find maximum likelihood estimation (MLE) so it is more statistically efficient and spectral method does not aim to find MLE and therefore the middle variables in general have no statistical meaning.
- For the solution, EM may get stuck in local-optima while spectral methods are local-optima-free.
- The EM methods are fast while spectral methods are slow.
- There are no theoretical guarantees for EM in consistency and it is probably consistent for the spectral method.
- It is easy to extend EM methods to derive for new models, but deriving for new models from spectral methods is challenging and it is unknown whether it can generalized to arbitrary loopy models.
- In EM methods, there are no issues with negative numbers, while spectral methods may encounter problems with negative numbers in matrix inverse.
- It is difficult to incorporate long-range features in EM due to the treewidth increase while incorporating long-range features into spectral methods is easy.
- Dealing with non-Gaussian continuous variables is difficult in EM but spectral methods can generalize easily to non-Gaussian continuous variables via Hilbert Space Embeddings.