10-708: Probabilistic Graphical Models 10-708, Spring 2016

Lecture 21: Spectral Learning for Graphical Models

Lecturer: Eric P. Xing *Scribes: Maruan Al-Shedivat, Wei-Cheng Chang, Frederick Liu*

1 Motivation

In modern machine learning, latent variables are often introduced into the models to endow them with learnable and interpretable structures. Examples of such models include various state space models of sequential data (such as hidden Markov models), mixed membership models (such as topic models), and stochastic grammars (such as probabilistic context free grammars) used to model grammatical structure of sentence or the structure of RNA sequences. Despite the flexibility of such models, the latent structure often complicates inference and learning (exact inference becomes statistically and computationally intractable) and forces one to employ approximate methods such as Expectation Maximization (EM) type of algorithms. The main drawbacks of EM are slow convergence and tendency to get stuck at local minima.

Spectral methods for learning latent variable models attempt to mitigate these issues by representing the latent structures implicitly through the spectral properties of different statistics of the observable variables. While some applications may require explicit representation of the latent variables (when the goal is to *interpret* the data), many applications are interested in *prediction*, and hence use latent structures merely as reasonable (often, domain-specific) constraints. Examples of such applications include forward prediction in various sequence models (Figure 1). In such case, spectral methods offer a powerful toolbox of techniques based on linear algebraic properties of the statistics of models with latent structures. These techniques not only have appealing theoretical properties, such as global consistency and *provable guarantees* on convergence to the *global optimum*, but often work orders of magnitude faster than EM-type algorithms in practice.

In this set of notes, we introduce and discuss the main ideas behind spectral learning techniques for discrete latent variable models. Our working examples are the mixture model and the hidden Markov model (HMM).

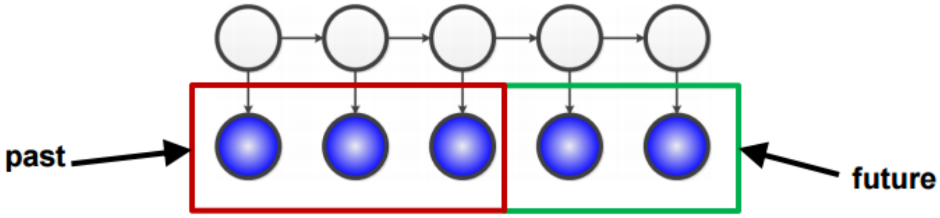

Figure 1: Prediction of the future observations given the past of a dynamical system using a state space model (SSM) or a hidden Markov model (HMM).

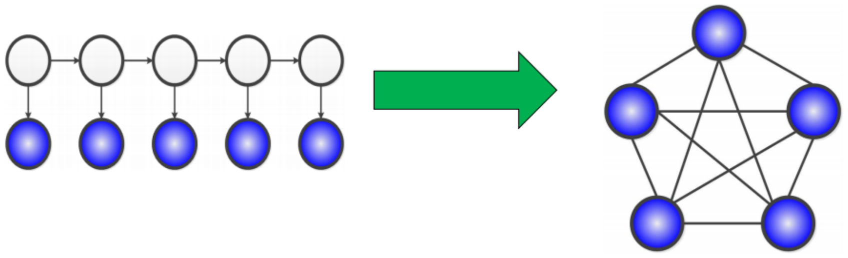

Figure 2: Marginalization of latent variables leads to fully connected graphs.

2 Latent Structure and Low Rank Factorization

Having motivated the main ideas behind spectral methods, we start by considering the following question. As mentioned, we do not care about the latent structure in prediction tasks explicitly. However, consider an HMM; if we marginalize out the latent variables of the model, we certainly arrive at a complete graph (Figure 2). Is there any difference between initially starting with a complete graphical model (i.e., encoding no structural assumptions) and the model we get via marginalization of an HMM? Even though all the observable variables become correlated after marginalization, the *structure of the correlations* between the variables is, in fact, controlled by the latent factors of the original model. Hence, there is a difference, and, as we will see, it manifests in the low rank structure of the joint probability distribution over the observables.

2.1 Mixture model example

To gain more intuition, consider an example of a mixture model with three discrete observable variables X_1, X_2, X_3, where $X_i \in [m] := \{1, \dots, m\}$ and a discrete latent variable $H \in [k]$. Consider the case when $k = 1$, i.e., the model has a single latent state. The joint probability distribution has the following form:

$$P(X_1, X_2, X_3) = P(H = 1) \prod_{j=1}^{3} P(X_j \mid H = 1) = \prod_{j=1}^{3} P(X_j \mid H = 1) \tag{1}$$

where, since there is a single latent state, $P(H = 1) = 1$, and the distribution factorizes over the observable variables, which means they are all independent (Figure 3, left).

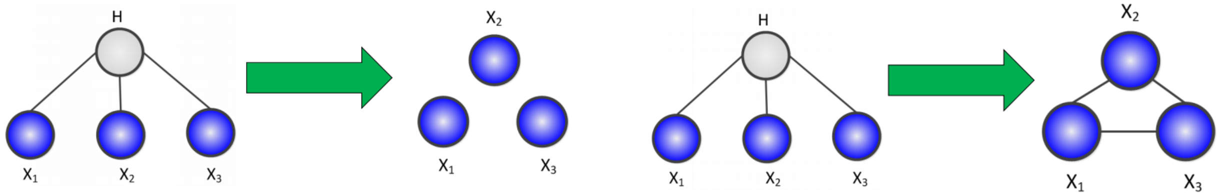

Figure 3: Depending on the number of latent states, a mixture model can be equivalent to a model with independent random variables (left) or to a fully connected graph (right).

Now, consider $k = m^3$, i.e., the number of latent states coincides with the number of all possible configurations of the values taken by X_1, X_2, X_3. In such case, the number of parameters in the mixture model is sufficient to encode any discrete distribution over X_1, X_2, X_3, and hence the model is equivalent to a complete connected graph (i.e., the structural assumptions are effectively void). What happens when the number of latent states is at neither of these two extremes?

2.2 Independence and Rank

To answer the stated question, first, consider the sum rule and the chain rule from purely algebraic perspective. Let $A \in [m], B \in [n]$ are discrete random variables. The sum rule can be represented as simply matrix to vector multiplication:

$$P(A) = \sum_b P(A \mid B = b)P(B = b) = P(A \mid B) \cdot P(B), \tag{2}$$

where $P(A \mid B)$ is an $m \times n$ matrix of conditional probabilities and $P(B)$ is a vector of size n. The chain rule can be represented in a similar form:

$$P(A, B) = P(A \mid B)P(B) = P(A \mid B) \cdot P(\oslash B), \tag{3}$$

where $P(\oslash B) := \text{diag}[P(B)]$ denotes a diagonal matrix with $P(B)$ entries on the diagonal. Diagonal is used to keep B from being marginalized out.

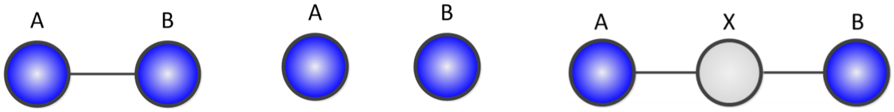

Figure 4: Graphical models for two random variables: arbitrarily dependent (left), completely independent (center), dependent through a latent variable (right).

Now, we are ready to consider the joint distribution of simple two variable graphical model, with two discrete variables, A and B. In general case, A and B are arbitrarily dependent (Figure 4, left) and $P(A, B)$ is an arbitrary table with the only condition that its entries sum up to 1. If A and B are independent (Figure 4, center), then $P(A = a, B = b) = P(A = a)P(B = b)$, and hence the joint probability table can be represented as an outer product of $P(A)$ and $P(B)$ vectors:

$$P(A, B) = P(A) \otimes P(B), \tag{4}$$

which means that $P(A, B)$ is of rank 1. If we introduce a latent variable, X, with $k \leq \min\{m, n\}$ between A and B (Figure 4, right), we can write the joint probability distribution in the following form:

$$P(A, B) = P(A \mid X)P(\oslash X)P(B \mid X)^\top, \tag{5}$$

where we used the introduced algebraic representation of the chain and sum rules. Figure 5 depicts the joint distribution in the form of matrix multiplications. Note that under the condition $k \leq \min\{m, n\}$, $P(A, B)$ is of rank k, i.e., neither full-column, nor full-row rank. Hence, assumptions about certain latent structures result into low rank dependencies between the random variables that can be further exploited by using standard tools from linear algebra: ranks, eigen spaces, singular values decomposition, etc.

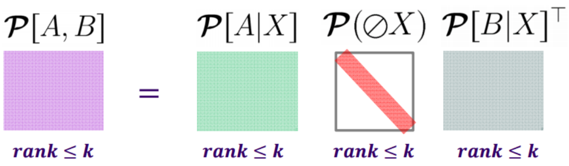
$\boldsymbol{\mathcal{P}}[A, B]$
$=$
$\boldsymbol{\mathcal{P}}[A|X]$
$\boldsymbol{\mathcal{P}}(\oslash X)$
$\boldsymbol{\mathcal{P}}[B|X]^\top$
rank ≤ k
rank ≤ k
rank ≤ k
rank ≤ k

Figure 5: Low rank decomposition of the joint distribution over the observable variables of a model with a latent variable with k states.

3 An Alternate Factorization

In the previous sections, we discover that the joint probability of two random variables could be written into low rank factorization such as

$$M = LR, \tag{6}$$

assuming M has rank k. However, it's well known that factorization is not unique. By multiplying a rotation matrix and its inverse, we have

$$M = LSS^{-1}R, \tag{7}$$

. An interesting question is that could we have an factorization that only depends on observed variables ? To see this, let us continue with the HMM example. We want to factorize a matrix of 4 variables

$$P[X_{1,2}, X_{3,4}] \tag{8}$$

such that the factorization matrices only contain at most three observed variables. First we factorize the following two matrix based on the formula (5),

$$P[X_{1,2}, X_3] = P[X_{1,2}|H_2]P[\oslash H_2]P[X_3|H_2]^T \tag{9}$$

$$P[X_2, X_{3,4}] = P[X_2|H_2]P[\oslash H_2]P[X_{3,4}|H_2]^T \tag{10}$$

By multiplying equation (9) and (10), we obtain

$$P[X_{1,2}, X_3]P[X_2, X_{3,4}] = P[X_{1,2}, X_{3,4}]P[X_2, X_3],$$

and finally,

$$P[X_{1,2}, X_{3,4}] = P[X_{1,2}, X_3]P[X_2, X_3]^{-1}P[X_2, X_{3,4}] \tag{11}$$

Thus we have an alternate factorization that depends only on the observed variables (No H_i), which we hereby referred as observable factorization. This means that these factors can be directly computed from the observed data without the EM algorithm. An example of this procedure is illustrated in Figure 6. As mentioned above, there's no unique factorization, so in practice one could combine both factorization to obtain a better empirically stable counting estimator of the joint probability.

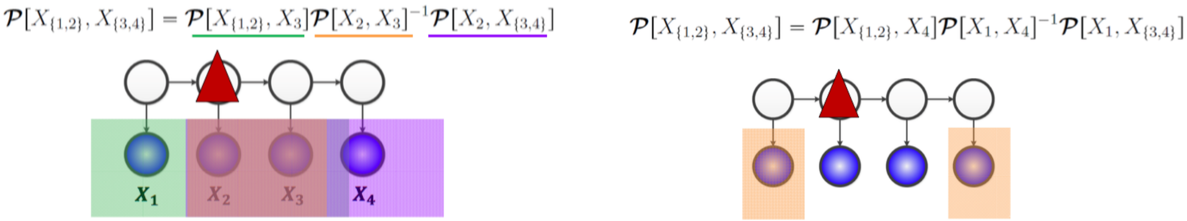
$\mathcal{P}[X_{\{1,2\}}, X_{\{3,4\}}] = \underline{\mathcal{P}[X_{\{1,2\}}, X_3]}\,\underline{\mathcal{P}[X_2, X_3]^{-1}}\,\underline{\mathcal{P}[X_2, X_{\{3,4\}}]}$
X_1
X_2
X_3
X_4
$\mathcal{P}[X_{\{1,2\}}, X_{\{3,4\}}] = \mathcal{P}[X_{\{1,2\}}, X_4]\mathcal{P}[X_1, X_4]^{-1}\mathcal{P}[X_1, X_{\{3,4\}}]$

Figure 6: Two ways of doing observable factorization for the same joint distribution.

At this point, it may not be very encouraging since we have only reduced the joint distribution of four random variables to factor of three random variables multiplications. Nonetheless, the amazing part of the observable factorization is that every latent tree of V variables could be recursively applying this factorization technique to an extend that all factors are of size 3 and that all factors are only functions of observed variables.

4 Training, Testing, and Consistency

4.1 Training and Testing

In training, we replace each probability matrix with its MLE and get

$$P_{MLE}[X_{1,2}, X_3], P_{MLE}[X_2, X_3]^{-1}, P_{MLE}[X_2, X_{3,4}]. \tag{12}$$

For the discrete case, the MLE matrices correspond to frequency counts.In test time we replace variables with certain values and inference is just the look up table.

4.2 Consistency

It is well known that the maximum likelihood estimator (MLE) is consistent for the true joint probability. However, simply estimate the big probability table from the data is not very statistically efficient. An alternative is to first factorize the joint probability into smaller pieces according to the graphical model latent structure, and estimate those small tables based on the EM algorithms. Nevertheless, running EM algorithm may suffer from getting stuck in local optima and thus is not guaranteed to obtain the MLE of the factorized model.

In spectral learning, we could estimate the joint probability by the observable factorization, which is

$$P_{MLE}[X_{1,2}, X_3 P_{MLE}[X_2, X_3]^{-1} P_{MLE}[X_2, X_{3,4}] \longrightarrow P[X_1, X_2; X_3, X_4].$$

In this way, it enjoy the consistency property and is computationally tractable. The only issue now turns to finding the inverse of the probability matrix.

5 The Existence of Inverse

We now look at the conditions for the inverse $P[X_2, X_3]^{-1}$ to be well defined.

$$P[X_2, X_3] = P[X_2|H_2]P[\oslash H_2]P[X_3|H_2]^T \tag{13}$$

All the matrices on the right hand side must be full rank. We will discuss the following cases where $k \neq m$. Where k can be regarded as the number of latent states and m as the number of observe states.

5.1 m > k

The inverse does not exist for this case. However, this can be solved easily by projecting the matrix to a lower dimensional space.

$$P[X_2, X_3]^{-1} = V(U^T P[X_2, X_3]V)^{-1}U^T \tag{14}$$

where U, V are the top left/right k singular vectors of $P[X_2, X_3]$

5.2 k < m

The inverse does not exist for this case either. This is difficult to fix and intuitively corresponds to how the problem becomes intractable if $k >> m$. The case can be interpreted as the number of observed states are not powerful enough to express the relationship. For example, frequency counting does not capture the relationship and some information is missing from just counting the frequency. Intuitively, large k, small m means long range dependencies. We try to solve this with long range features discussed in the next section.

6 Empirical Results with Spectral Learning for Latent Probabilistic Context Free Grammars

The F1 measure in [Cohen et al. 2013] did not show great improvement. However, the run time of the algorithm reduces to $\frac{1}{20}$.

7 Spectral Learning With Features

By using more complex feature, such as $E[\phi_L \otimes \phi_R]$ to represent the original variables, $P[X_2, X_3] = E[\delta_2 \otimes \delta_3]$. We are able to solve the case where $k << m$ because $E[\phi_L \otimes \phi_R]$ is no longer constraint as an $n \times n$ matrix.

8 Summary

Experimentally, the spectral method performs comparatively to the EM algorithm but is much faster. In this section, we summarize the advantages and disadvantages of EM methods and spectral methods.

8.1 Advantages of EM

1. 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.
2. 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.
3. In EM methods, there are no issues with negative numbers, while spectral methods may encounter problems with negative numbers in matrix inverse.

8.2 Advantages of Spectral

1. For the solution, EM may get stuck in local-optima while spectral methods are local-optima-free.
2. There are no theoretical guarantees for EM in consistency and it is probably consistent for the spectral method
3. 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.
4. 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.