CS839: Probabilistic Graphical Models

Second-half

Theo Rekatsinas
What have we seen so far

• Representations
  • Directed GMs
  • Undirected GMs

• Exact Inference
  • Variable Elimination
  • Sum-Product
  • Junction trees

• Learning
  • Parameter learning
  • Structure learning
  • Missing values

• Approximate Inference
  • Variational methods
  • Sampling
Next classes (6 till Thanksgiving + 4 afterwards)

• Advanced Graphical Models
  • Spectral methods for GMs
  • Markov-logic Networks

• Deep learning and GMs
  • Comparison-Overview
  • DL models 1 (VAEs/GANs/domain knowledge in DNNs)
  • DL models 2 (CNNs/RNNs/Attention)

• Scalable Systems
  • Distributed Algorithms for ML
  • Distributed Systems for ML

• Applications
  • Knowledge Base Construction
  • Data Cleaning

• Project presentations
Project Deliverables

- Proposal due: Nov 8
- Mid-report due: Nov 27
- Proposal presentations: Dec 11
CS839: Probabilistic Graphical Models

Lecture 16: Spectral Algorithms for GMs

Theo Rekatsinas
Latent Variable Models

Sequence models

Parsing

Mixed membership models

Ho. et al. 2012

NP³ VP²

NP³ VP²

D¹ N² V¹ P¹

the dog saw him
Latent Parameters (EM)

- Latent variables are not observed in the data: use EM to learn parameters
  - Slow and local minima

\[
P[X_1, ..., X_5, H_1, ..., H_5] = P[H_1] \prod_{i=2}^{5} P[H_i | H_{i-1}] \prod_{i=1}^{5} P[X_i | H_i]
\]
Spectral Learning

• Different paradigm of learning in the presence of latent variables
  • Based on linear algebra

• Theoretically
  • Provably consistent
  • Can offer deep insights into identifiability

• Practically
  • Local minima free
  • Faster than EM: in some cases 10-100x speed-up
References

• Hsu et al. 2009 – Spectral HMMs
• Siddiqi et al. 2009 – Features in Spectral Learning
• Parikh et al. 2011/2012 – Tensors to Generalize to Trees/Low Treewidth Graphs
• Cohen et al. 2012/2013 – Spectral Learning of latent PCFGs
• Song et al. 2013 – Spectral Learning as Hierarchical Tensor Decomposition
Focus on Predictions

• In many applications that use latent variable models, the end task is not to recover the latent states but use the model for prediction among the observed variables.

• Example: predict the future given the past.
Focus on Predictions

• Only use quantities related to the observed variables:

\[ P[X_1, X_2, X_3, X_4, X_5] \]

• Do not care about latent variables explicitly

• Do we still need EM to learn the parameters?
Focus on Predictions

• Why don’t we just integrate them out?
Focus on Predictions

• Why don’t we just integrate them out?
Marginal does not factorize

\[ P[X_1, X_2, X_3, X_4, X_5] = \sum_{H_1, \ldots, H_5} P[H_1]P[H_1] \prod_{i=2}^{5} P[H_i|H_{i-1}] \prod_{i=1}^{5} P[X_i|H_i] \]

- Does not factorize due to the outer sum
HMM and cliques

• Is an HMM different from a clique?
• It depends on the number of latent states!
• Example:
What if H has only one state?
What if $H$ has only one state?

- The observed variables are independent
What if $H$ has only many states?

- If $X_1$, $X_2$, $X_3$ have $m$ states each and $H$ has $m^3$
What if H has only many states?

- If $X_1$, $X_2$, $X_3$ have $m$ states each and $H$ has $m^3$
- The model can be exactly equivalent to a clique
What about cases between 1 and $m^3$?

- Under existing methods, latent models require EM regardless of the number of hidden states

- Is there a formulation of latent variable models where the difficulty of learning is a function of the number of latent states?

- We will answer this by adopting a **spectral view**.
Sum Rule (Matrix Form)

• Sum Rule

\[ P[X] = \sum_Y P[X|Y]P[Y] \]

• Equivalent view using Matrix Algebra

\[
\begin{align*}
P[X] &= P[X|Y] \times P[Y] \\
\begin{pmatrix}
P[X = 0] \\
P[X = 1]
\end{pmatrix} &= \begin{pmatrix}
P[X = 0|Y = 0] & P[X = 0|Y = 1] \\
P[X = 1|Y = 0] & P[X = 1|Y = 1]
\end{pmatrix} \times \begin{pmatrix}
P[Y = 0] \\
P[Y = 1]
\end{pmatrix}
\end{align*}
\]
Chain Rule (Matrix Form)

• Sum Rule

\[
\]

• Equivalent view using Matrix Algebra

\[
P[X, Y] = P[X|Y] \times P[\emptyset Y] = 
\begin{pmatrix}
P[X = 0, Y = 0] & P[X = 0, Y = 1] \\
P[X = 1, Y = 0] & P[X = 1, Y = 1]
\end{pmatrix} = 
\begin{pmatrix}
P[X = 0|Y = 0] & P[X = 0|Y = 1] \\
P[X = 1|Y = 0] & P[X = 1|Y = 1]
\end{pmatrix} \times 
\begin{pmatrix}
P[Y = 0] & 0 \\
0 & P[Y = 1]
\end{pmatrix}
\]
GMs: The linear algebra view

• Is there something we can say about this matrix?

\[ \mathcal{P}[A, B] \]

A and B have m states each.
Independence: The linear algebra view

- What if A and B are independent?

A and B have m states each.
Independence: The linear algebra view

- What can we say about this matrix?
Independence: The linear algebra view

- What can we say about this matrix? **It is rank one**
Independence and Rank

• What about rank in between 1 and m?
Low Rank Structure

• A and B are not marginally independent (conditionally independent given X)

• If X has k states (while A and B have m states):

\[ \text{rank}(\mathcal{P}[A, B]) \leq k \]
Low Rank Structure

\[ \mathcal{P}[A, B] = \mathcal{P}[A|X] \mathcal{P}(\emptyset X) \mathcal{P}[B|X]^T \]

\( \text{rank} \leq k \)

\( \text{rank} \leq k \)

\( \text{rank} \leq k \)

\( \text{rank} \leq k \)
Spectral View

• Latent variable models encode low rank dependencies among variables (both marginal and conditional)

• Use tools from linear algebra to exploit this structure:
  • Rank
  • Eigenvalues
  • SVD
  • Tensors
Example: HMM

\[ \mathcal{P}[X_{\{1,2\}}, X_{\{3,4\}}] \]

\{X_1, X_2\} has rank \( k \)

\{X_3, X_4\}
Low Rank Matrices Factorize

\[ M = LR \]

If \( M \) has rank \( k \)

We already know a factorization (introduced by the graph structure)
Low Rank Matrices Factorize

\[ \mathcal{P}[X_{\{1,2\}}, X_{\{3,4\}}] \]

\[ \mathcal{P}[X_{\{1,2\}}|H_2]\mathcal{P}[\Theta H_2]\mathcal{P}[X_{\{3,4\}}|H_2]^T \]

Is this useful?
Alternate Factorizations

• This factorization is not unique

• Standard Matrix Factorization trick: Add any invertible transformation

\[ M = LR \]
\[ M = LSS^{-1}R \]

• There exists a different factorization that only depends on observed variables!
An Alternate Factorization

• Consider

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

• Let’s factorize it in a product of matrices over three observed variables

• Example:

\[ P[X_{\{1,2\}}, X_3] \]

\[ P[X_2, X_{\{3,4\}}] \]
An Alternate Factorization

• We have:

\[ P[X_{\{1,2\}}, X_3] = P[X_{\{1,2\}} | H_2] P[\ominus H_2] P[X_3 | H_2]^\top \]
\[ P[X_2, X_{\{3,4\}}] = P[X_2 | H_2] P[\ominus H_2] P[X_{\{3,4\}} | H_2]^\top \]

• Product of green terms is:

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

• Product of read terms is:

\[ P[X_2, X_3] \]
An Alternate Factorization

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

- Factors are only function of observed variables: No EM needed!
- Some factors are no longer probability tables
- We call this the \textit{observable factorization}
Graphical Relationship

\[ \mathcal{P}[X_{\{1,2\}, X_{\{3,4\}}}] = \mathcal{P}[X_{\{1,2\}, X_3}] \mathcal{P}[X_2, X_3]^{-1} \mathcal{P}[X_2, X_{\{3,4\}}] \]
What does learning mean here?

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

- We learn only the tables over observed variables
- No need to learn H (No EM)
Another Factorization (not unique)

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

- Some factors are no longer probability tables
- We call this the observable factorization
Relationship to Original Factorization

\[ \mathcal{P}[X_{\{1,2\}}, X_{\{3,4\}}] = \mathcal{P}[X_{\{1,2\}}|H_2] \mathcal{P}[\emptyset|H_2] \mathcal{P}[X_{\{3,4\}}|H_2]^T \]

\[ M = LR \]

\[ M = LSS^{-1}R \]

- What is the **algebraic relationship** between the original factorization and the new factorization?
Relationship to Original Factorization

• Consider:

\[
S := \mathcal{P}[X_3|H_2]
\]

\[
\mathcal{P}[X_{\{1,2\}}, X_{\{3,4\}}] = \mathcal{P}[X_{\{1,2\}}, X_3] \mathcal{P}[X_2, X_3]^{-1} \mathcal{P}[X_2, X_{\{3,4\}}] = LS = S^{-1}R
\]
Alternate Factorization

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

- We reduced the size of the factor by 1 (not very impressive?)
  - We can recursively factorize many GMs
Alternate Factorization

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

- We reduced the size of the factor by 1 (not very impressive?)
  - We can recursively factorize many GMs

- Every latent tree of V variables has such a factorization where:
  - All factors are of size 3
  - All factors are only functions of observed variables
Training/Testing with Spectral Learning

• We have that:

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

• In training we get the MLE of

\[
\mathcal{P}_{MLE}[X_{\{1,2\}}, X_3] \quad \mathcal{P}_{MLE}[X_2, X_3]^{-1} \quad \mathcal{P}_{MLE}[X_2, X_{\{3,4\}}]
\]

• In test time we compute probability estimates

\[
\hat{\mathcal{P}}_{spec}[x_1, x_2, x_3, x_4] = \mathcal{P}_{MLE}[x_{\{1,2\}}, X_3] \mathcal{P}_{MLE}[X_2, X_3]^{-1} \mathcal{P}_{MLE}[X_2, x_{\{3,4\}}]^T
\]
Generalizing to More Variables

• Consider an HMM with 5 observations. We have:

\[
\mathcal{P}[X_{\{1,2\}}, X_{\{3,4,5\}}] = \mathcal{P}[X_{\{1,2\}}, X_3] \mathcal{P}[X_2, X_3]^{-1} \mathcal{P}[X_2, X_{\{3,4,5\}}]
\]

Reshape and decompose recursively

\[
\mathcal{P}[X_{\{2,3\}}, X_{\{4,5\}}] = \mathcal{P}[X_{\{2,3\}}, X_4] \mathcal{P}[X_3, X_4]^{-1} \mathcal{P}[X_3, X_{\{4,5\}}]
\]
Consistency

• Estimate joint distribution
  • It is consistent. We are simply using maximum likelihood estimation

\[ P_{MLE}[X_1, X_2; X_3, X_4] \rightarrow P[X_1, X_2; X_3, X_4] \quad \text{as number of samples increases} \]

• However, it is not very statistically efficient
Consistency

• A better estimate is to compute likelihood estimates of the factorization

\[ P_{MLE}[X_{1,2}|H_2] P_{MLE}[\emptyset H_2] P_{MLE}[X_{3,4}|H_2]^T \]

\[ \rightarrow P[X_1, X_2; X_3, X_4] \]

• But this requires EM
Consistency

• In spectral learning, we estimate the alternate factorization

\[ \mathcal{P}_{MLE}[X_{\{1,2\}}, X_3]\mathcal{P}_{MLE}[X_2, X_3]^{-1}\mathcal{P}_{MLE}[X_2, X_{\{3,4\}}] \rightarrow \mathcal{P}[X_1, X_2; X_3, X_4] \]

• This is consistent and computationally tractable (we lose some statistical efficiency due to the dependence on the inverse)
The Inverse Catch

• Before we had the clique problem: where does this appear in our factorization?

• Utility of hidden variables: Make the model simpler

• How does this manifest in our factorization?

\[
P[X_{\{1,2\}}, X_{\{3,4\}}] = P[X_{\{1,2\}}, X_3]P[X_2, X_3]^{-1}P[X_2, X_{\{3,4\}}]
\]
The Inverse Catch

• Before we had the clique problem: where does this appear in our factorization?

• Utility of hidden variables: Make the model simpler

• How does this manifest in our factorization?

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

When does this exist?
When does the inverse exist?

\[ \mathcal{P}[X_2, X_3] = \mathcal{P}[X_2|H_2] \mathcal{P}[\Theta H_2] \mathcal{P}[X_3|H_2]^\top \]

- All the matrices on the right hand side must have full rank (and square).
- Full rank: All rows and columns are linearly independent
- This is a requirement of spectral learning
- Is this interesting?
When does the inverse exist?

\[ \mathcal{P}[X_2, X_3] = \mathcal{P}[X_2|H_2] \mathcal{P}[\mathcal{O}H_2] \mathcal{P}[X_3|H_2]^\top \]

• All the matrices on the right hand side must have full rank (and square).
• Full rank: All rows and columns are linearly independent
• This is a requirement of spectral learning
• Is this interesting? E.g.: This means that the hidden states in H2 have to be the same as X2

• We benefit only if \( k < m \) (we get a reduction in representation complexity)
• What about \( k > m \)?
When does the inverse exist?

\[
\mathcal{P}[X_2, X_3] = \mathcal{P}[X_2|H_2] \mathcal{P}[\emptyset H_2] \mathcal{P}[X_3|H_2]^{\top}
\]

• All the matrices on the right hand side must have full rank (and square).
• Full rank: All rows and columns are linearly independent
• This is a requirement of spectral learning
• Is this interesting? E.g.: This means that the hidden states in H2 have to be the same as X2

• We benefit only if k < m (we get a reduction in representation complexity)
• What about k > m? Feature extraction: think of deep learning
When $m > k$

- The inverse cannot exist, but we can fix this: project onto a lower dimensional space

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

- $U, V$ are the top left/right $k$ singular vectors of $\mathcal{P}[X_2, X_3]$
When $k > m$

• The inverse does exist. But it no longer satisfies that:

$$\mathcal{P}[X_2, X_3]^{-1} = (\mathcal{P}[X_3|H_2]^\top)^{-1} \mathcal{P}[\otimes H_2]^{-1} \mathcal{P}[X_2|H_2]^{-1}$$

• More difficult to fix and intuitively corresponds to how the problem becomes intractable if $k \gg m$
When $k > m$

- The inverse does exist. But it no longer satisfies that:

\[
\mathcal{P}[X_2, X_3]^{-1} = (\mathcal{P}[X_3|H_2]^{\top})^{-1} \mathcal{P}[\varnothing H_2]^{-1} \mathcal{P}[X_2|H_2]^{-1}
\]

- More difficult to fix and intuitively corresponds to how the problem becomes intractable if $k \gg m$

- Let’s ignore it for now 😊
Spectral Learning in Practice

• We will use marginals of pairs/triples of variables to construct the full marginal among the observed variables.
• Only works when \( k < m \)
• However, we need to capture longer range dependencies
Use of Long-Range Features

Construct feature vector of left side

\[ \phi_L \]

Construct feature vector of right side

\[ \phi_R \]
Spectral Learning with Features

\[ \mathcal{P}[X_2, X_3] = \mathbb{E}[\delta_2 \otimes \delta_3] := \mathbb{E}[\delta_2 \delta_3^T] \]

Rewrite using indicator features \( \delta \)
Spectral Learning with Features

\[ \mathcal{P}[X_2, X_3] = \mathbb{E}[\delta_2 \otimes \delta_3] := \mathbb{E}[\delta_2 \delta_3^\top] \]

Use more complex feature instead:

\[ \mathbb{E}[\phi_L \otimes \phi_R] \]

\[ \mathcal{P}[X_{\{1,2\}}, X_{\{3,4\}}] = \mathbb{E}[\delta_{1\otimes 2}, \delta_{3\otimes 4}] = \mathbb{E}[\delta_{1\otimes 2}, \phi_R] V (U^\top \mathbb{E}[\phi_L \otimes \phi_R] V)^{-1} U^\top \mathcal{P}[\phi_L, X_{\{3,4\}}] \]
Experimentally

- Many results show that spectral methods achieve comparable results to EM but are 10-100x faster
## Summary

<table>
<thead>
<tr>
<th><strong>EM</strong></th>
<th><strong>Spectral</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Aims to find MLE in a statistically efficient manner</td>
<td>• Does not aim to find MLE/less statistically efficient</td>
</tr>
<tr>
<td>• Can get stuck in local-optima</td>
<td>• Local-optima-free</td>
</tr>
<tr>
<td>• Limited theoretical guarantees</td>
<td>• Provably consistent</td>
</tr>
<tr>
<td>• Slow</td>
<td>• Fast</td>
</tr>
<tr>
<td>• Easy to derive for new models</td>
<td>• Challenging to derive for new models (unknown if it generalizes to arbitrary loopy models)</td>
</tr>
</tbody>
</table>