Exact MAP Estimation of Graph Edges with Degree Priors

Bert Huang
with Tony Jebara
Columbia University Machine Learning Laboratory
Outline

• Graph sparsification example
• Generalized problem statement
• Algorithm description
• Algorithm Analysis
Advertisement via b-matching

<table>
<thead>
<tr>
<th>Advertisers</th>
<th>chicken</th>
<th>burger</th>
<th>nutrition</th>
</tr>
</thead>
<tbody>
<tr>
<td>McDonalds</td>
<td>$5</td>
<td>$6</td>
<td>$0</td>
</tr>
<tr>
<td>Burger King</td>
<td>$0</td>
<td>$7</td>
<td>$5</td>
</tr>
<tr>
<td>KFC</td>
<td>$10</td>
<td>$0</td>
<td>$0</td>
</tr>
</tbody>
</table>
Advertisement via b-matching

<table>
<thead>
<tr>
<th>Advertisers</th>
<th>chicken</th>
<th>burger</th>
<th>nutrition</th>
</tr>
</thead>
<tbody>
<tr>
<td>McDonalds</td>
<td>$5</td>
<td>$6</td>
<td>$0</td>
</tr>
<tr>
<td>Burger King</td>
<td>$0</td>
<td>$7</td>
<td>$5</td>
</tr>
<tr>
<td>KFC</td>
<td>$10</td>
<td>$0</td>
<td>$0</td>
</tr>
</tbody>
</table>
Degree-based Graph Sparsification

- Advertisement allocation in search engines: advertisers pay per edge
- Assigning conference reviewers to submitted papers
- Designing (computer/sensor) network topology
Graph Sparsification in Machine Learning

- Useful in Machine Learning for
  - kNN classification
  - $b$-matching spectral clustering [JebShc06]
  - $b$-matching for embedding [ShaJeb07, ShaJeb09]
  - $b$-matching for semi-supervised learning [JebWanCha09]
Generalized Form

\[ \Pr(X|A) \Pr(A) = \prod_{ij} \Pr(x_i, x_j | A_{ij}) \prod_i \Pr(|\tilde{A}_i|) \]

- Singleton edge probabilities
- Degree priors (multinomial)
- log-concave if made continuous
• Since we observe $X$, the $1^\text{st}$ term makes all edges independent

• but the $2^\text{nd}$ term adds dependencies
Conversion to \( b \)-Matching
Conversion to $b$-Matching
Conversion to $b$-Matching

- Add auxiliary nodes
Conversion to $b$-Matching

- Add auxiliary nodes
- Reward/penalize aux. edges
- Constrain original node degrees to N
Conversion to $b$-Matching

- Add auxiliary nodes
- Reward/penalize aux. edges
- Constrain original node degrees to $N$

Degree = 0
6 aux. nodes
Conversion to $b$-Matching

- Add auxiliary nodes
- Reward/penalize aux. edges
- Constrain original node degrees to N

Degree = 1
5 aux. nodes
Conversion to $b$-Matching

- Add auxiliary nodes
- Reward/penalize aux. edges
- Constrain original node degrees to N

Degree = 2
4 aux. nodes
Conversion to $b$-Matching

- Add auxiliary nodes
- Reward/penalize aux. edges
- Constrain original node degrees to $N$

Degree = 3
3 aux. nodes
Auxiliary Edges

- Penalize degrees other than peak $k$
- When deg. = $k$, also $N-k$ aux. edges
- $N-k$ positive-weight aux. edges
- $k$ negative-weight aux. edges

$$j^{\text{th}} \text{ edge} = \psi_i(j - 1) - \psi_i(j)$$
Algorithm Illustration

Augmented Weight Matrix

b-matching

Augmented \( b \)-Matching

MAP Solution
\(b\)-matching via Belief Propagation

- \(b\)-matching can be solved \textbf{quickly} with belief propagation \cite{HuaJeb07}
- Converges in \(O(|E|)\) under mild assumptions \cite{SalSha2009}
- Guaranteed in \(O(|V||E|)\) otherwise
Generalized Problems

• Framework generalizes classical problems by setting different degree priors:
  • kNN – each node must have indegree $k$
  • $b$-matching – each node must have degree $b$
  • $\epsilon$-balls – each edge must have weight at least $\epsilon$
  • MST – each node must have degree at least one and total edges must be $N-1$
Summary

• Graph structure estimation with degree priors efficiently solvable
• Solve by converting soft-degree problem to hard constraint $b$-matching
• Graph sparsification can be viewed as MAP estimation
Bonus Slides
Future Work

- Class of now solvable problems is too rich!
- How do we set edge probabilities and degree priors so they are useful?
- Can’t simply cross-validate over $k, b$
- Consolidate machine learning processes with graph sparsification components
kNN Classification

\[ Pr(Y|X) = Pr(Y|A) \prod_{ij} Pr(x_i, x_j|A_{ij}) \prod_i Pr(|\tilde{A}_i|) \]

- Y is not used when learning A, so kNN is essentially unsupervised.

- Metric learning methods start to address this, but based on intuitions, not statistics.

- Learn function \( Pr(x_i, x_j|A_{ij}) \) as Mahalanobis distance based on an idealized A estimate [WeiTes07, WeiSau09]

- Can we also learn degree priors? Our tool might be able to help.