Community Discovery in Dynamic Networks via non-negative matrix factorization

Niloufar Afsariardchi
Supervisor: Mark Coates
McGill University
March 15, 2011
Outline

- Introduction
- Related Work
- Goals
- Non-negative matrix factorization
- Automatic relevance determination
- Dynamic model
- Experimental results
- Summary and future work
Communities Everywhere...

Why Dynamic Clustering?

- Recommendation Systems
- Detecting anomalies
- Studying dynamic features of social and biological networks

Social Networks: Facebook
Goals

- Soft clustering
- Addressing the problem of unknown number of clusters over time
- Handling changing number of clusters over time
- Temporal smoothness
Learning the parts of objects by non-negative matrix factorization [Lee1999]

\[ V_{ij} \approx (WH)_{ij} = \sum_{k=1}^{K} W_{ik} H_{kj} \]

- **V**: image database (nxm, m facial image each has n pixel)
- **W**: Basis images (nxK)
- **H**: Encoding (Kxm)

NMF for Clustering in Network

\[ V_{ij} \approx (WH)_{ij} = \sum_{k=1}^{K} W_{ik} H_{kj} \]

- **\( V \)**: Matrix of observation
- **\( W_{ik} \)**: Probability of participation of node i in k-th community
- **\( h_{ki} \)**: Probability of individual i to be attracted by k-th community

\[ V \approx WH \]

\[ p(i,j) \approx p(i|c_1)p(j|c_1) + p(i|c_2)p(j|c_2) \]
NMF inference

- MAP estimation
- Maximizing likelihood function with Poisson distribution = minimizing KL-divergence

\[
W^*, H^* = \max p(W, H | V)
\]

coordinate decent

\[
H \leftarrow \left( \frac{H}{W^T} \right) \cdot \left[ W^T \left( \frac{V}{WH} \right) \right]
\]

\[
W \leftarrow \left( \frac{W}{H^T} \right) \cdot \left[ \left( \frac{V}{WH} \right) H^T \right]
\]
Goals

✓ Soft clustering
  - Unknown number of clusters over time
  - Changing number of clusters over time
  - Temporal smoothness
Consider $\beta_k$s as prior of each community and $a, b$ as hyperparameters.

Bayesian NMF Model

\[
p(\beta_k | a_k, b_k) = \frac{b_k^{a_k}}{\Gamma(a_k)} \beta_k^{a_k-1} \exp(-\beta_k b_k)
\]

\[
p(w_{fk} | \beta_k) = \mathcal{H}(w_{fk} | 0, \beta_k^{-1})
\]

\[
p(h_{kn} | \beta_k) = \mathcal{H}(h_{kn} | 0, \beta_k^{-1})
\]

Log priors can be written as:

\[
-\log p(W | \beta) = \sum_k \sum_f \frac{1}{2} \beta_k w_{fk}^2 - \frac{F}{2} \log \beta_k,
\]

\[
-\log p(H | \beta) = \sum_k \sum_n \frac{1}{2} \beta_k h_{kn}^2 - \frac{N}{2} \log \beta_k.
\]
We can rewrite the likelihood function,

\[- \log p(W, H, \beta | V) \overset{c}{=} - \log p(V | W, H) - \log p(W | \beta) - \log p(H | \beta) - \log p(\beta)\]

The objective function we want to minimize,

\[
\min_{W,H,\beta} C_{\text{MAP}}(W, H, \beta) \overset{\Delta}{=} - \log p(W, H, \beta | V)
\]
Goals

✔ Soft clustering
✔ Unknown number of clusters over time
✔ Changing number of clusters over time
  - Temporal smoothness
Dynamic Model

\[
\log p(W_t | W_{t-1}) = \log \frac{1}{B(\psi_t)} \prod w_{ik,t}^{\mu w_{ik,t-1}} \\
= \sum_{ik} \mu w_{ik,t-1} \log w_{ik,t} + c
\]

\[
\log p(H_t | H_{t-1}) = \log \frac{1}{B(\psi'_t)} \prod h_{ki,t}^{\mu h_{ki,t-1}} \\
= \sum_{ki} \mu h_{ki,t-1} \log h_{ki,t} + c
\]
Parameter inference

- Parameter inference using multiplicative update rules
- Point estimation to find the maximum of likelihood function

\[
\log L(W_t, H_t) = \log P(W_t, H_t, \beta_t | V_t) + \log P(W_t | W_{t-1}) + \log P(W_t | W_{t-1})
\]

- Replacing \( \mu = \frac{1 - \alpha}{\alpha} \). We have,

\[
\begin{align*}
H & \leftarrow \frac{\alpha H}{W^T 1_{F \times N} + \text{diag}(\beta)H} \cdot \left[ W^T \left( \frac{V}{WH} \right) \right] + (1 - \alpha)W_{t-1} \\
W & \leftarrow \frac{\alpha W}{1_{F \times N} H^T + W \text{diag}(\beta)} \cdot \left[ \left( \frac{V}{WH} \right) H^T \right] + (1 - \alpha)H_{t-1} \\
\beta & \leftarrow \frac{F+N+2(a-1)}{1_{1 \times F}(W \cdot W) + (H \cdot H)1_{N \times 1} + 2b}
\end{align*}
\]
Experiment 1

- 128 nodes
- 4 clusters of 32 nodes
- Average degree =16
- Zout= 2,3,4
- Alpha=0.9

- Dynamic part: 3 nodes from each cluster leave their original cluster and join other cluster
Experiment 2

- 300 nodes
- Average degree=16
- Alpha=0.9
- Changing number of clusters
Summary and Future work

- Used non-negative matrix factorization for soft clustering
- Introduced automatic relevance determination
- Proposed dynamic model
- Parameter inference
- Showed that DMNF gives competitive results and is capable of detecting changing number of clusters
- How to set alpha? How to avoid local optiums?
Questions?

Thanks!