Some overlapping community detection models
DRAFT

Konstantin Slavnov
Skoltech
k.slavnov@skoltech.ru

Maxim Panov
Skoltech
panov.maxim@gmail.com

Abstract

Basic summary for new overlapping community detection models. Our aim to understand its difference and create own model with logistic loss function. We consider next models and algorithms: MMSB [2], DC-MMSB [8], GeoNMF [10], OCCAM [14], SAAC [9]. There are also "colored edges" [4], AGM (BigCLAM) [12], [13] models existed.

1. TODO:

1. Sample splitting technique CProj algorithm.
2. Add SAAC description

2. Intro

Community detection is the greatest problem ever [6]!

3. Models

MMSB (Mixed membership stochastic blockmodel) [2], description from [10]. One of the basic models in this field. And big amount of other models have the mixed membership stochastic block model or its variations in background.

In this model each node i has a discrete probability distribution $\theta_i = (\theta_{i1}, \dots, \theta_{iK})$ over K clusters, and connections between nodes i and j are generated by first drawing from θ_i and θ_j independently, and then creating a connection based on cluster-specific matrix $B \in [0,1]^{K \times K}$. Specifically,

$$\begin{aligned} c_i &\sim \theta_i, \\ c_j &\sim \theta_j, \\ P(A_{ij} = 1 \,|\, c_i, c_j) &= B_{c_i, c_j}, \end{aligned}$$

where A represents the adjacency matrix of the generated graph. B should have higher values on its diagonal as compared to the off-diagonal, implying that nodes in the same cluster are more likely to form an edge.

In matrix form we can consider the probability matrix of the network P.

$$P = \rho \Theta B \Theta^T, \tag{1}$$

where $\Theta \in \mathbf{R}^{n \times K}$ is the node-community distribution matrix, whose entries Θ_{ij} give the probability that node i is in community j. The row vector $\Theta_i \, (i \in [n])$ is the community membership distribution of node i, and sums to 1. The parameter $B \in [0,1]^{K \times K}$ is the community-community interaction matrix, where B_{ij} is the probability of having a link between node in community i and node in community j. The parameter ρ controls the sparsity of the graph and also needs for theoretical results.

Additionally, for each node $i \in [n]$, we draw $\Theta_i \sim \mathrm{Dir}(\alpha)$, where $\alpha \in \mathrm{R}^K$.

The adjacency matrix of the graph is generated by:

$$A = \mathrm{Bernoulli}(P) = \mathrm{Bernoulli}(\rho \Theta B \Theta^T),$$

GeoNMF (Geometric Non-negative matrix factorization) [10]. There are MMSB model and symmetric non-negative matrix factorization approach in background.

It is tries to infer $W = \sqrt{\rho} \Theta B^{1/2}$ that factorizes $P = WW^T$ from MMSB model (1). $B \geq 0$ — diagonal for theoretical result. Sample splitting technique [11] and k-means clustering are used for algorithm construction. This approach is very close to **OCCAM**'s one. Later we describe intuitive understanding of method.

Algorithm:

1. Randomly split the set of nodes into two equal-sized parts S and $\bar{S}$.
2. Get the rank-K eigen-decompositions $A(S,S) = \hat{V}_1 \hat{E}_1 \hat{V}_1^T$ and $A(\bar{S},\bar{S}) = \hat{V}_2 \hat{E}_2 \hat{V}_2^T$
3. Calculate degree matrices D_{12} and D_{21} for the rows of $A(S,\bar{S})$ and $A(\bar{S},S)$ respectively.

```
4. $\hat{X} = D_{21}^{-1/2} \hat{V}_1 \hat{E}_1^{1/2}$
5. $\mathcal{T} = \left\{ i : \|\hat{X}(i,:)\|_2 \geq \max_j \|\hat{X}(j,:)\|_2 \left(1 - c\sqrt{\frac{\log n}{(n\rho^2)}}\right) \right\}$
6. Run K-means clustering on $\hat{X}(\mathcal{T},:)$, where each row is a point, then pick up only one point from each cluster to construct $S_p$.
7. $\hat{X}_p = \hat{X}(S_p,:)$
8. Get $\hat{B} = \mathrm{diag}(\hat{B}_i)$, $\hat{B}_i = \|e_i^T D_{21}^{1/2}(S_p,S_p)\hat{X}_p\|_F^2$, $i \in [K]$
9. $\hat{\rho} = \max_i \hat{B}_i$
10. $\hat{B} = \hat{B}/\hat{\rho}$
11. $\Theta(\bar{S}) = D_{21}^{1/2} \hat{X} \hat{X}_p^{-1} D_{21}^{-1/2}(S_p,S_p)$
12. Repeat steps with $D_{12}$, $\hat{V}_2$, and $\hat{E}_2$ to obtain $\hat{\Theta}(S)$.
```

Sample splitting technique [11]. This is technique can be applied to many graph problems. Here we describe the approach on graph partitioning task.

Consider simple stochastic block model $\mathcal{G}(\psi, P)$ for graph G. Let $\psi : \{1,\dots,n\} \to \{1,\dots,k\}$ be a partition of n nodes to k classes. Let P be a $k \times k$ matrix, where $P_{ij} \in [0,1]$ for all i,j. Include edge (u,v) with probability $P_{\psi(u)\psi(v)}$. Denote $G_{uv} = P_{\psi(u)\psi(v)}$ and $\hat{G} \sim \mathcal{G}(\psi, P)$ — sampled adjacency matrix.

Now we can formulate **Planted Partition Problem:** Given a graph $\hat{G}$, produce a partition $\hat{\phi} : \hat{\phi}(u) = \hat{\phi}(v)$ iff $\phi(u) = \phi(v)$.

If we have G, it is easy to reconstruct ψ by clustering columns of G. Also we know several facts:

For any ψ and P, $\mathrm{rank}(G) = k$.
If P_G is the projection on the column space of G:

$$|P_G(G_u) - G_u| = 0$$
$$\left|P_G(G_u) - \hat{G}_u\right| = \varepsilon \text{ is small.}$$

We do not have access to P_G. So, the approach is to approximate P_G by P_X projector so that:

$$|P_X(G_u) - G_u| \text{ is small}$$
$$\left|P_X(G_u - \hat{G}_u)\right| \text{ is small}$$

Main conclusion that $\left|P_X(\hat{G}_u) - G_u\right| = \varepsilon$ is small. If $|G_u - G_v|$ for $\phi(u) \neq \phi(v)$ is much larger then approximation error ε we can use simple clustering method to the $P_X(\hat{G}_u)$.

Let τ be clustering threshold parameter and **CProj** be a function for projection matrix computing. Also we will split $\hat{G}$ matrix into two parts. This is done to avoid the conditioning between the error $|\hat{G} - G|$ and **CProj**$(\hat{G})$.

Algorithm:

```
1. Randomly divide vertex set $\{1,\dots,n\}$ into two parts. Then columns of $\hat{G}$ will split as $[\hat{A}\,|\,\hat{B}]$.
2. Build 2 projectors: $P_{\hat{B}} = \mathbf{CProj}(\hat{B})$, $P_{\hat{A}} = \mathbf{CProj}(\hat{A})$.
3. $\hat{H} = [P_{\hat{B}}(\hat{A})\,|\,P_{\hat{A}}(\hat{B})]$. Here we see splitting effect: no $P_{\hat{G}}(\hat{G})$ estimations.
4. While there are unpartitioned nodes
    (a) Choose an unpartitioned node $u_i$ arbitrary.
    (b) For each node $v$ set $\hat{\psi}(v) = i$ if $|\hat{H}_{u_i} - \hat{H}_v| \leq \tau$.
5. Return partition $\hat{\psi}$.
```

Also we can understand this splitting is technique to avoid overfitting (or avoid build projector and run clusterisation on the same data).

So now we should understand how **CProj** works. Notation $\hat{A}_v^T$ is the vth column of the transpose of $\hat{A}$, which is vector of roughly $n/2$ coordinates.

Algorithm CProj$(\hat{A}, k, s_m, \tau)$ not clear:

```
1. While there are at least $s_m/2$ unclassified nodes
    (a) Choose an unclassified node $v_i$ randomly.
    (b) Let $T_i = \{u : |P_{\hat{A}^T}(\hat{A}_{v_i}^T - \hat{A}_u^T)| \leq \tau\}$.
    (c) Mark each $u \in T_i$ as classified.
2. Assign each remaining node to the $T_i$ with closest projected $v_i$.
3. Let $\hat{c}_i$ be the characteristic vector of $T_i$.
4. Return $P_{\hat{c}}$ — the projection onto the span of the $\hat{c}_i$
```

If the $\hat{c}_i$ were characteristic vectors of ψ, this projection would be exactly P_A. Instead, we will see that the $\hat{c}_i$ are not unlike the characteristic vectors of ψ.

For τ we know theoretical formula.

DCSBM (Degree-corrected stochastic blockmodel) [8], description from [14]. In MMSB (also as in just stochastic blockmodel), a node's community determines its behavior entirely, and thus all nodes in the same community are "stochastically equivalent", and in particular have the same expected degree. This is known to be often violated in practice, due to commonly present "hub" nodes with many more connections than other nodes in their community. Or just free-scale distribution on node's degrees. The degree-corrected stochastic block model (DCSBM) (Karrer and Newman, 2011) was proposed to address this limitation, which multiplies the probability of an edge between nodes i and j by the product of node-specific positive "degree parameters" d_i, d_j.

In DCSBM we represent the probability matrix of the network in following way:

$$P = \rho D\Theta B\Theta^T D, \qquad (2)$$

The $n \times n$ diagonal matrix $D = \mathrm{diag}(d_1, \ldots, d_n)$ contains non-negative degree correction terms that allow for heterogeneity in the node degrees. There is main differences between two models.

OCCAM (The overlapping continuous community assignment model). [14], presentation [1] This model is similar to GeoNMF, but instead of MMSB model it uses OCCAM. Main approach remains the same. Suppose that P factorizes $P = WW^T$. Thus, from (2) we have $W = \sqrt{\rho} D\Theta B^{1/2}$. $B \succ 0$, $B_{kk} = 1$ in this model.

Main intuition is follows. Let's ignore Θ for now. Consider a pure node from community k. Pure means that the node is contained in only one community: $\Theta_i = (0, \ldots, 0, 1, 0, \ldots, 0)$. Then $W_i = \Theta_i B^{1/2} = (B^{1/2})_k$ — k-th column of $B^{1/2}$ matrix. Thus, $B^{1/2}$ is equal to latent positions of pure nodes. We can think about this vertex as community centers. Any row of W is a linear combination of community centers with coefficients Θ_i. So we have vector embedding for each vertex in pure node basis.

Here we have strategy for solving the problem. First, find community centers, i.e. estimate the rows of $B^{1/2}$. Second, project rows of W onto span of $B^{1/2}$ rows to estimate Θ.

Algorithm:

1. Compute the leading K eigenvectors U and eigenvalues Λ of A, set $\hat{W} = U\Lambda^{1/2}$ (so $A \approx \hat{W}\hat{W}^T$)
2. Normalize rows of $\hat{W}$. $\hat{W}_i^* = \dfrac{\hat{W}_i}{\|\hat{W}_i\| + \tau_n}$.
3. Apply K-medians clustering to rows of $\hat{W}^*$ to estimate community centers, i.e. positions of pure nodes: $\hat{S} = \{\hat{s}_1, \ldots, \hat{s}_K\}$.
4. Project the rows of $\hat{W}^*$ onto $\hat{S}$ to obtain the coefficients
$$\tilde{\Theta} = \hat{W}^* \hat{S}^T (SS^T)^{-1}.$$
and normalize rows to obtain the final membership vectors $\hat{\Theta}_i = \dfrac{\tilde{\Theta}_i}{\|\tilde{\Theta}_i\|_2}$.

Here $\tau_n > 0$ is a small regularizer ensuring numerical stability in practice and concentration in theory.

SMBO (Stochastic block model with overlaps) description from [9]. One another approach to generalize Stochastic block model to overlapping community case. It is very close to MMSB model, but has had community affiliation:

$$\theta \in \{0, 1\}^{n \times K}.$$

And

$$p = \rho\Theta B\Theta^T, B \in [0;\, 1]^{K \times K}.$$

It is not difficult to understand that SMBO with K communities is equal to SBM [7] with up to 2^K communities. Next method use this generalization to recover community structure.

SAAC (Spectral algorithm with additive clustering) [9]. SMBO is in background (see previous section).

It has links to AGM, OCCAM and other models. add about it. Uses USVT for eigenvectors selection [5].

Algorithm:

```
1. Selection of the eigenvectors. Form $\hat{U}$ a matrix whose columns are $\hat{K}$ eigenvectors of $A$ associated to eigenvalues $\lambda$ satisfying condition from USVT:
       $|\lambda| > \sqrt{2(1+\nu)\hat{d}_{\max}\log(4n^{1+r})}.$
2. Initialization. $\hat{\Theta} = 0 \in \mathbf{R}^{n \times \hat{K}}$. $\hat{B} \in \mathbb{R}^{\hat{K} \times \hat{K}}$ initialized with k-means++ applied to $\hat{U}$, the first centroids being chosen at random among nodes with degree smaller than the median degree.
3. while ( $Loss - \|\hat{U} - \hat{\Theta}\hat{B}\|_F^2 > \varepsilon$) do
4.     $Loss = \|\hat{U} - \hat{\Theta}\hat{B}\|_F^2$
5.     Update membership vectors (like k-means):
           $\forall i: \quad \hat{\Theta}_i = \underset{\theta \in [0,1]^{\hat{K}}: 1 \le \|z\|_1 \le m}{\arg\min} \|\hat{U}_i - \theta\hat{X}\|.$
6.     Update centroids: $\hat{B} = (\hat{\Theta}^T\hat{\Theta})^{-1}\hat{\Theta}^T\hat{U}$.
```

Here k-means++ is initialization. It is a randomized procedure that picks as initial centroids rows from $\hat{U}$ that should be far from each other [3].

References

[1] *Overlapping community detection by spectral methods. presentation.*, 2014, http://stat.cornell.edu/sites/cstats/files/Liza%20Levina%20cornell.pdf 2016-12-15.

[2] Edoardo M Airoldi, David M Blei, Stephen E Fienberg, and Eric P Xing, *Mixed membership stochastic blockmodels*, Journal of Machine Learning Research **9** (2008), no. Sep, 1981–2014, http://www.jmlr.org/papers/volume9/airoldi08a/airoldi08a.pdf.

[3] David Arthur and Sergei Vassilvitskii, *k-means++: The advantages of careful seeding*, Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, Society for Industrial and Applied Mathematics, 2007, `http://ilpubs.stanford.edu:8090/778/1/2006-13.pdf`, pp. 1027–1035.

[4] Brian Ball, Brian Karrer, and Mark EJ Newman, *Efficient and principled method for detecting communities in networks*, Physical Review E **84** (2011), no. 3, 036103, `https://arxiv.org/pdf/1104.3590.pdf`.

[5] Sourav Chatterjee et al., *Matrix estimation by universal singular value thresholding*, The Annals of Statistics **43** (2015), no. 1, 177–214, `https://arxiv.org/pdf/1212.1247.pdf`.

[6] Santo Fortunato, *Community detection in graphs*, Physics Reports **486** (2010), no. 3, 75–174, `http://www.arxiv.org/abs/0906.0612`.

[7] Paul W Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt, *Stochastic blockmodels: First steps*, Social networks **5** (1983), no. 2, 109–137, `https://www.researchgate.net/profile/Kathryn_Laskey/publication/247650049_Stochastic_Blockmodels_First_Steps/links/562e3a5108ae04c2aeb5bb56.pdf`.

[8] Brian Karrer and Mark EJ Newman, *Stochastic blockmodels and community structure in networks*, Physical Review E **83** (2011), no. 1, 016107, `https://arxiv.org/pdf/1008.3926.pdf`.

[9] Emilie Kaufmann, Thomas Bonald, and Marc Lelarge, *A spectral algorithm with additive clustering for the recovery of overlapping communities in networks*, arXiv preprint arXiv:1506.04158 (2015), `https://hal.archives-ouvertes.fr/hal-01163147v2/document`.

[10] Xueyu Mao, Purnamrita Sarkar, and Deepayan Chakrabarti, *Provable symmetric nonnegative matrix factorization for overlapping clustering*, arXiv preprint arXiv:1607.00084 (2016), `https://arxiv.org/pdf/1607.00084v1.pdf`.

[11] Frank McSherry, *Spectral partitioning of random graphs*, Foundations of Computer Science, 2001. Proceedings. 42nd IEEE Symposium on, IEEE, 2001, `http://www.cc.gatech.edu/~mihail/D.8802readings/mcsherrystoc01.pdf`, pp. 529–537.

[12] Jaewon Yang and Jure Leskovec, *Community-affiliation graph model for overlapping network community detection*, 2012 IEEE 12th International Conference on Data Mining, IEEE, 2012, `http://www-cs-faculty.stanford.edu/people/jure/pubs/agmfit-icdm12.pdf`, pp. 1170–1175.

[13] ———, *Overlapping community detection at scale: a nonnegative matrix factorization approach*, Proceedings of the sixth ACM international conference on Web search and data mining, ACM, 2013, `http://i.stanford.edu/~crucis/pubs/paper-nmfagm.pdf`, pp. 587–596.

[14] Yuan Zhang, Elizaveta Levina, and Ji Zhu, *Detecting overlapping communities in networks using spectral methods*, arXiv preprint arXiv:1412.3432 (2014), `https://arxiv.org/pdf/1412.3432.pdf`.