情報科学科総合演習課題レポート

情報科学科 3 年　中嶋一貴

平成 28 年 11 月 29 日

1　はじめに

今回は情報科学科総合演習においてグラフサンプリングについて学んだ。グラフとは頂点と枝からなるもので、ネットワークの様子を表すことができる。通常、このグラフのすべての頂点や枝、隣接関係といった情報を知ることはできない。しかし、どうにかグラフの特徴を知りたいとしたときに、グラフの一部分の情報 (全体ノードの 1%など) を取り出し、全体像を推定するというのがグラフサンプリングの基本的な考え方である。今回は全体ノードから一部分を取り出し、クラスタ係数やネットワークサイズを推定することを主なテーマとした。

2　基礎的な定義

2.1　グラフとは

ノード (頂点) の集合とエッジ (枝) の集合で構成されるものである。グラフによって様々なものを表すことができ、例えばノードをユーザー、エッジを交友関係とすればフェイスブックといったソーシャルネットワークについて表すことができる。またグラフには有向グラフと無向グラフがあり、前者はエッジに矢印があるグラフで、後者はエッジに矢印がないグラフである。今回はネットワークについて考えるので無向グラフである。V をノードの集合,E はエッジの集合として

$$G; = (V, E) \tag{1}$$

V の要素はノードで v_iのように表記し、E の要素は例えば v_iと v_jがつながっている場合は (v_i, v_j) のように表記する。また任意の 2 頂点間に枝があるグラフを完全グラフといい、n 頂点の完全グラフを K^n と表記する。

$$K^n = (V, E) : \forall v_i, v_j \in V \ s.t. \ (v_i, v_j) \in E \tag{2}$$

2.2　隣接ノード

ネットワークを考える上で隣接ノードは非常に重要である。v_iの隣接ノードの集合を N_i とすると

$$N_{v_i} = \{v_j | (v_i, v_j) \in E\} \tag{3}$$

2.3 次数

頂点 v_iの次数 d_{v_i}は v_iの隣接ノード集合 N_{v_i}の要素数で次のように定義される。

$$d_{v_i} = |N_{v_i}| \tag{4}$$

またグラフ G の平均次数 d は次のように定義される。

$$d = \frac{\sum_{i=1}^{|V|} |N_{v_i}|}{|V|} \tag{5}$$

3 NMSE

$NMSE$ とはサンプリング手法の精度を測るもので、推定値の平均が真値とどれだけはなれているか、そして推定値がどれだけばらついているかをしめす。$NMSE$ が小さいほど、いい精度と言える。N を測定回数, y を真値, y_i を i 回目の推定値とすると

$$NMSE = \frac{\sum_{i=1}^{N} (y - y_i)^2}{Ny} \tag{6}$$

4 複雑ネットワーク

ソーシャルネットワークといった巨大で複雑なネットワークには共通する性質がある。

4.1 スケールフリー性

一部の頂点が他のたくさんの頂点と繋がっている一方で、ほとんどの頂点はごくわずかの頂点しか繋がっていないという性質である。たとえばツイッターではフォロー数が数万を超えるユーザーはごく一部で、ほとんどのユーザーは数百といった程度である。数学的には頂点が次数 k を持つ確率 $p(k)$ の確率分布が $p(k) \propto k^{-\gamma}$のべき乗則になると表現される。グラフがこのような性質をもつことをスケールフリーと呼ぶ。

4.2 スモールワールド性

任意の 2 つの頂点が中間にわずかな数の頂点を介するだけで接続されるという性質である。たとえば一見、赤の他人に見えても中間に少数の知人を介することでつながっているといったことである。これを数学的に定義するときに平均最短距離という概念が必要である。平均最短距離 L は、無向グラフにおいて任意の頂点 v_iから v_jへ行くまでに通過しなければならない辺の本数

をパス長といい、その中で最短のものを最短距離 D_{ij} とすると D_{ij} の平均値が L である。このとき、n が増大したときに L が高々 $log(n)$ に比例する程度でゆるやかに増加するときスモールワールド性を満たすと定義される。また L は頂点数を n とすると次のように求められる。

$$L = \frac{\sum_{i>j} D_{ij}}{\frac{n(n-1)}{2}} \tag{7}$$

4.3 クラスター性

密度の濃い繋がりが多数存在していることを示すものである。知人関係において、自分と知人 A さんがいるときに自分も A さんもどちらも知っている共通の知人 B さんのような人が一人もいないという状況はまずありえない、ということである。数学的にはクラスタ係数 C によって表現される。頂点 v_i のクラスタ係数 C_i とは頂点 v_i の隣接ノードの中から任意に 2 個選んだときにその 2 個が隣接しているかの割合を表す量で、, e_i = (v_i と隣接しているノードの中に存在するエッジの数) とすると次のように定義できる。

$$C_{v_i} = \begin{cases} 0 & (d_{v_i} = 0, 1) \\ \frac{2e_i}{d_{v_i}(d_{v_i}-1)} & (otherwise) \end{cases}$$

平均クラスター係数 C は次のように定義される。

$$C = \sum_{i=1}^{i=|V|} \frac{C_{v_i}}{|V|} \tag{8}$$

この平均クラスター係数 C は様々なネットワークで計測されており、0.1 から 0,7 程度である。

5 プログラムの基本的な関数

プログラムでは $numpy, networkx, matplotlib$ を用いて可視化を図った。

5.1 グラフ作成

グラフ作成は関数 $makeG()$ によって行われる。各行にエッジを表す 2 つの整数が書かれているテキストファイルを読み込み、そこからノードの集合 V とエッジの集合 E を生成し、グラフ G を定義する。はじめは $makeG()$ でグラフ G を作成し、ノードやエッジの集合にアクセスして隣接リストなどを生成するという方法だったが、膨大な時間がかかったためテキストファイル読み込みの際にグラフ G 作成と並行してできる作業をすべて行うようにし

た。グローバル変数にノードのリスト $vlist$, エッジのリスト $elist, vlist$ の長さ $lenofvlist, elist$ の長さ $lenofelist$, 隣接リストの辞書 $nlist$, クラスター係数の辞書 $clist$, 平均クラスタ係数 $avec$ を定義する。$vlist$ の生成には、テキストファイルの各行の数字をグラフ G のノード集合に add していき、すべておわったら $vlist = G.nodes()$ とした。$G.nodes()$ にすでにある頂点 v について $G.add\_node(v)$ としても重複生成されないという便利な性質があり、if 文を省略することができた。$elist$ にはテキストファイルの各行を add していき、最後に $elist = G.edges()$ とした。$lenofvlist, lenofelist$ は $vlist, elist$ の長さを $len()$ を用いた。$nlist$ の生成には $defaultdict$ という辞書機能を利用した。$nlist$ という辞書はキー i に対してノード i の隣接ノードのリストを持っている。たとえばテキストファイルのある行で「a b」とあった場合は (a, b) というエッジが存在するということなので、キー a の隣接リストに b を追加し、キー b の隣接リストに a を追加する。またテキストファイルは、エッジは重複したものは存在せず各列のノードは昇順であるので隣接リストは重複は存在せず、各隣接リストは昇順となる。このようにはじめに値を記憶させておくことで実行時間を大幅にへらすことができた。平均クラスタ係数 $avec$ もここで計算しておくことで $makeG()$ が終わった時点で必要な情報はすべて記憶してある状態になっている。

5.2 グラフの描画

グラフの描画は関数 $drawG()$ によってできる。これは $networkx$ と $matoplotlib$ を用いた。

5.3 完全グラフの作成

完全グラフの作成には $compG()$ を用いる。頂点数 n さえわかればよい。頂点 0 から $n-1$ をグラフ G のノード集合に add していき、エッジについては $0 \leq j \leq n-1,\ 0 \leq k \leq n-1$ を満たす (j, k) をすべて追加していく。作成が終わったらグローバル変数にそれぞれ代入する。

5.4 隣接リスト

ノード v の隣接リストを得るには $neighbors()$ を用いる。ノード v の隣接リストは $nlist[v]$ を得ればよい。

5.5 クラスタ係数

$clustering()$ を用いる。クラスタ係数 C_{v_i} はノード v_iの隣接ノードの 2 つがつながっているかを数える。このためには v_iの隣接ノード v_jを一つ決めて

、$v_k \in N_{v_i}$ が N_{v_j} に含まれるかを調べていけばよい。クラスタ係数 C_{v_i} を求めたら辞書 $clist$ のキー i に値を登録する。またグラフの平均クラスタ係数は $aveclustering()$ を用いる。定義通り、順々に各ノードのクラスタ係数を求めて平均を取ればよい。

5.6 次数分布のグラフの表示

これには plot() を用いる。http://yamaguchiyuto.hatenablog.com/entry/2014/12/14/121755 を参照した。

5.7 次数

次数は定義より隣接リストの長さであるのでノード v の次数 d を得るときは $len(nlist[v])$ を得ればよい。またグラフの平均次数を得る関数は $aved()$ であり、これは定義どおり、順々に各ノードの次数を求め平均値をとればよい。

6 主なサンプリング手法

主なサンプリング手法として、BFS,RW,MHRW について述べる。

6.1 初期ノードの取り方

初期ノードはグラフ G の頂点集合 V から一様に 1 つ選ぶ。BFS では途中のステップで隣接リストが空の場合を考えている。RW,MHRW では初期ノードは次数 0 でないノードになるようにする。次数が 0 のノードを選んだ場合はそのノードをサンプリングノード列に追加し、また一様に集合 V から選ぶ。

6.2 BFS

BFS のアルゴリズムを以下に示す。

1. 初期ノードを 1 つ選ぶ。
2. vlist をグラフ G の頂点リストとし、queue と sampling_v を用意する
3. 初期ノードを queue に入れる
4. queue の先頭ノードを v とし、queue と vlist から v を消去する

5. v が sampling_v になければ追加し、n を 1 減らす。v の隣接リストが空の場合は queue に vlist の先頭ノードを追加する。空でないときは隣接リストのそれぞれのノードについて sampling_v に入ってないものを queue,sampling_v に追加し n を 1 減らしていく。

6. n=0 または queue が空になるまで 3,4 を繰り返す。

n は残りサンプル数を示しており、0 のとき、サンプリングノード列を返し、0 でないとき、サンプル数を抽出できなかったことを伝える。BFS はグラフによらず完全な解をみつけることができ、開始ノードと終了ノードの長さが最も少ない辺となる。一方で巨大なグラフに対しては非現実的な実行時間がかかり、またグラフが無限で検索対象が存在しない時には関数が終了しない。また次に触れる RW や $MHRW$ とくらべてマルコフ性がないためサンプリングにおいては好まれない。

6.3 RW

RW のアルゴリズムは以下のとおりである。

1. sampling_v を用意する。

2. 次数が 0 でないような初期ノードを 1 つ選ぶ。

3. v を sampling_v に追加し、v の隣接リストの中から一様にノードをひとつえらびそれを v とする。

4. 3 をサンプル数に達するまで行う。

$R = (x_1, ..., x_r)$ を RW によって得られたサンプリングのインデックスの列とする。r は合計ステップ回数である。d_vを v の次数として、遷移確率行列を $\mathbf{P} = \{P(v,w)_{v,w \in V}\}$ とすると

$$C_i = \begin{cases} \frac{1}{d_v} & (w \in N_v) \\ 0 & (otherwise) \end{cases}$$

事象 A の起こる確率を P(A) と表すことにすると RW による分布は

$$\pi = (P(x_r = 1), P(x_r = 2), ..., P(x_r = n)) \tag{9}$$

r が十分大きいときの確率 $P(x_r = i)$ を$\pi(i)$ と定義すると$\pi(i) = \frac{d_{v_i}}{|V|d}$に収束する。よって RW では高い次数のノードに訪れやすく誤差が生じやすい。

6.4 MHRW

$MHRW$ は RW の高い次数のノードに訪れやすいという欠点を補った手法であり、アルゴリズムは以下のようになる。

1. sampling_v を用意する。
2. 次数が 0 でないような初期ノードを 1 つ選ぶ。
3. v の隣接リストから一様にノード w を選ぶ。v を $sampling\_v$ に追加し、0〜1 の一様乱数 p と $k = \frac{d_v}{d_w}$ に対して $p \leq k$ ならば次のノードを w とする。$p > k$ ならば次のノードを v とする。
4. 3 をサンプル数に達するまで行う。

3 番目で p¿k のときにノード v をもう一度サンプリングすることにより、次数によるサンプリングノードの偏りを小さくすることができる。初期ノードで次数 0 のノードをとっていないため、次数 0 のノードに訪れることはありえない。

7 RW,MHRW によるクラスタ係数の推定

$RW, MHRW$ による 1 つのサンプリングノード列にたいして推定値を返す関数をそれぞれ $RW\_ev, MHRW\_ev$ とした。ともに、サンプリングノード列のそれぞれのノード v に対して C_v を計算してそれらの平均値を取り、それを推定値とする。また $RW\_est, MHRW\_est$ はそれぞれ $RW\_ev, MHRW\_ev$ をある回数求めて、それらの平均、分散、$NMSE$ を求める。

7.1 実験

BA グラフと amazon グラフにたいして実験を行った。BA グラフはノード数 10000, エッジ数 15875,amazon グラフはノード数 334863, エッジ数 925872 である。サンプル数は全ノードの 1%、試行回数は、BA グラフは真値がとても小さいため 10000 回、amazon グラフは 1000 回とした。

グラフ	手法	平均	分散	NMSE	真値
BA	RW	0.00202936	1.39676e-05	1.978953	0.00188985
BA	MHRW	0.001779415	0.000275409	8.78153	0.00188985
amazon	RW	0.358300	0.00106840	0.127192	0.396746
amazon	MHRW	0.395563	0.00231301	0.121257	0.396746

7.2 考察

BA モデルに対しては RW のほうが精度が高く、$amazon$ グラフに対しては $MHRW$ はとても高い精度を示した。BA モデル中の次数 1 のノードは 21 個で、$amazon$ グラフの次数が 1 のノードは 252 個である。$MHRW$ では次数が低いノードほどサンプリング数が多くなり、さらに次数が 1 のときはクラスタ係数が 0 であるため、BA グラフは頂点数に対して次数 1 のノードの割合が高いため、$MHRW$ では誤差が大き成るのではないかと推測する。よってグラフによって手法を適切に選ばないといけないことがわかる。

8 ネットワークサイズ推定

8.1 推定値の求め方

RW によるサンプリングでネットワークサイズ n を推定する。RW において互いにインデックスがある値以上の差があるようなノードのペアを観察する。RW によるサンプリングノード列を $\{x_0, ..., x_{r-1}\}$ サンプリング数を r としてそれらの集合 I は

$$I = (k, l) : m \leq |k - l| \text{ かつ } 0 \leq k, l < r \tag{10}$$

またここで新しい変数$\phi_{k,l}$を導入する。

$$\phi_{k,l} = \begin{cases} 1 & (x_k = x_l) \\ 0 & (otherwise) \end{cases}$$

さらに

$$\Phi_n = \frac{1}{|I|} \sum_{(k,l)\in I} \phi_{k,l} \tag{11}$$

$$\Psi_n = \frac{1}{|I|} \sum_{(k,l)\in I} \frac{d_{x_k}}{d_{x_l}} \tag{12}$$

と定義すると、ネットワークサイズの推定値 $\bar{n}$ は

$$\bar{n} = \frac{\Psi_n}{\Phi_n} \tag{13}$$

と求まる。

8.2 実験

関数 $estimation()$ があるグラフに対してネットワークサイズ推定を返す関数である。また c_iは信頼区間を返す関数である。ここではサンプル数 r は全体のノード数 N の $1\%, m$ は n の 2.5%と設定した。各グラフについて 10 回推定値をだし、そこから信頼区間を求めた。

8.2.1 BA グラフ

「BA グラフ」について 10 回実行したときの各値について表で示す。BA グラフはノード数 10000, エッジ数 15875 である。信頼区間は $[357.6, 741.4]$ ともとまった。

$\bar{n}$	Ψ	Φ
1056.35830722	34719.9878641	32.8676242019
420.149746951	23364.0788417	55.6089323181
282.508000566	37614.879574	133.146245411
775.297304103	28261.1376334	36.4520003924
493.793484495	19023.6346669	38.5254874037
846.076614776	23423.682911	27.6850612604
440.46197549	29675.2038634	67.3729073443
342.846053569	22012.0904354	64.2040070353
228.928767577	22126.6088312	96.6528106773
608.322886098	42946.8218697	70.5987278321

8.2.2 amazon グラフ

「$amazon$ グラフ」について 10 回実行したときの各値について表で示す。$amazon$ グラフはノード数 334863, エッジ数 1049866 である。信頼区間は $[11687.1, 43676.7]$ ともとまった。

$\bar{n}$	Ψ	Φ
1435.74980514	19176574.6492	13356.4877255
4948.0639549	20237928.3312	4090.07007906
144179.275296	20392920.8252	141.441415788
98979.47409	23652390.4447	238.962579486
78068.2193486	27148550.525	347.754191802
11292.3716687	21636965.9187	1916.06923272
5458.91720275	30082940.8553	5510.78899679
28660.5092959	19082216.523	665.801724804
106718.104004	26841173.3744	251.514713693
22402.7593904	20654428.4121	921.959123525

8.2.3 DBLP グラフ

「$com-DBLP$」について 10 回実行したときの各値について表で示す。$com-DBLP$ グラフはノード数 317080, エッジ数 925872 である。信頼区間は $[112089.6, 357840.4]$ ともとまった。

$\bar{n}$	Ψ	Φ
425348.132367	32507173.8761	76.4248656627
5365.40628651	28749035.9407	5358.22161556
343191.817066	32338509.5006	94.2286729825
97736.2557599	32876306.1724	336.377794676
94261.9261444	31202806.1181	331.022369204
51186.6731675	30332194.0742	592.579907958
393663.453426	30059441.2702	76.3582217465
349940.940429	35027919.8474	100.096661467
230752.590796	34224044.0919	148.31488554
276776.657713	32463272.9378	117.290501323

8.3 考察

BA グラフ,amazon グラフともにとても精度の低い結果となっている。これに対して DBLP ではある程度の精度の結果となった。これにより、この推定方法はグラフに依存することがわかる。また DBLP についても、Φ のばらつきがおおきく、値の誤差が生じている。これは Phi の定義にもどればサンプリングノード列に等しいものが多数あるということでこれは RW の高い次数に訪れやすいという性質によるものだと考える。I とはインデックスが m 以上離れたノードのペアの集合であった。$\bar{n}$ を求めるのに I から要素を取り出しているため、インデックスが離れた同士のノードのつながりが疎であるほど精度が低くなると予想する。また $O(n^2)$ の実装しかできなかったため、$O(n)$ の実装ができれば DBLP に関してはさらに良い結果がでるはずである。

9 おわりに

グラフサンプリングは初学で、最初推定値を求めるのに数時間かかるといった具合でした。今回の演習を通じて知識量がかなり増えました。しかしまだ深く考察するまでのグラフサンプリングの知識量がなく、「ネットワークサイズ推定手法がグラフに依存する要因はなんなのか」という疑問が生じましたが、解決できませんでした。今回は初歩的な研究ということでしたが、個人的には面白いテーマが見つかり、とても充実しました。知識不足で何度もご質問して丁寧に教えてくださりありがとうございました。首藤先生の研究室に配属したらぜひこの「ネットワークサイズ推定の手法がグラフに依存する要因はなにか」について研究したと思います。