

クレジット:

UTokyo Online Education データマイニング入門 2018 森 純一郎

ライセンス:

利用者は、本講義資料を、教育的な目的に限ってページ単位で利用することができます。特に記載のない限り、本講義資料はページ単位でクリエイティブ・コモンズ 表示-非営利-改変禁止 ライセンスの下に提供されています。

<http://creativecommons.org/licenses/by-nc-nd/4.0/>

本講義資料内には、東京大学が第三者より許諾を得て利用している画像等や、各種ライセンスによって提供されている画像等が含まれています。個々の画像等を本講義資料から切り離して利用することはできません。個々の画像等の利用については、それぞれの権利者の定めるところに従ってください。



データマイニング入門 第6回

2018年度

学習目標

ネットワークデータの行列表現（隣接行列）を理解する

重み付き、有向ネットワークについて理解する

ネットワークの最短経路長について理解する

ネットワークの中心性について理解する

- ・ 次数、近接、媒介

平均パス長、クラスタリング係数について理解する

ネットワークのコミュニティ抽出について理解する

Pythonでネットワークデータの基本的な処理を理解する

ネットワーク分析

グラフマイニング、リンク解析とも呼ばれる

応用

- ソーシャルネットワーク分析
- 引用分析
- 物質・化合物、タンパク質相互作用
- サプライチェーンネットワーク
- 交通・輸送ネットワーク
- インターネット
- パワーグリッド
- 神経ネットワーク
- 生体ネットワーク
- など

データとは

一般的な多次元データ

- レコードの集合
 - レコード：データポイント、インスタンス、example、エンティティ、オブジェクト、特徴量ベクトル
- 各レコードはフィールドの集合からなる
 - フィールド：属性、次元、特徴量、変数

通常はレコード間には依存関係がなく独立であることを仮定

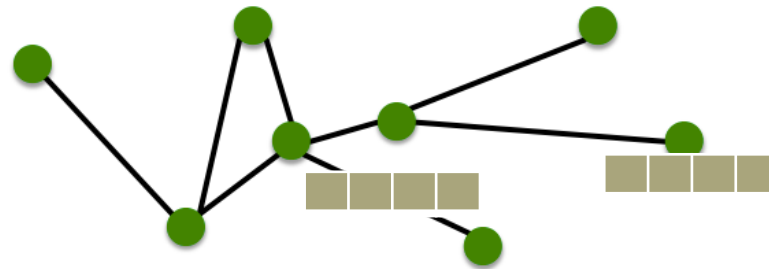
レコード間に依存関係があるデータもある

- レコード間の意味的、時間的、空間的な関係
 - 時系列、ネットワーク、文字列、空間データなど
- レコード間の関係性を考慮したデータ分析が必要

ネットワークデータ

ネットワーク（グラフ）はリンクで連結されたノードの集合

- ノードの集合 N
 - ノードの数 $|N|=n$
- リンクの集合 E
 - リンクの数 $|E|=m$
- グラフ $G(N,E)$



Components of a Network
Stanford CS224w: "Social and Information Network Analysis", p39
<https://slideplayer.com/slide/14061351/>
(ref. 20 Dec 2018)

各ノードをデータの各レコードと考えるとネットワークはレコード間の関係を表している

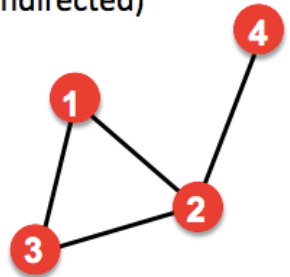
ノード、リンクはそれぞれフィールド（属性）を持つこともある

隣接行列

ネットワークのノード間の関係を行列として表したものの

- ノードを行（列）とする
- ノード間にリンクがあれば隣接行列の対応する要素は1となる
- 各行（列）はあるノードの他のノードとのつながりを表すベクトルとなっている
- ネットワークが無向であれば隣接行列は対称
- リンクに重み（数値）があるネットワークでは隣接行列の要素は重みを表す実数値
 - *重みは負でもよい

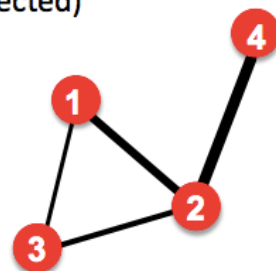
(undirected)



無向重みなしネットワーク

$$A_{ij} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

(undirected)



無向重みありネットワーク

$$A_{ij} = \begin{pmatrix} 0 & 2 & 0.5 & 0 \\ 2 & 0 & 1 & 4 \\ 0.5 & 1 & 0 & 0 \\ 0 & 4 & 0 & 0 \end{pmatrix}$$

隣接行列

リンクに方向性があるネットワーク

- 有向ネットワークの隣接行列の要素は方向が反映される
 - ノード*j*からノード*i*に方向性があるリンクがあれば隣接行列の $A_{ij}=1$, $A_{ji}=0$
- 隣接行列は一般に非対称

ノード間に複数のリンクがあるネットワーク（やや例外的）

- 隣接行列の対応する要素にリンクの数を表す値
- 重みありネットワークともみなせる

ノード自身にリンクが自己ループしているネットワーク（やや例外的）

- 隣接行列の対応する対角要素に自己ループの数を表す値
 - 有方向なら1つの自己ループを1と数える
 - 無方向なら1つの自己ループを2（リンクの両端分）と数える

隣接行列

完全グラフ

- すべてのノード間にリンクがあるネットワーク
 - $m = n(n-1)/2$ (無方向ネットワーク)
 - ノードの次数は $n-1$

疎グラフ

- 現実の多くのネットワークではリンクが疎 (隣接行列は0要素が多い)

強連結グラフ

- 任意のノード間にパス (有方向の時はリンクの方向も考慮したパス) が存在するネットワーク

弱連結グラフ

- リンクの方角を考慮しなければ任意のノード間にパスが存在するネットワーク

パス（経路）

ネットワーク上であるノードからノードへの経路

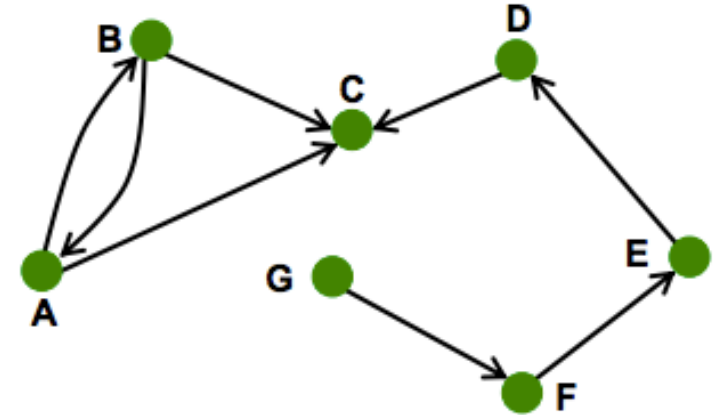
- 経路はリンクで連結される
- 経路はノードの連なりからなる
- 有向ネットワークの場合は、経路はリンクの方向に従う必要がある
- 無向ネットワークの場合は、経路はリンクのどちらの方向でもよい

パス（経路）長

- 経路に含まれるリンクの数

重みなしネットワークの隣接行列を n 乗した A^n について

- A^n_{ij} はノード j から i へのパス長 n の経路の数を表す



Directed links

Stanford CS224w: "Social and Information Network Analysis", p44

<https://slideplayer.com/slide/14061351/>
(ref. 20 Dec 2018)

最短経路長

あるノードからノードの間の最短の経路

- 重みなしネットワークの場合

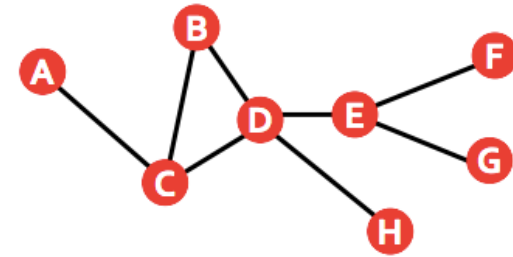
- 幅優先探索

- ノードの隣接ノードを逐次的に調べる
- 始点以外のノードの距離を空に初期化
- 始点の距離を0とする
- 始点の隣接点の距離を1とする
- 距離1のノードの隣接点で距離が空のノードの距離を2とする
- ...繰り返し、終端のノードが見つければ距離を返して終了

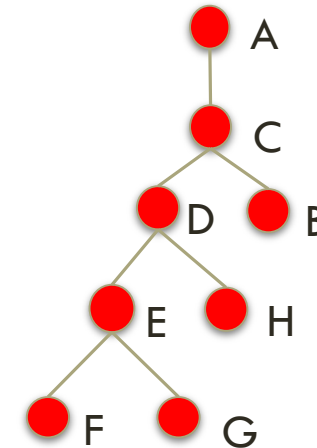
- 重みありネットワークの場合

- ダイクストラ法

- 隣接行列を用いたアルゴリズム



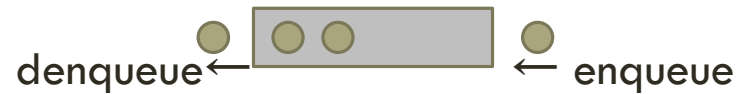
始点を頂点とした木の探索とみなせる



幅優先探索

- `visited[i]`: ノード*i*の探索状態 (True, False)
- `d[i]`: 始点からノード*i*への距離
- `Q`: 次に探索するノードを格納するキュー
 - *課題のコードでは両端キューを使用している
- `s`: 始点

```
visited[s]=True
d[s]=0
Q.enqueue(s)
while Qが空でない
    i=Q.dequeue()
    for iの各隣接ノードjについて
        if visited[j] == False
            visited[j] == True
            d[j]=d[i]+1
            Q.enqueue(j)
```



幅優先探索の時間計算量

キューの操作に $O(n)$ 、各ノードの隣接リストの走査に $O(m)$
全体として $O(n+m)$

中心性

著作権の都合により
ここに挿入されていた画像を削除しました

Ortiz-Arroyo, "Discovering Sets of Key Players in Social Networks. In: Abraham", A., Hassanien, A.-E., Sn'asel, V. (Eds.): Computational Social Network Analysis, Computer Communications and Networks. Springer London, 2010, pp. 27–47.

Fig.2.1 Diverse centrality measures applied on an example network

中心性

ネットワークの各ノードがどれくらい「中心的」であるかを示す指標

ノードの重要度

- ノードの存在がネットワーク全体の構造に影響する度合い
 - 情報の伝搬
 - 感染症の広がり
 - ネットワークの対故障性
 - 人の影響度 など

代表的な中心性

- 次数
- 近接
- 媒介
- 固有値
- PageRank など

中心性

次数中心性

著作権の都合により
ここに挿入されていた画像を削除しました

Ortiz-Arroyo, "Discovering Sets of Key Players in Social Networks. In: Abraham", A., Hassanien, A.-E., Sn´ačsel, V. (Eds.): Computational Social Network Analysis, Computer Communications and Networks. Springer London, 2010, pp. 27–47, Fig.2.1 Diverse centrality measures applied on an example network

- ノードにつながっているエッジの数
- 隣接行列から数えられる
 - $k_i = \sum_j A_{ij}$
- 無方向ネットワークで各ノードの次数の和はリンク数(m)の2倍に等しくなる
 - $\sum_i k_i = 2m$
 - ネットワーク全体の平均次数： $(\sum_i k_i)/n = 2m/n$
- 有向ネットワーク場合
 - 入次数
 - ノードに入ってくるリンクの数
 - 出次数
 - ノードから出ていくリンクの数
- ネットワークの最大次数 (n-1) で正規化することもある

中心性

近接中心性

- あるノードから他のノードへの最短経路長の逆数の和
 - 任意のノードに平均的に短い経路で到達できるノード
 - $C_i = \sum_{j \neq i} 1/d_{ij}$
 - d_{ij} : ノード*i*から*j*への最短経路長
- 最短経路長の和の逆数のパターンもある
 - $1/\sum_{j \neq i} d_{ij}$
- 有向ネットワークは最短経路で方向を考慮

著作権の都合により
ここに挿入されていた画像を削除しました

Ortiz-Arroyo, "Discovering Sets of Key Players in Social Networks. In: Abraham", A., Hassanien, A.-E., Sn'á'asel, V. (Eds.): Computational Social Network Analysis, Computer Communications and Networks. Springer London, 2010, pp. 27–47, Fig.2.1 Diverse centrality measures applied on an example network

中心性

媒介中心性

- あるノードが他の任意のノード間の最短経路に位置する度合い
 - そのノードを通らないと他のノードに行けない
 - 「橋」の役割
- $B_i = \sum_{st} n_{st}^i / g_{st}$
 - n_{st}^i
 - 1 if ノード*i*がノード*s*から*t*への最短経路に含まれる
 - 0 else
 - g_{st} : ノード*s*から*t*へのすべての最短経路の数
- 有向ネットワークは最短経路で方向を考慮

著作権の都合により
ここに挿入されていた画像を削除しました

Ortiz-Arroyo, "Discovering Sets of Key Players in Social Networks. In: Abraham", A., Hassanién, A.-E., Sn'ásel, V. (Eds.): Computational Social Network Analysis, Computer Communications and Networks. Springer London, 2010, pp. 27–47, Fig.2.1 Diverse centrality measures applied on an example network

中心性

固有ベクトル中心性

行列とベクトルの積

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad \vec{x} = \begin{pmatrix} x \\ y \end{pmatrix} \quad A\vec{x} = \begin{pmatrix} ax + by \\ cx + dy \end{pmatrix}$$

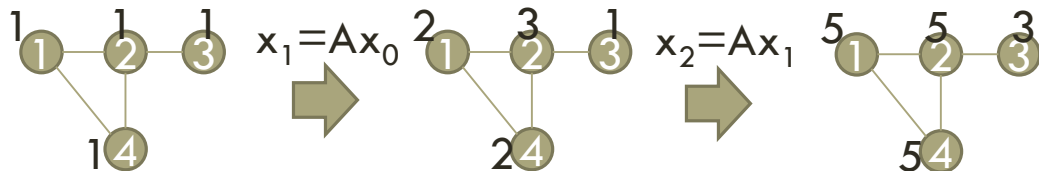
行列Aを隣接行列、ベクトルxを中心性ベクトルとするとAxは各ノードの隣接ノードの中心性値の和をそのノードの新たな中心性値として更新することになる

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \quad Ax_0 = \begin{pmatrix} 0*1+1*1+0*1+1*1 \\ 1*1+0*1+1*1+1*1 \\ 0*1+1*1+0*1+0*1 \\ 1*1+1*1+0*1+0*1 \end{pmatrix}$$

$$x_0 = (1, 1, 1, 1)$$

$$x_1 = (2, 3, 1, 2)$$

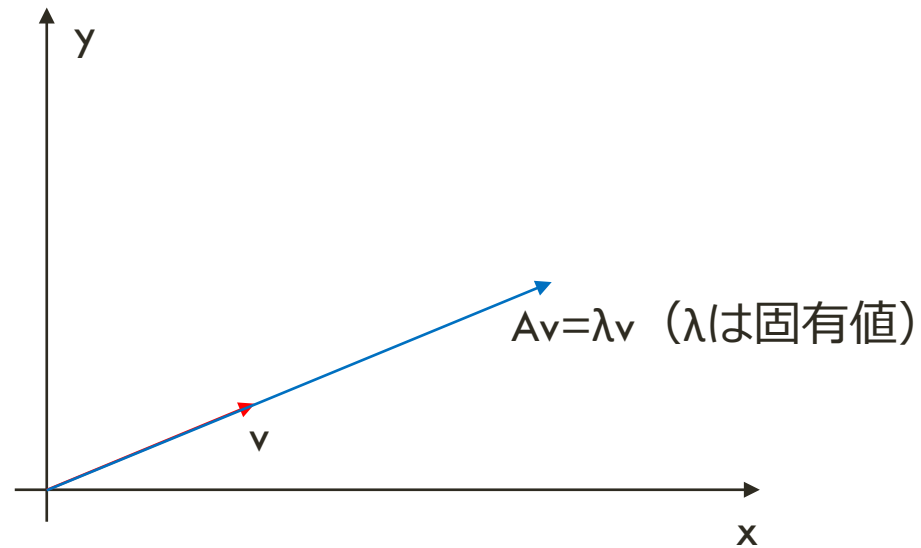
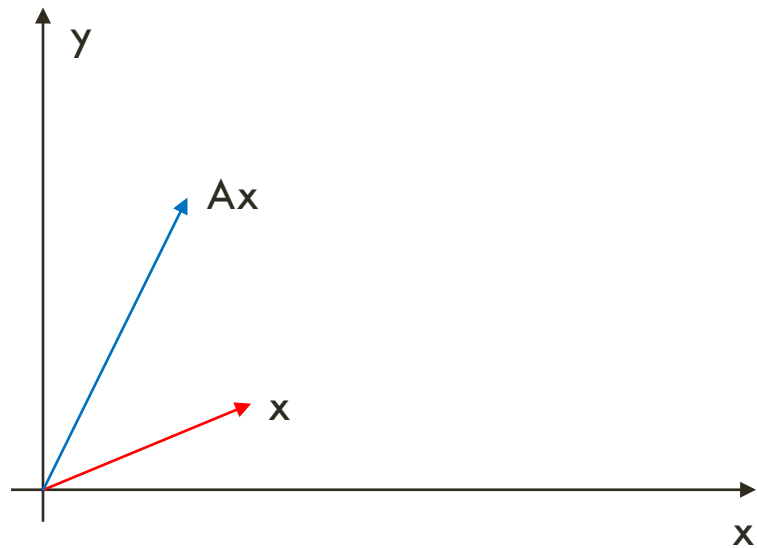
$$x_2 = (5, 5, 3, 5)$$



この中心性の更新を繰り返していくと
中心性ベクトルは定常的なベクトルに
収束する

$$Ax = cx \quad (c \text{ は定数})$$

固有値と固有ベクトル

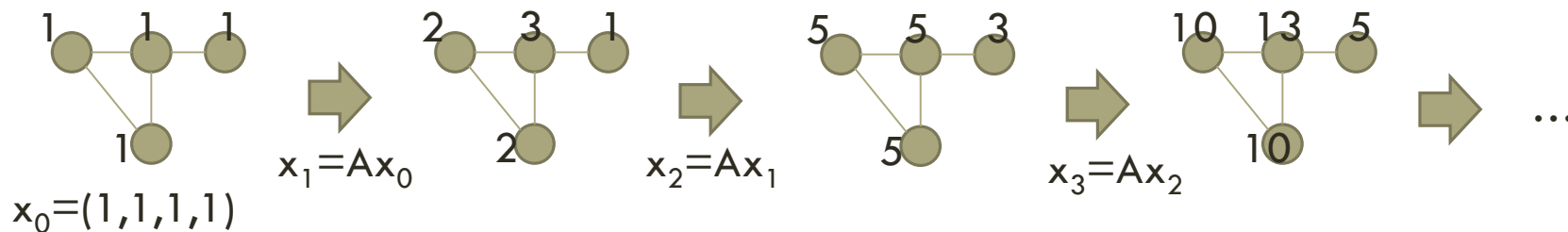


ベクトル x に行列 A をかけることは x を線形変換（回転や折り返しなど）していることになる（標準基底による座標 (x,y) であらわされる点を別の基底による座標点 (x,y) に変換している）

固有ベクトル v については固有値を λ とすると $Av = \lambda v$ であり、行列 A による変換が元のベクトル v の定数倍 λv となる

中心性 固有ベクトル中心性

- 重要なノードにつながっているノードは重要
- ノードの重要性を考慮した次数中心性とも考えられる
- 隣接行列の主固有値ベクトルに対応
 - $Av = \lambda v$ を満たす固有値 λ の中で最大値の固有値 λ_1 に対応する固有ベクトル
 - 各ノードの中心性を表すランダムな初期のベクトル x_0 に隣接行列をかけていくことで、べき乗法により主固有ベクトルが求まる

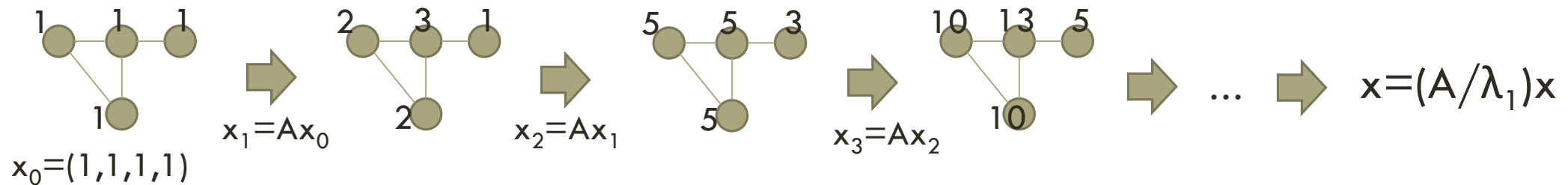


隣接行列をかけていくと各ノードは隣接ノードの中心性値を繰り返し受け取ることになる
*実際は値が発散しないように隣接行列をかけるごとに x の大きさを正規化

固有ベクトル中心性

中心性

- x は主固有ベクトルに収束 (べき乗法)
- x の各ノードに対応する要素の値がそのノードの固有ベクトル中心性



固有ベクトル中心性の問題

- 強連結ではない (隣接行列が既約ではない) 有向グラフの場合、固有ベクトル・固有値が一意に定まらない
- 非連結なグラフの時、最大連結成分に属さないノードの中心性は0になってしまう

中心性

PAGERANK

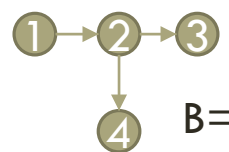
- 非連結なグラフや強連結でない有向グラフにも適用可能
- 隣接行列から推移確率行列Aを生成
 - あるノードから他のノードへ移動する確率を行列で表したものの
 - 各列の値（出次数）をその列の値（出次数）の総和で割る
 - 列の総和が0（そのノードの出次数が0）の時は列の総和が1になるようにする
 - $1/n$ をその列の値とする（任意のノードへ確率 $1/n$ で移動する）
 - もしくは、ノード自身への移動確率を1（そのノードに留まる）



中心性

PAGERANK

- 推移確率行列Aに基づくノード間の移動に加えて
 - 一定の確率で他ノードへテレポート移動する
 - 通常移動とテレポート移動の割合 α を指定 (例えばGoogleのPageRankでは0.85)



$$B = 0.85 * \begin{pmatrix} 0, 0, & 1/4, 1/4 \\ 1, 0, & 1/4, 1/4 \\ 0, 1/2, & 1/4, 1/4 \\ 0, 1/2, & 1/4, 1/4 \end{pmatrix} + 0.15 * \begin{pmatrix} 1/4, 1/4, 1/4, 1/4 \\ 1/4, 1/4, 1/4, 1/4 \\ 1/4, 1/4, 1/4, 1/4 \\ 1/4, 1/4, 1/4, 1/4 \end{pmatrix}$$

- 以上で得られた行列Bの主固有ベクトルの各要素が対応するノードのPageRank
 - 固有ベクトル中心性と同様に初期のベクトル x_0 に行列Bをかけていけば主固有ベクトルが求まる

$$x_0 = (1/n, 1/n, \dots, 1/n)$$

$$B = \alpha A + (1-\alpha)/n$$

$$Bx = x$$

* 確率行列のため最大固有値は1

PageRankは確率行列の定常分布を求めているとも考えられる
(グラフのランダムウォークとテレポートで到達しやすいノード)

(補) べき乗法

行列 A は対角化可能とし、固有値 λ_i が以下を満たすとする

$$|\lambda_1| > |\lambda_2| > \dots > |\lambda_n|$$

規格化した固有ベクトル v_i を基底とした時、初期ベクトル x_0 は定数 c_i を用いて以下のように固有ベクトル v_i の線型結合として表せる

$$x_0 = c_1 v_1 + c_2 v_2 + \dots + c_n v_n$$

初期ベクトル x_0 に行列 A を t 回かけていくと

$$A^t x_0 = c_1 \lambda_1^t v_1 + c_2 \lambda_2^t v_2 + \dots + c_n \lambda_n^t v_n$$

右辺を最大固有値 λ_1 で括ると

$$A^t x_0 = c_1 \lambda_1^t (v_1 + \frac{c_2}{c_1} (\frac{\lambda_2}{\lambda_1})^t v_2 + \dots + \frac{c_n}{c_1} (\frac{\lambda_n}{\lambda_1})^t v_n)$$

$|\lambda_1| > |\lambda_2| > \dots > |\lambda_n|$ より、十分大きい t においては

$$A^t x_0 = c_1 \lambda_1^t v_1$$

ネットワークの特徴 次数分布

- 大規模なネットワークにおいては小さい次数を持つ多数のノードと大きい次数を持つ少数のノードが存在 (power laws)

- ノードの次数が k である確率

- $P_k = n_k / n$ ($n_k =$ 次数 k であるノード数)

- 次数の分布

- $\ln p_k = -\alpha \ln k + c$

- $p_k = Ck^{-\alpha}$

- スケールフリーネットワーク

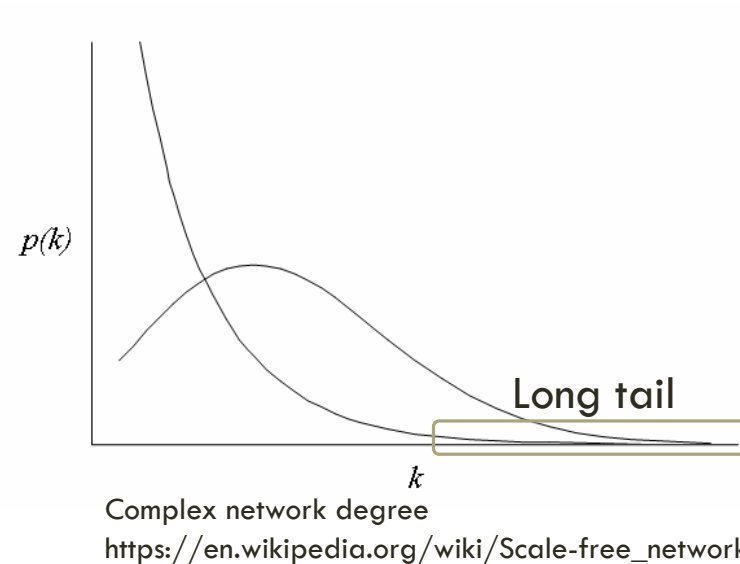
- power lawの次数分布に従うネットワーク

- 現実の多くのネットワークに当てはまる

- **Barabási-Albert**モデル

- 優先的選択：追加されるノードは、すでにネットワークにあるノードのうち次数の高いものに対して優先的にリンクをはる

- 多くのリンクを持ったノードであるハブの出現、スケールフリーネットワークの形成



ネットワークの特徴 パス長とクラスタリング係数

平均パス長

- ネットワークのすべてのノード間の最短経路長の平均
- 現実の多くのネットワークでは平均パス長は小さい
 - 60年代の社会実験：友人を介しての手紙の転送は6次の隔たり

平均クラスタリング係数

- ノードの隣接ノード同士がつながっている確率
 - ローカルクラスタリング係数
 - $c_i = \text{ノード}i\text{の隣接ノード同士のつながり数} / \text{ノード}i\text{の隣接ノード同士の組み合わせ数}$
- $C = \sum C_i / N$
- 現実の多くのネットワークではクラスタリング係数は大きい
 - 友人が友人同士である

スモールワールドネットワークの特徴

スモールワールドネットワーク

- 現実の多くのネットワークでは
 - 平均パス長は小さい
 - クラスタリング係数は大きい
- ネットワークのあるコミュニティに属してノードは、異なるコミュニティに属している他のノードとも比較的小さい距離でつながっている
 - 友達が友達同士であり近傍は密なつながり
 - 遠方の人とも程よい距離でつながる
- Watts-Strogatzモデル
 - クラスタリング係数が大きく、平均パス長も大きいネットワークから始める
 - 各リンクの片端を一定の確率 p で、ランダムに選んだ別のノードへつなぎかえる
 - p を変化させながらリンクのつなぎかえをするとスモールワールドネットワークが形成

著作権の都合により
ここに挿入されていた画像を削除しました

Duncan J. Watts, Steven H. Strogatz, "Collective dynamics of 'small-world' networks", Nature vol. 393, pages 440–442 (04 June 1998), Fig. 1
<https://www.nature.com/articles/30918#rightslink>

ネットワークの特徴

	Network	Type	n	m	c	S	ℓ	α	C	C_{ws}	
Social	Film actors	Undirected	449 913	25 516 482	113.43	0.980	3.48	2.3	0.20	0.78	0.
	Company directors	Undirected	7 673	55 392	14.44	0.876	4.60	-	0.59	0.88	0.
	Math coauthorship	Undirected	253 339	496 489	3.92	0.822	7.57	-	0.15	0.34	0.
	Physics coauthorship	Undirected	52 909	245 300	9.27	0.838	6.19	-	0.45	0.56	0.
	Biology coauthorship	Undirected	1 520 251	11 803 064	15.53	0.918	4.92	-	0.088	0.60	0.
	Telephone call graph	Undirected	47 000 000	80 000 000	3.16			2.1			
	Email messages	Directed	59 812	86 300	1.44	0.952	4.95	1.5/2.0		0.16	
	Email address books	Directed	16 881	57 029	3.38	0.590	5.22	-	0.17	0.13	0.
	Student dating	Undirected	573	477	1.66	0.503	16.01	-	0.005	0.001	-0.
	Sexual contacts	Undirected	2 810					3.2			
Information	WWW nd . edu	Directed	269 504	1 497 135	5.55	1.000	11.27	2.1/2.4	0.11	0.29	-0.
	WWW AltaVista	Directed	203 549 046	1 466 000 000	7.20	0.914	16.18	2.1/2.7			
	Citation network	Directed	783 339	6 716 198	8.57			3.0/-			
	Roget's Thesaurus	Directed	1 022	5 103	4.99	0.977	4.87	-	0.13	0.15	0.
	Word co-occurrence	Undirected	460 902	16 100 000	66.96	1.000		2.7		0.44	
Technological	Internet	Undirected	10 697	31 992	5.98	1.000	3.31	2.5	0.035	0.39	-0.
	Power grid	Undirected	4 941	6 594	2.67	1.000	18.99	-	0.10	0.080	-0.
	Train routes	Undirected	587	19 603	66.79	1.000	2.16	-		0.69	-0.
	Software packages	Directed	1 439	1 723	1.20	0.998	2.42	1.6/1.4	0.070	0.082	-0.
	Software classes	Directed	1 376	2 213	1.61	1.000	5.40	-	0.033	0.012	-0.
	Electronic circuits	Undirected	24 097	53 248	4.34	1.000	11.05	3.0	0.010	0.030	-0.
	Peer-to-peer network	Undirected	880	1 296	1.47	0.805	4.28	2.1	0.012	0.011	-0.
Biological	Metabolic network	Undirected	765	3 686	9.64	0.996	2.56	2.2	0.090	0.67	-0.
	Protein interactions	Undirected	2 115	2 240	2.12	0.689	6.80	2.4	0.072	0.071	-0.
	Marine food web	Directed	134	598	4.46	1.000	2.05	-	0.16	0.23	-0.
	Freshwater food web	Directed	92	997	10.84	1.000	1.90	-	0.20	0.087	-0.
	Neural network	Directed	307	2 359	7.68	0.967	3.97	-	0.18	0.28	-0.

M. E. J. Newman 『Networks : An Introduction』
 Chapter 8 The Large-scale structures of networks,
 Table 8.1, Oxford Univ Pr (2010/5/20)
<http://www.oxfordscholarship.com/view/10.1093/acprof:oso/9780199206650.001.0001/acprof-9780199206650-chapter-8>

コミュニティ抽出

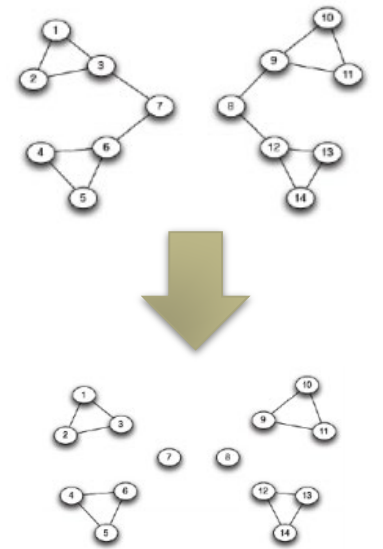
- ネットワークのノードをコミュニティ（グループ、クラスター）に分割する
- コミュニティ内のノードはより密につながっているが、コミュニティ間のノードはより疎なつながりを持つ
- ネットワークの全体構造を把握できる



コミュニティ抽出 エッジ媒介中心性コミュニティ抽出 (GIRVAN-NEWMAN法)

エッジ媒介中心性

- ネットワークのノード間の最短経路にそのエッジが含まれる度合い
- エッジ媒介性の高いエッジを切断していけば、コミュニティが抽出できる
 - ネットワークの各エッジの媒介中心性を計算する
 - 最もエッジ媒介中心性が高いエッジをネットワークから取り除く
 - 再びネットワークの各エッジの媒介中心性を計算する
 - 最もエッジ媒介中心性が高いエッジをネットワークから取り除く
 - ...繰り返し
 - エッジが1つもなくなり、ノード1つ1つがコミュニティとなったら終了
- 最終的にノード間の階層関係（デンドログラム）を抽出できる
- 適切なコミュニティ数を決めるには評価関数（コミュニティのよさを決める）が必要
- 計算量 $O(mn(m+n))$ 、ネットワークが疎なら $O(n^3)$



コミュニティ抽出

モジュラリティ

HomophilyもしくはAssortative mixing

- 人は同種・同質の人とつながりを持つ傾向がある
- コミュニティを同種・同質のノードの集合と考えると同一コミュニティ内のノード間のリンクは多い

モジュラリティ

- コミュニティ内のリンク数の期待値に対して実際にどれぐらいのリンクが存在するか
- 同じコミュニティのノード間の実際のリンク数がリンク数の期待値より多ければ正となる

モジュラリティ Q は、ネットワークのリンク数を m 、隣接行列を A 、ノード i の次数を k_i とした時、以下で定義されます。

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(c_i, c_j)$$

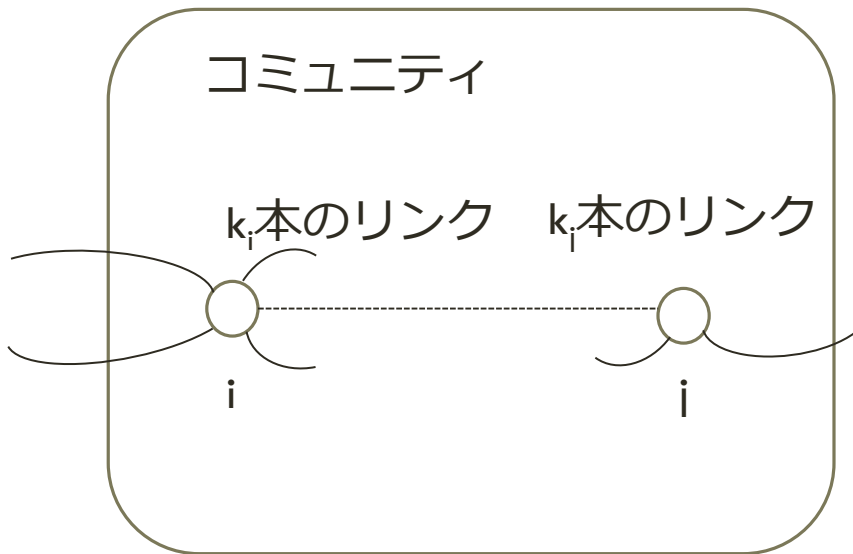
ノード i のコミュニティを c_i とすると $\delta(c_i, c_j)$ はノード i と j が同じコミュニティの時1、異なるコミュニティの時0となります。

ノード i のあるリンクの片方がノード j である確率は $k_i/2m$

ノード i には k_i 本リンクがあるのでノード i, j のリンクの期待値は $(k_i k_j)/2m$

コミュニティ抽出

モジュラリティ



ノード*i*のあるリンク片端がノード*i*である確率は $k_i/2m$
(ネットワーク全体では $2m$ のリンク片端があり
それらの方端がノード*i* (次数 k_i)である確率は $k_i/2m$)

ノード*i*には k_i 本リンクがあるのでノード*i-j*のリンクの
期待値は $(k_i k_j)/2m$

一方、ノード*i-j*の実際のリンクの有無は隣接行列の A_{ij}

モジュラリティの $A_{ij} - (k_i k_j)/2m$ はリンクの期待値に対する
実際のリンクの偏りを評価している

モジュラリティが大きい = コミュニティ内のリンクが期待される
より多い

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(c_i, c_j)$$

モジュラリティ

- モジュラリティは以下のようにも表せる
 - コミュニティ c について m_c をコミュニティ内のリンク数、 k_c をコミュニティ内のノードの次数の和とすると

$$Q = \sum_{c=1}^n \left(\frac{m_c}{m} - \left(\frac{k_c}{2m} \right)^2 \right)$$

- リンクのランダムなつながりと比較した各コミュニティ内のリンク密度
- モジュラリティはコミュニティ抽出の評価関数として使える
 - モジュラリティが大きくなるようにコミュニティ抽出をすればよい
 - しかし、任意のコミュニティの組み合わせを考え評価していくのは時間がかかる
 - 実用的にはヒューリスティックなモジュラリティの最大化方法が使われる

コミュニティ抽出 モジュラリティ最大化法

Greedy algorithms [Claust 04]

- 各ノードをそれぞれ自身をコミュニティとしてノード数と同じコミュニティ数から始める
- 任意のノードペアについてそれらをコミュニティとした時のモジュラリティの増減を調べる

$$\Delta Q = \frac{\text{コミュニティ}i\text{とコミュニティ}j\text{の間のリンク数}}{m} - \frac{\text{コミュニティ}i\text{のノードの次数の和} * \text{コミュニティ}j\text{のノードの次数の和}}{2m^2}$$

- 増減が最大となるノードペアを新しいコミュニティとする
- すべてのノードが1つのコミュニティになるまで繰り返す
- 最初のステップまで振り返り、モジュラリティが最大だったステップでコミュニティ分割

計算量は $O(n(m+n))$, ネットワークが疎なら $O(n^2)$

より効率的な方法もある: Louvain法 [Blondel 08] (計算量は $O(m)$)

参考書

著作権の都合により
ここに挿入されていた画像を削除しました

書籍『Networks : An Introduction』表紙

M. E. J. Newman 著

Oxford Univ Pr (2010/5/20)

<http://www.oxfordscholarship.com/view/10.1093/acprof:oso/9780199206650.001.0001/acprof-9780199206650>

著作権の都合により
ここに挿入されていた画像を削除しました

書籍『Network Science』表紙

Albert-László Barabási 著

Cambridge University Press (2016/7/21)

<http://networksciencebook.com/>

ツールとデータセット

Pythonのネットワーク分析ライブラリ

- NetworkX
 - <https://networkx.github.io/>

ネットワーク分析ライブラリ（スタンフォード大学）

- SNAP
 - <http://snap.stanford.edu/>

Stanford Large Network Dataset Collection

- <http://snap.stanford.edu/data/>

Network Data by M. Newman

- <http://www-personal.umich.edu/~mejn/netdata/>