



Networkモデル

Random network
モデリングとシミュレーション

2019年度

要素の関係をグラフ・ネットワークとして捉える

- 要素と要素の繋がり方
- 実在するネットワークに共通的性質
- なぜそのような構造なのか
 - ネットワーク形成メカニズム
 - 構造の利点
 - 最適化？
- ネットワーク上の過程

実際のネットワークの例

- インターネット
- Webのリンク
- 知人の繋がり
- 映画俳優の共演関係
- 論文を共著する関係
- 食物連鎖
- たんぱく質の反応連鎖
- 電力網

ネットワークの例

- ▶ JR九州路線図

- ▶ <https://www.jrkyushu.co.jp/railway/routemap/>

- ▶ 佐賀市営バス路線図

- ▶ <http://www.bus.saga.saga.jp/pdf/rosenzu011001.pdf>

- ▶ SINET

- ▶ <https://www.sinet.ad.jp/aboutsinet>

- ▶ The Internet map

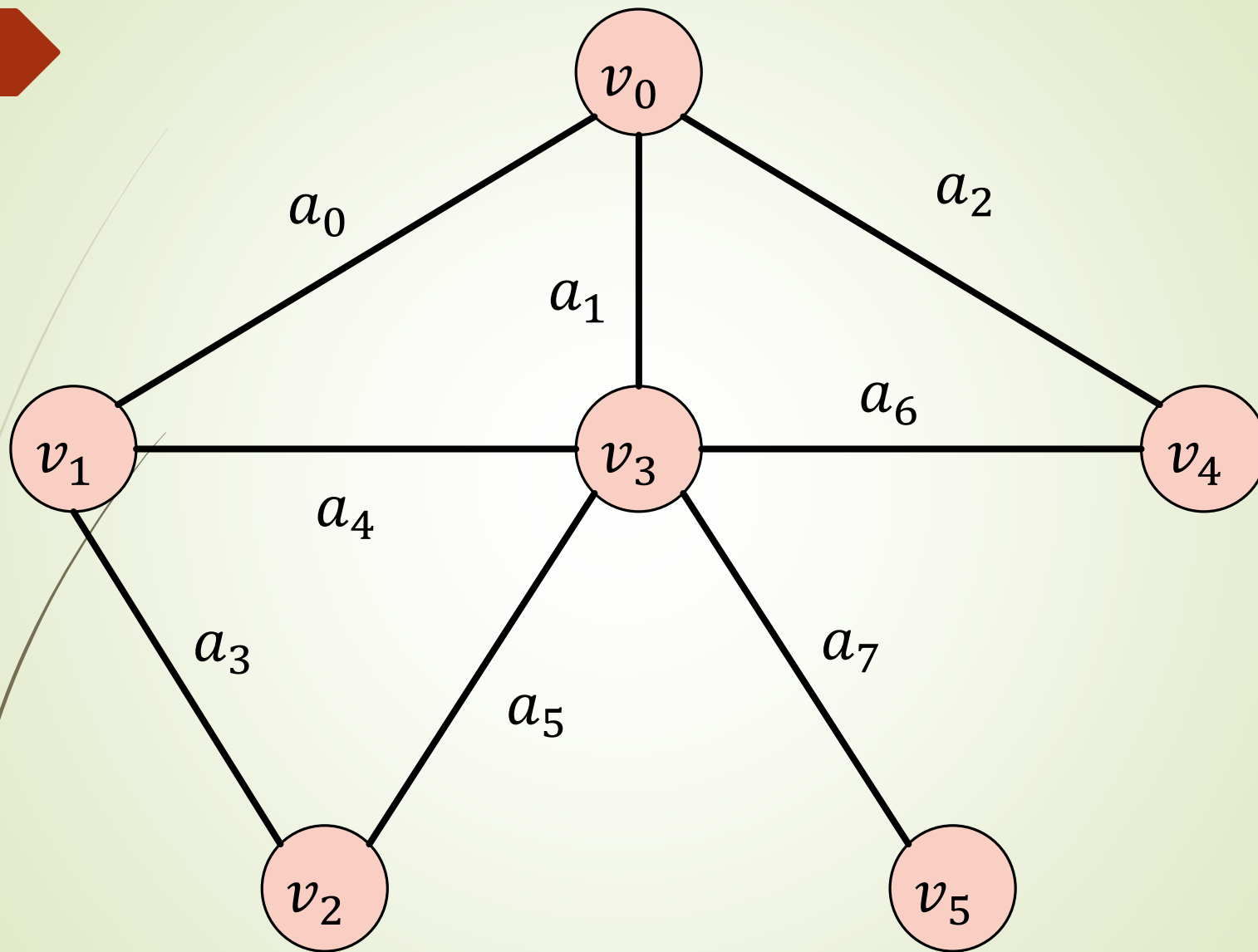
- ▶ <https://internet-map.net/>

Network (Graph)

- 頂点 (node)を辺 (edge)で結んだもの
- 結び方そのものに関心

グラフの記述

- ▶ V : 頂点の集合
- ▶ A : 辺の集合
- ▶ $\delta: V \rightarrow 2^A$: 指定した頂点に接続する辺の集合
- ▶ $\partial: A \rightarrow V \times V$: 辺の両端の頂点



$$V = \{v_0, v_1, v_2, v_3, v_4, v_5\}$$

$$A = \{a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7\}$$

$$\delta(v_0) = \{a_0, a_1, a_2\} \quad \delta(v_1) = \{a_0, a_3, a_4\} \quad \delta(v_2) = \{a_3, a_5\}$$

$$\delta(v_3) = \{a_1, a_4, a_5, a_6, a_7\} \quad \delta(v_4) = \{a_2, a_6\} \quad \delta(v_5) = \{a_7\}$$

$$\partial(a_0) = (v_0, v_1) \quad \partial(a_1) = (v_0, v_3) \quad \partial(a_2) = (v_0, v_4)$$

$$\partial(a_3) = (v_1, v_2) \quad \partial(a_4) = (v_1, v_3) \quad \partial(a_5) = (v_2, v_3)$$

$$\partial(a_6) = (v_3, v_4) \quad \partial(a_7) = (v_3, v_5)$$

グラフの記述 隣接行列

$$A_{ij} = \begin{cases} 1 & \text{頂点}i\text{と頂点}j\text{が結ばれている} \\ 0 & \text{その他} \end{cases}$$

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Network のプログラムとしての の実装

AbstractNetwork class

String label

Int numNode

List<Node> nodes

Node class

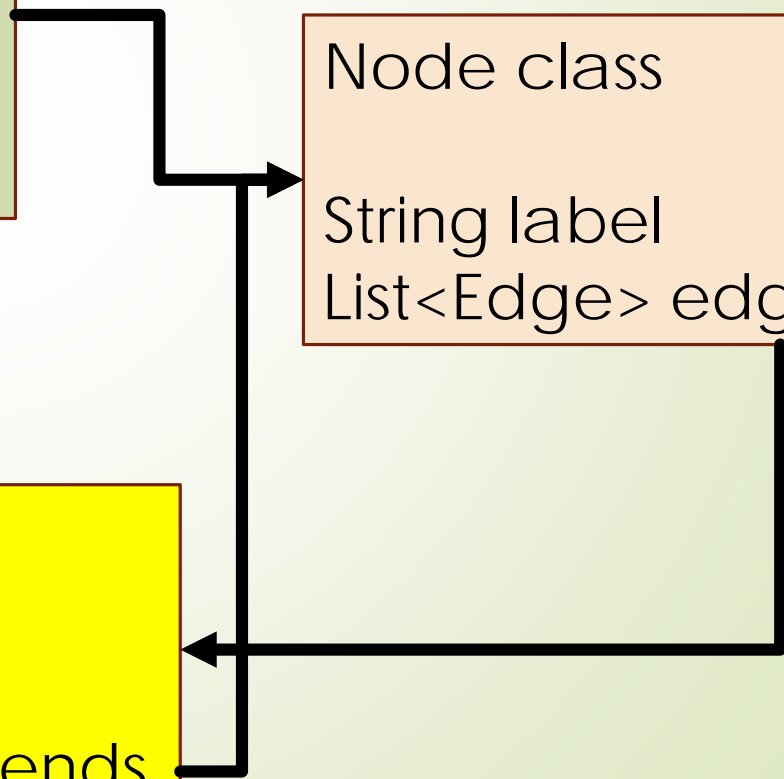
String label

List<Edge> edges

Edge class

String label

List<Node> ends



Erdős-Rényi Random Graph

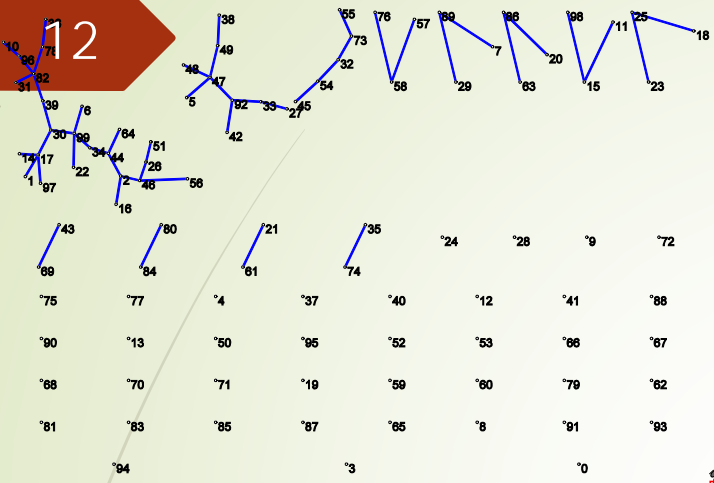
- ▶ 一定数 N の頂点を用意する
- ▶ 一定数 L の辺をランダムに選んだ頂点の組に接続する: $0 \leq p < 1$

$$L = p \frac{N(N-1)}{2}$$

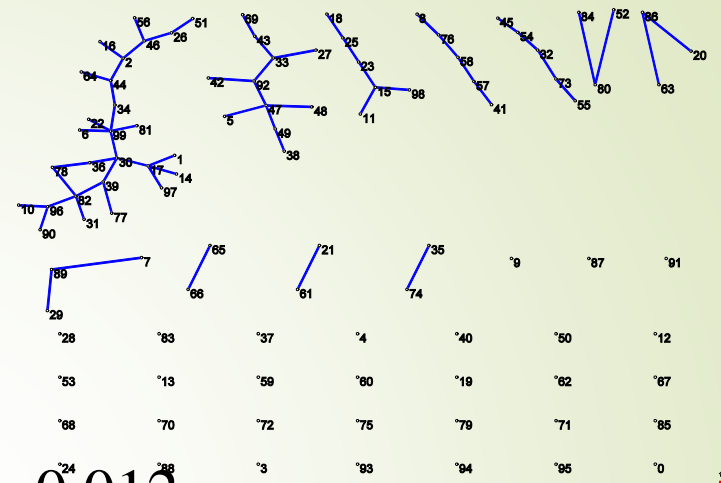
- ▶ 単純グラフであることは求めない
 - ▶ 頂点 v から頂点 v への辺：孤立辺
 - ▶ 頂点 v から頂点 w への辺が複数：並列辺

$N = 100$

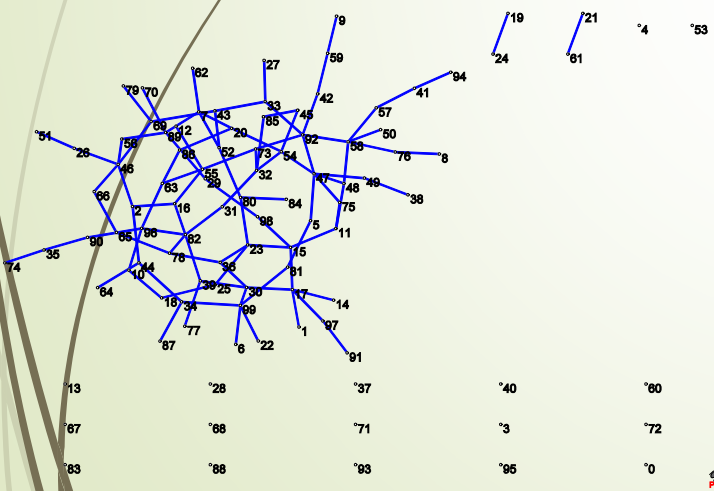
12



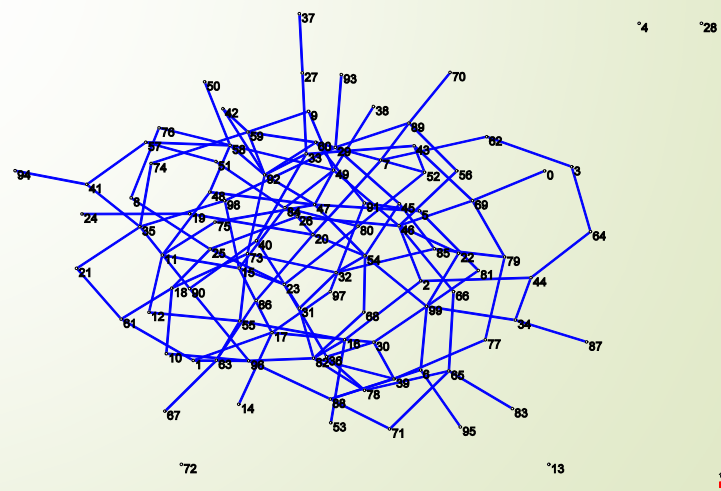
$p = 0.01$



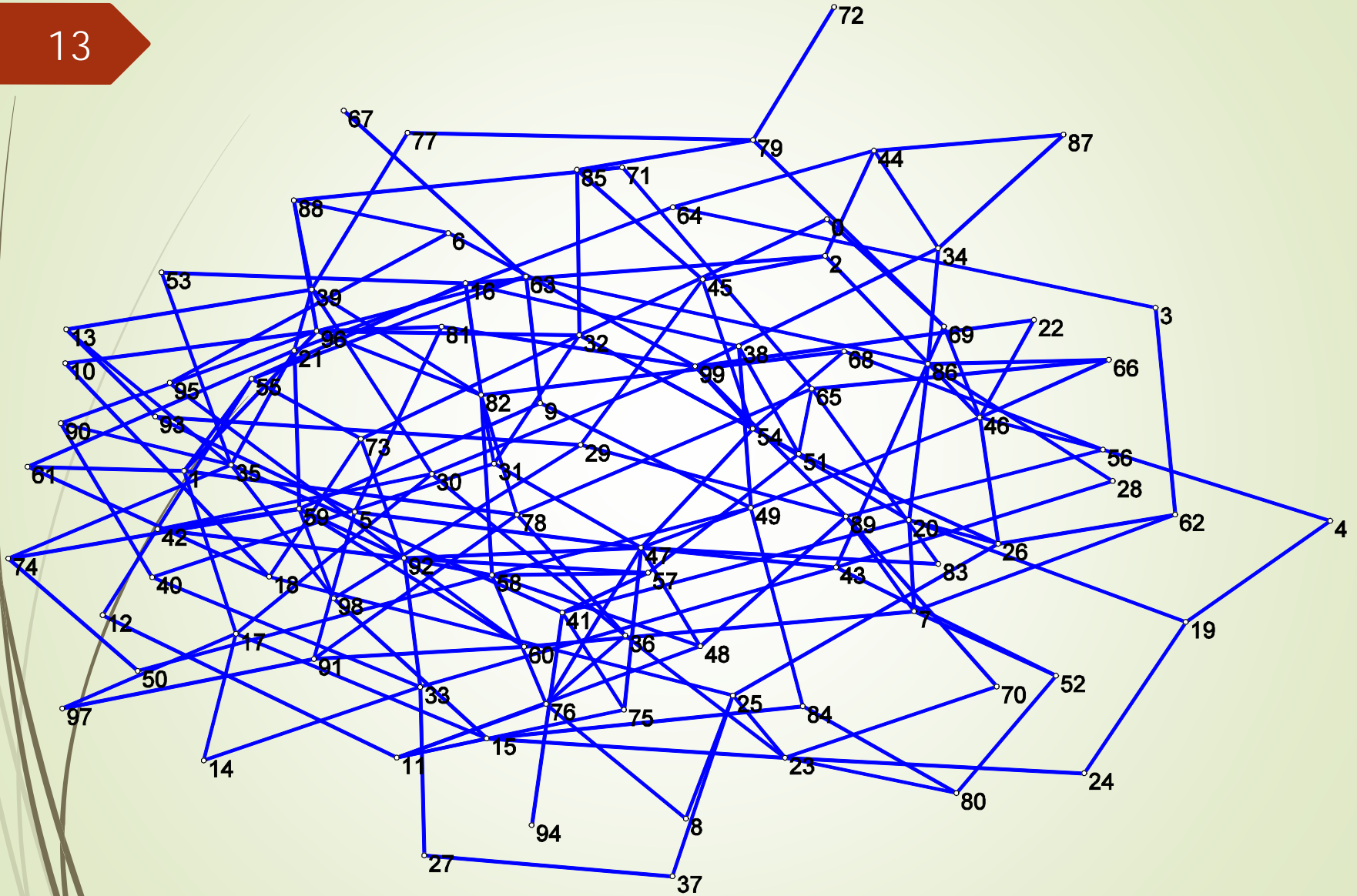
$p = 0.012$



$p = 0.02$



$p = 0.03$



$p = 0.04$

Giant Component

- ▶ 辺を持つ確率 p が小さい
 - ▶ 連結されていない小さいネットワークに分かれている
 - ▶ 平均距離は短い:連結の場合だけを考える
- ▶ 辺を持つ確率 p が非常に大きい
 - ▶ 全体が連結されている
 - ▶ 多くの頂点と直接結ばれている:平均距離は短い
- ▶ 中間の p で何が起こる?
- ▶ 全頂点とほぼ同じサイズの連結成分をgiant componentと呼ぶ

- ▶ 任意の頂点がgiant componentに属していない確率 u
- ▶ giant componentに属していないためには
 - ▶ 全く隣接頂点がない
 - ▶ 全ての隣接頂点が、giant componentに属さない

- 他の頂点と繋がっていない確率： $1 - p$
- 隣接頂点は繋がっているが、 *giant Component* に属さない： pu
 - k 個の頂点とつながり、その全てが *giant component* に属さない確率

$$\binom{N-1}{k} (1-p)^{N-1-k} (pu)^k$$

$$\begin{aligned}u &= (1-p)^{N-1} + \sum_{k=1}^{N-1} \binom{N-1}{k} (1-p)^{N-1-k} (pu)^k \\&= \sum_{k=0}^{N-1} \binom{N-1}{k} (1-p)^{N-1-k} (pu)^k = (1-p+pu)^{N-1} \\&= \left[1 - \frac{c}{N-1} (1-u) \right]^{N-1} \\c &\equiv (N-1)p\end{aligned}$$

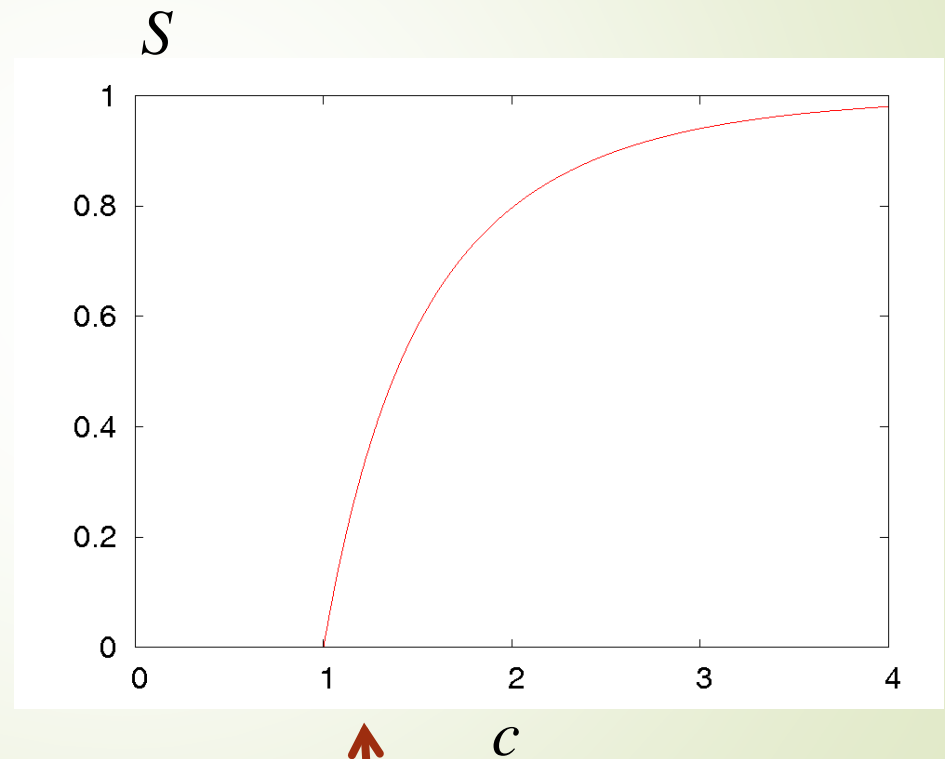
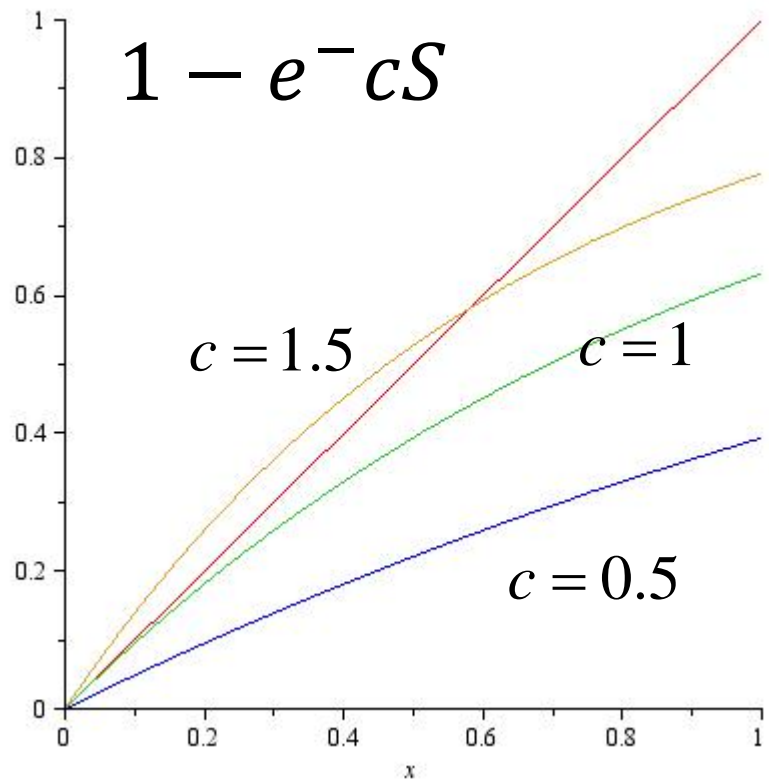
$$\ln u = (N-1) \ln \left[1 - \frac{c}{N-1} (1-u) \right]$$

$$\simeq (N-1) \left[-\frac{c}{N-1} (1-u) + O\left((N-1)^{-2}\right) \right] = -c(1-u)$$

$$u = e^{-c(1-u)}$$

- ▶ Giant componentに入っている節の比 : $S = 1 - u$
 - ▶ $S = 1 - e^{-cS}$
- ▶ S に対する方程式に対して、自己無撞着(self-consistent)に数値的解を得る

$c > 1$ で曲線と直線の交点が発生



突然giant componentが出現

Simulation

- ▶ 頂点を連結した頂点の集合に分類する
- ▶ 一番大きな集合のサイズが p に対して変化する様子を調べる

幅優先探索

```
BFS( $U, v_0$ ) { //  $U$ 対象頂点の集合、 $v_0$ 始点
   $L = \emptyset$  // 処理済み頂点集合
   $Q$  // 調査すべき頂点の待ち行列
   $Q \leftarrow v_0$  //  $Q$ に $v_0$ を追加
  while ( $Q \neq \emptyset$ ) {
     $v = Q.\text{poll}$  //  $Q$ の先頭要素を取り出し
    forall ( $a \in \delta^+ v$ ) { //  $a$ から出る全ての弧
       $w = \partial^- a$  //  $a$ の反対側の頂点
      if ( $w \notin L \wedge w \notin Q$ ) {  $Q \leftarrow w$  }
    }
     $L \leftarrow L \cup \{v\}$  //  $L$ へ $v$ を追加
  }
  return  $L$  // 始点から到達可能な頂点の集合
}
```

頂点を連結集合（クラスタ）へ 分割：幅優先探索の応用

- $U \subseteq V$: 未分類の頂点の集合。初期値は $U = V$
- $C \subseteq 2^V$: クラスタの集合。初期値は $C = \emptyset$

```
while ( $U \neq \emptyset$ ){  
     $w \in U$  //  $U$ から選ぶ  
     $L = \text{BSF}(U, w)$  //  $w$ から到達可能な頂点の集合  
     $U \leftarrow U \setminus L$  //  $U$ から $L$ を削除  
     $C \leftarrow C \cup \{L\}$   
}
```