



# 異なるseedで生成した 複雑ネットワークの特徴量

日本大学文理学部情報科学科

谷 聖一研究室

市川慶徳 湯浅真

# 目次

1. 複雑ネットワーク
2. 背景・演習の目的
3. 修正Configurationモデル
4. 特徴量
5. 実験方法
6. 実験結果
7. 考察・今後の課題



# 1. 複雑ネットワーク

1.1 複雑ネットワークとは

1.2 グラフの定義

1.3 複雑ネットワーク研究

1. 複雑ネットワーク  
1.1 複雑ネットワークとは

研究対象

複雑ネットワーク

# 1. 複雑ネットワーク

## 1.1 複雑ネットワークとは

- ネットワークとは  
頂点と枝から構成され、つながりや関係を表したものの

鉄道網

航空網

SNS

脳

食物網

インターネット

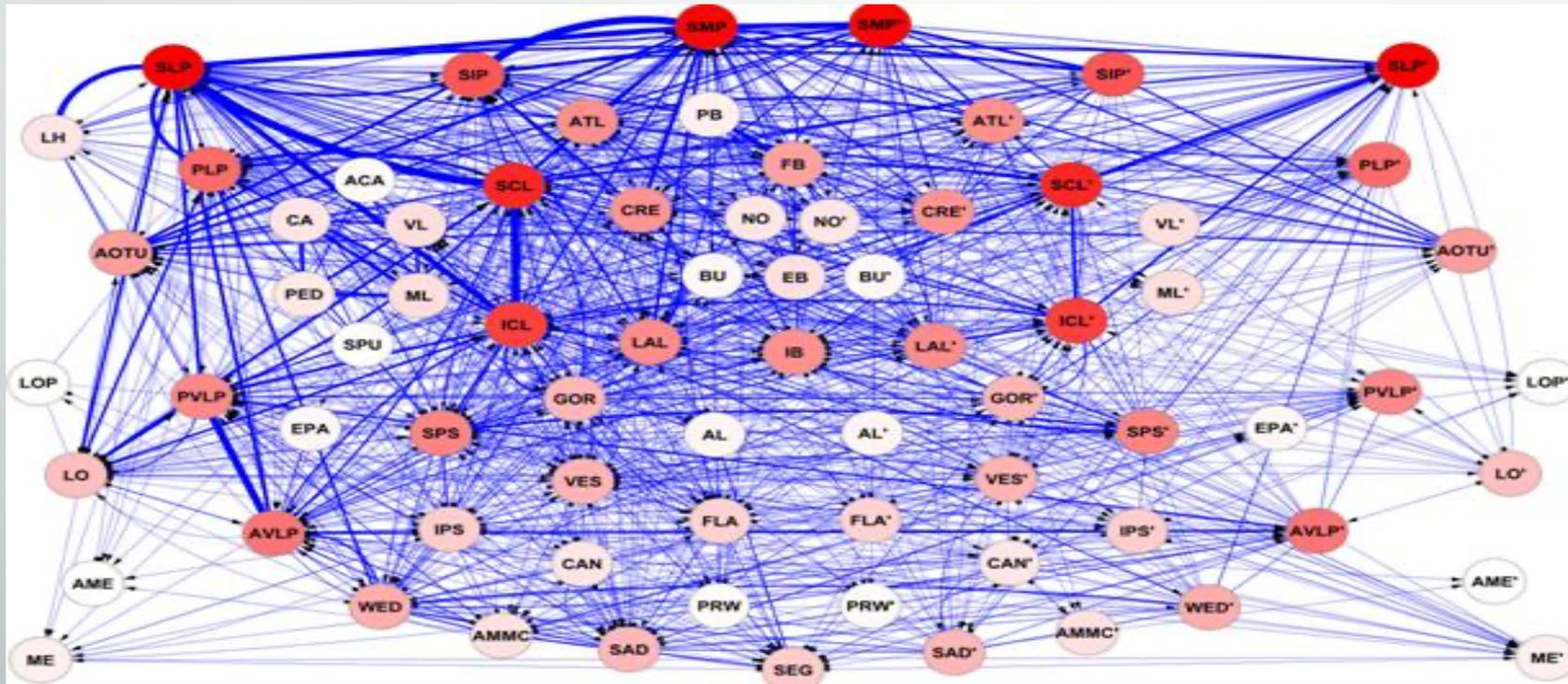
人間関係



# 1.複雑ネットワーク

## 1.1 複雑ネットワークとは

例：キイロショウジョウバエの脳神経のモデル



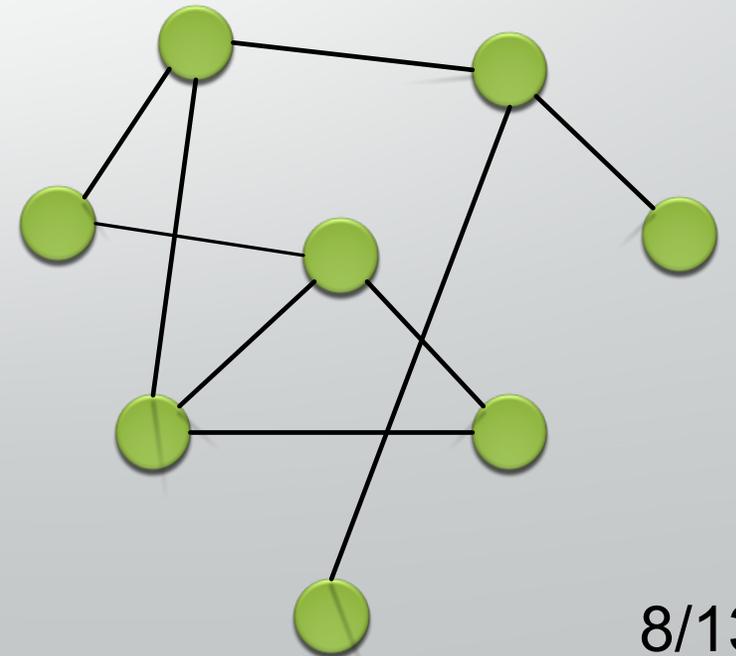
出典: SOHO 脳は神経ユニットチームの協働で動いていた/キイロショウジョウバエのコネクトーム  
<http://springstreet.blog69.fc2.com/blog-entry-2940.html>

# 1. 複雑ネットワーク

## 1.1 複雑ネットワークとは

- 複雑ネットワークとは
  - ・・・ 現実世界に存在する構造が複雑なネットワークのこと

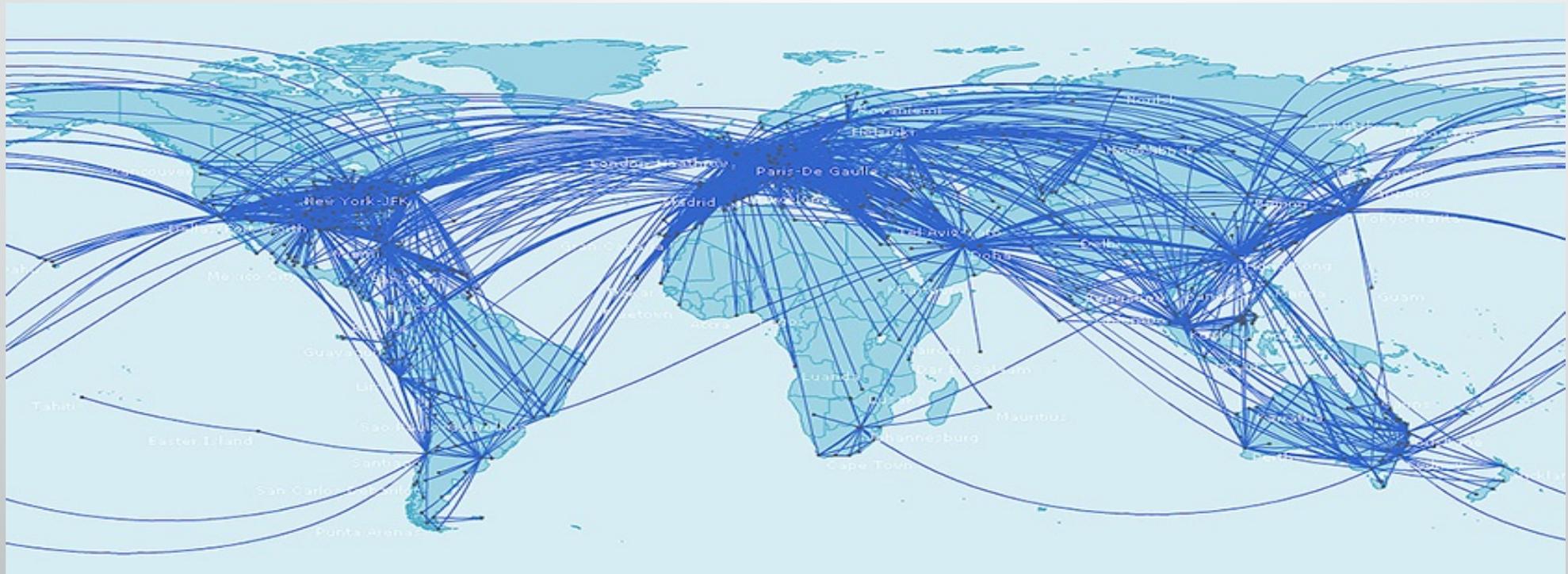
ネットワークをグラフとして  
表現することで  
数学的な解析の対象にできる



# 1.複雑ネットワーク

## 1.1 複雑ネットワークとは

例：航空網 ワンワールド アライアンス



出典: JAL [https://www.jal.co.jp/oneworld/global\\_network.html](https://www.jal.co.jp/oneworld/global_network.html)

# 1.複雑ネットワーク

## 1.1 複雑ネットワークとは

例：航路網 国際コンテナ船航路ネットワーク



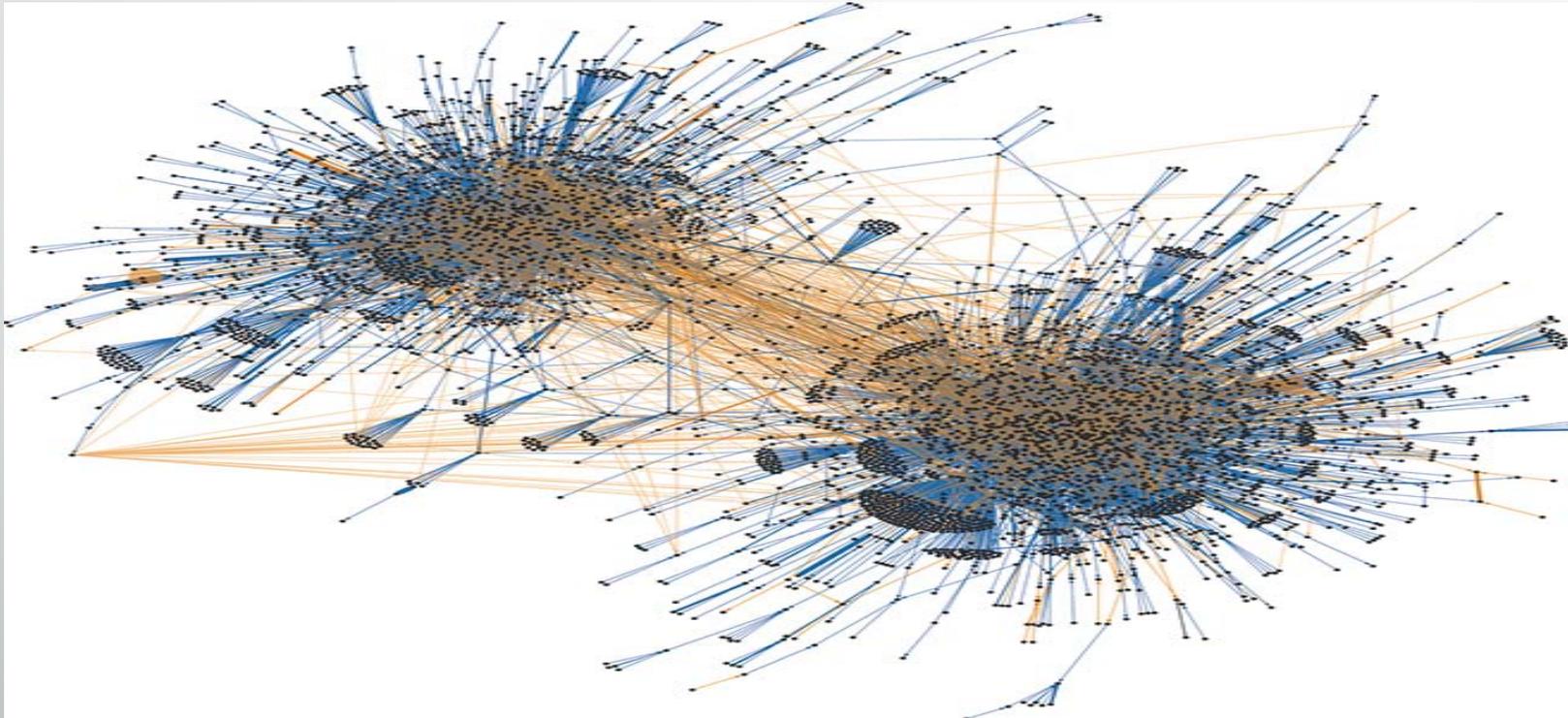
出典: 海上技術安全研究所

[http://www.nmri.go.jp/institutes/distribution\\_system/logistics\\_research/research/research\\_1.html](http://www.nmri.go.jp/institutes/distribution_system/logistics_research/research/research_1.html)

# 1.複雑ネットワーク

## 1.1 複雑ネットワークとは

例 : twitter ハッシュタグを利用した情報拡散の様子



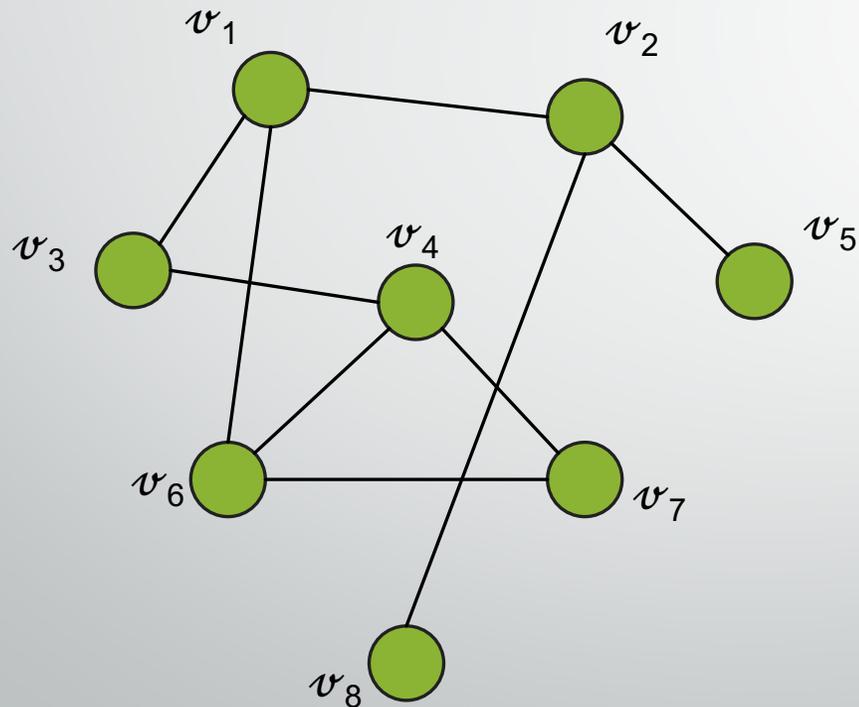
出典: WIRED 5回連続講義 : 『ビジュアル・コンプレキシティ』を読む -  
データ・ヴィジュアルライゼーション講座【2】

<http://wired.jp/2012/05/22/visual-complexity-2/>

# 1.複雑ネットワーク

## 1.2 グラフの定義

- グラフの定義



頂点数を $N$ , 枝数を $M$ とする

- 頂点集合 $V = \{v_1, v_2, \dots, v_N\}$
- 枝  $e \in E \times E$   $e = (v_a, v_b)$
- 枝集合 $E = \{e_1, e_2, \dots, e_M\}$
- グラフ $G = (V, E)$

# 1. 複雑ネットワーク

## 1.3 複雑ネットワーク研究

現実世界に存在するネットワークの解析

数学的・統計的なネットワークの解析

ネットワークの生成・シミュレーション→性質の解析

# 1. 複雑ネットワーク

## 1.3 複雑ネットワーク研究

### 数学的・統計的なネットワークの解析(例)

- ネットワーク上での情報伝搬に関する論文の主張

[TTMO]

Hiroshi Toyoizumi, Seiichi Tani, Naoto Miyoshi, Yoshio Okamoto

「Reverse preferential spread in complex networks」

Phys. Rev. E 86, 021103(2012)

ある伝搬モデルで情報を流す際, **Uncorrelated性**を持つスケールフリーネットワーク上では, 効率良くネットワーク全体に情報伝搬するには, **次数の小さい頂点に優先して流す方法が最適である**

# 1. 複雑ネットワーク

## 1.3 複雑ネットワーク研究

### ネットワークの生成・シミュレーション→性質の解析(例)

- 飯田佳徳 丸山隆太  
「BAモデルおよびConfigurationモデルで生成したネットワークのUncorrelated性について」(2016)

ネットワーク生成手法であるBAモデル及びConfigurationモデルを用いてネットワークを生成し,伝搬実験を行った

# 1. 複雑ネットワーク

## 1.3 複雑ネットワーク研究

- 複雑ネットワークを生成する手法の例
  - BAモデル
  - Configurationモデル → 本演習で使用
  - 頂点コピーモデル
  - Gohモデル



## 2. 背景・演習の目的

2.1 演習の背景

2.2 演習の目的

## 2.背景・演習の目的

### 2.1 演習の背景

#### 昨年の演習

Uncorrelated性を持つネットワークを生成できるとされている  
ヒューリスティックを用いてスケールフリーネットワークを生成し、  
伝搬実験を行った

## 2.背景・演習の目的

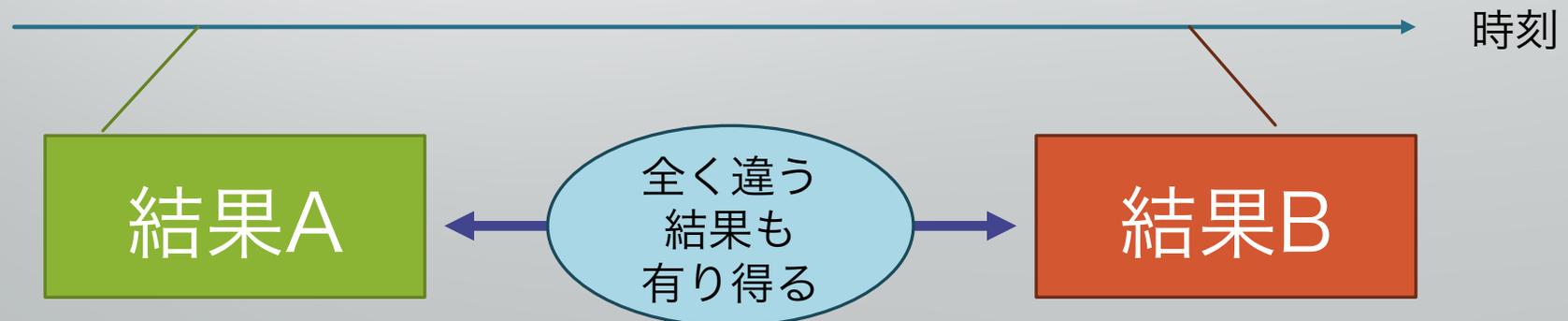
### 2.1 演習の背景

- 昨年のConfigurationモデルで生成したネットワークの生成個数
  - ネットワーク生成方法：9種類(生成手法3種類×べき指数3種類)
  - 1つの生成方法のネットワークの生成個数：100個
    - ネットワーク総数 900個
- 各ネットワークに対して伝搬シミュレーションを100回実行
  - 伝搬シミュレーション回数 90000回

## 2.背景・演習の目的

### 2.1 演習の背景

- 1つの生成方法で生成したネットワーク数 **100個**  
→生成個数は妥当なのか？
- 昨年のネットワークを生成するプログラムでは乱数を使用しており、seedを**現在時刻**として乱数を生成していた
- 生成されたネットワーク群の性質は使用したseed群により、大きく異なるかもしれない



## 2.背景・演習の目的

### 2.2 演習の目的

乱数生成時に使用するseedを変更し、  
実験結果がseedに依存しないか検証する

## 2.背景・演習の目的

### 2.2 演習の目的

- 検証方法

1. 統計的な解析

手法例：区間推定, 分散分析, 時系列解析, バッチ平均法など

2. 計算機実験(シミュレーション)

本演習では「計算機実験」を実施

## 2.背景・演習の目的

### 2.2 演習の目的

- 特定の時刻をseedに使った場合に結果が偏る可能性  
→seedの値にある連続した整数値100個を用いてそれぞれを比較する  
0~99をseedの値として設定
- seedにある連続した整数値100個を用いた場合でも偏りがあるのでは？  
→上の値から大きく離れた連続した整数値100個をseedとして実験  
17000~17099をseedの値として設定

## 2.背景・演習の目的

### 2.2 演習の目的

seed = 現在時刻の  
ネットワークの特徴量

seed = 00~99の  
ネットワークの特徴量

seed = 17000~17099の  
ネットワークの特徴量

これらの特徴量を比較し、結果に大きな差がないか検証



seedによって実験結果に偏りが無いことを証明する



## 3. 修正Configurationモデル

3.1 スケールフリー性

3.2 Configurationモデル

3.3 Uncorrelated性

3.3 修正Configurationモデル

## 3.修正Configurationモデル

### 3.1 スケールフリー性

- 複雑ネットワークに見られる性質  
スケールフリー性

# 3.修正Configurationモデル

## 3.1 スケールフリー性

- べき則

$$p(k) \propto k^{-\gamma} \quad (\gamma: \text{べき指数})$$

多くの複雑ネットワークにおいて,次数分布がべき則に従っていることが知られている

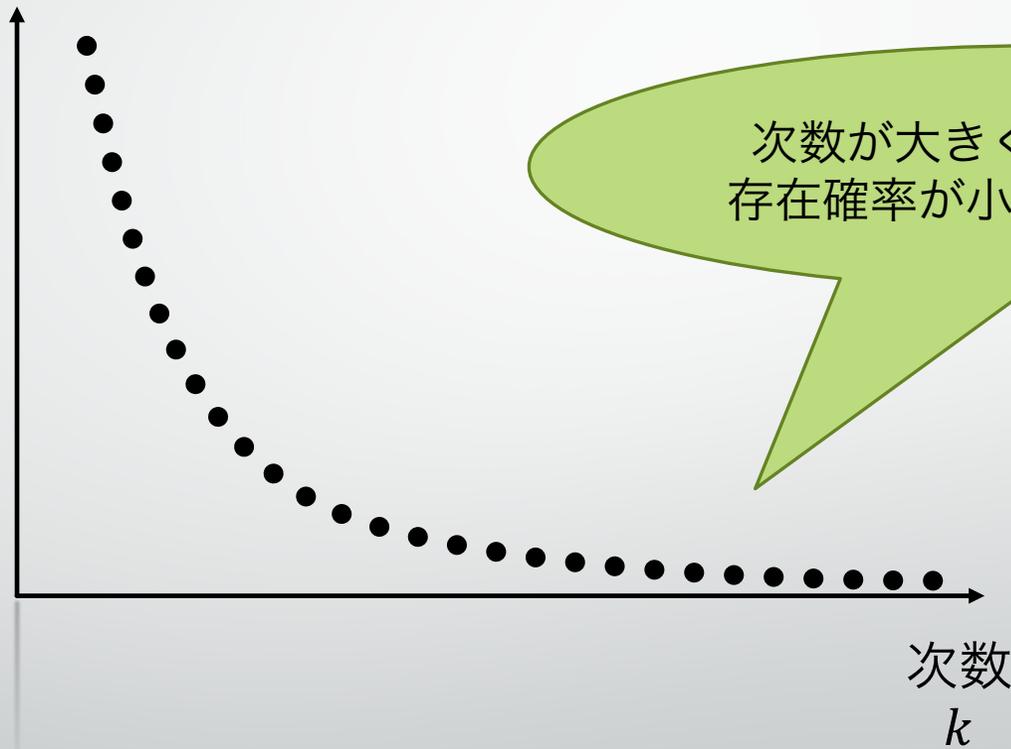
現実世界で見られる複雑ネットワークのべき指数の多くは2.0~3.0の間で観測される

### 3.修正Configurationモデル

## 3.1 スケールフリー性

べき則： $p(k) \propto k^{-\gamma}$

次数 $k$ が存在する確率  
 $p(k)$

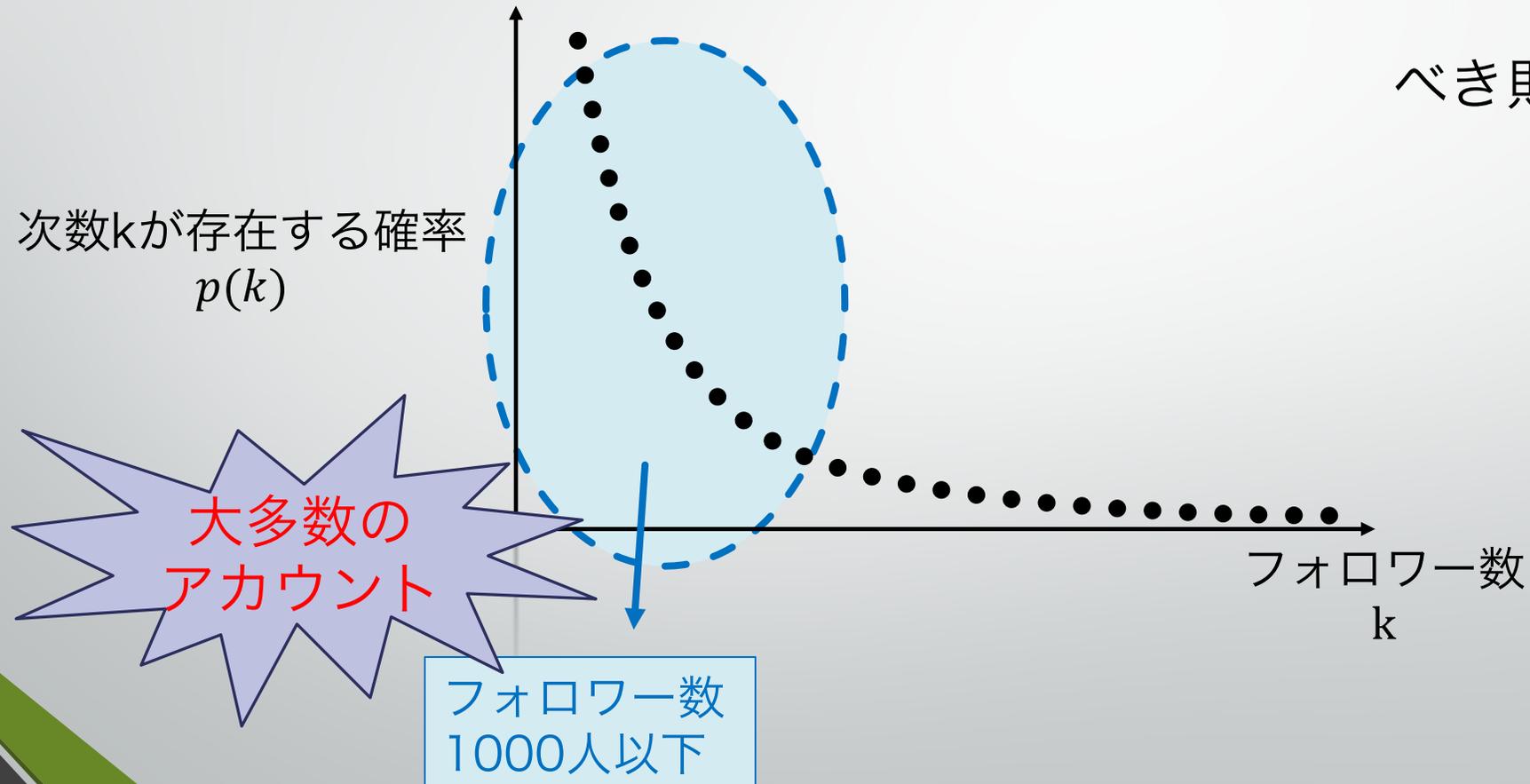


次数が大きくなるにつれて  
存在確率が小さくなっていく

### 3.修正Configurationモデル

## 3.1 スケールフリー性

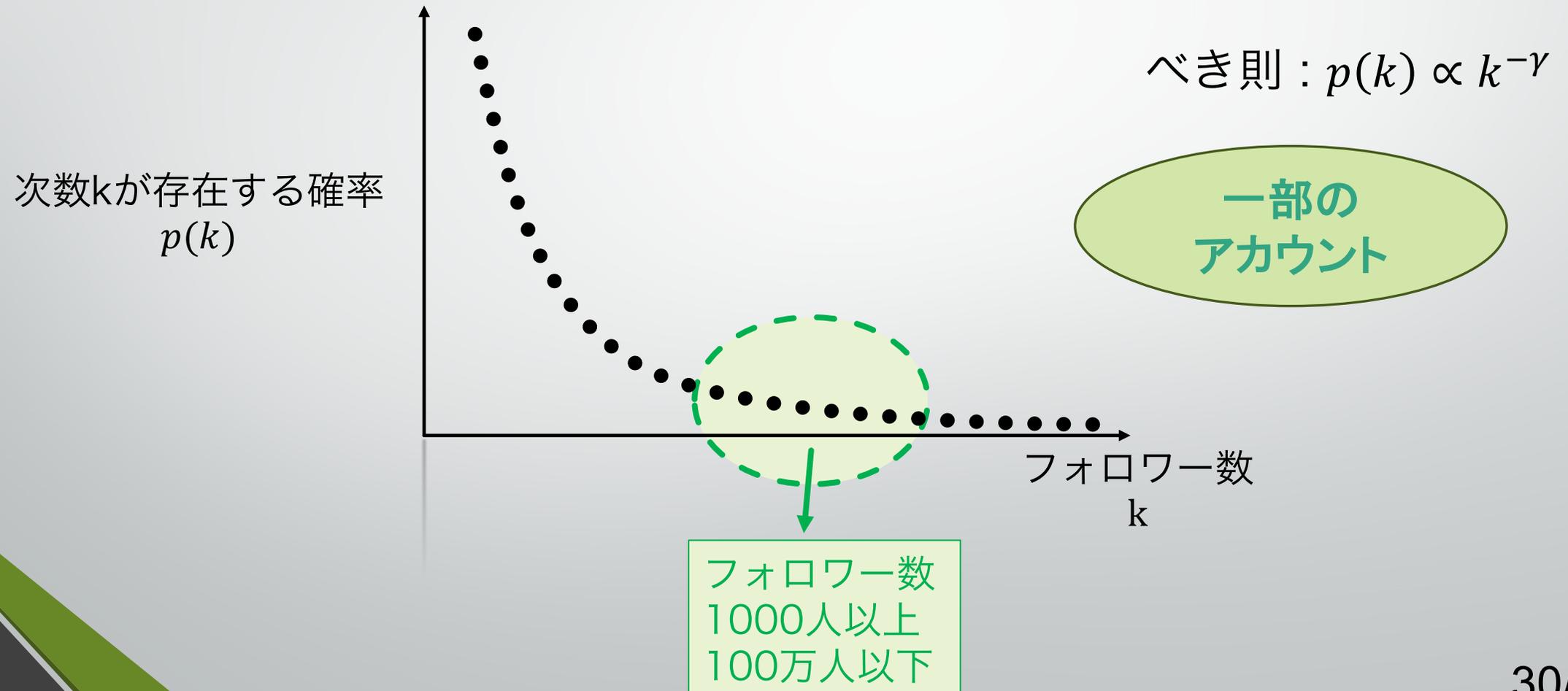
(例). Twitterのフォロワー数とべき則



### 3.修正Configurationモデル

## 3.1 スケールフリー性

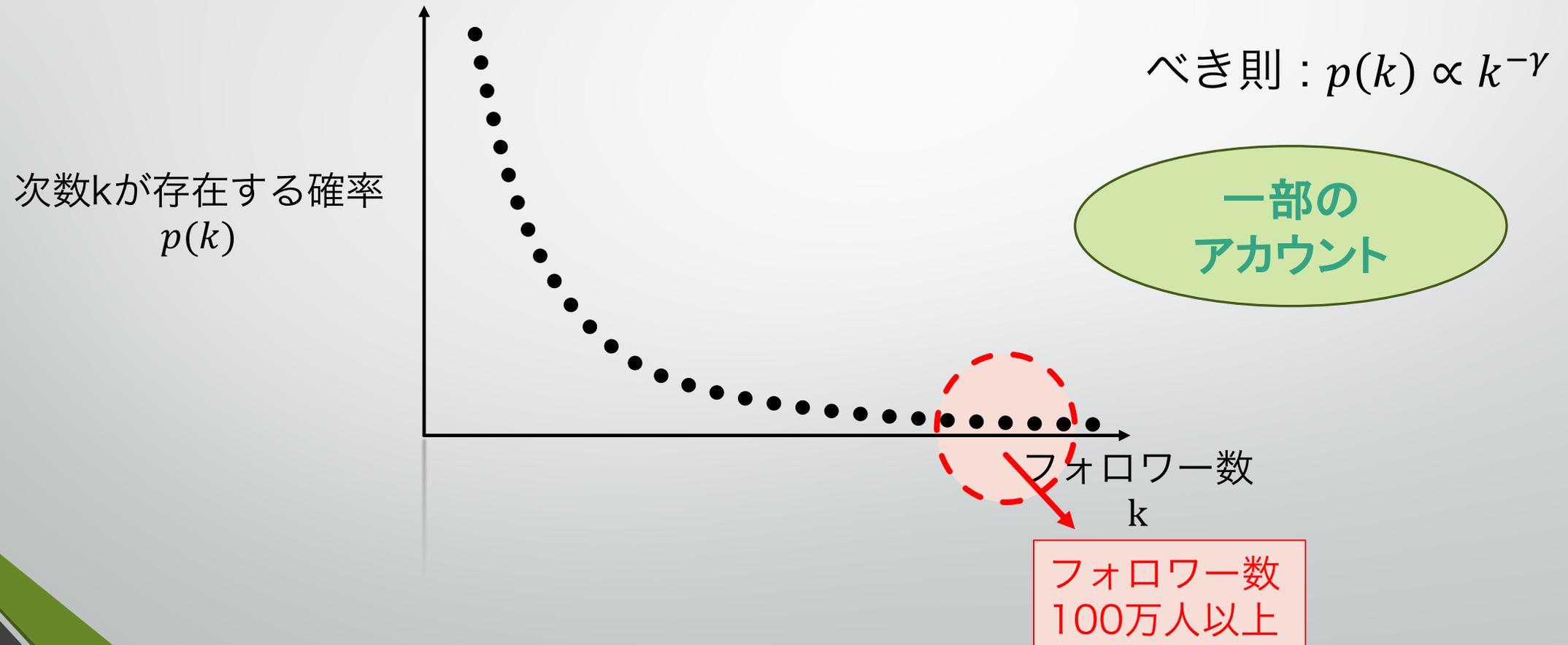
(例). Twitterのフォロワー数とべき則



### 3.修正Configurationモデル

## 3.1 スケールフリー性

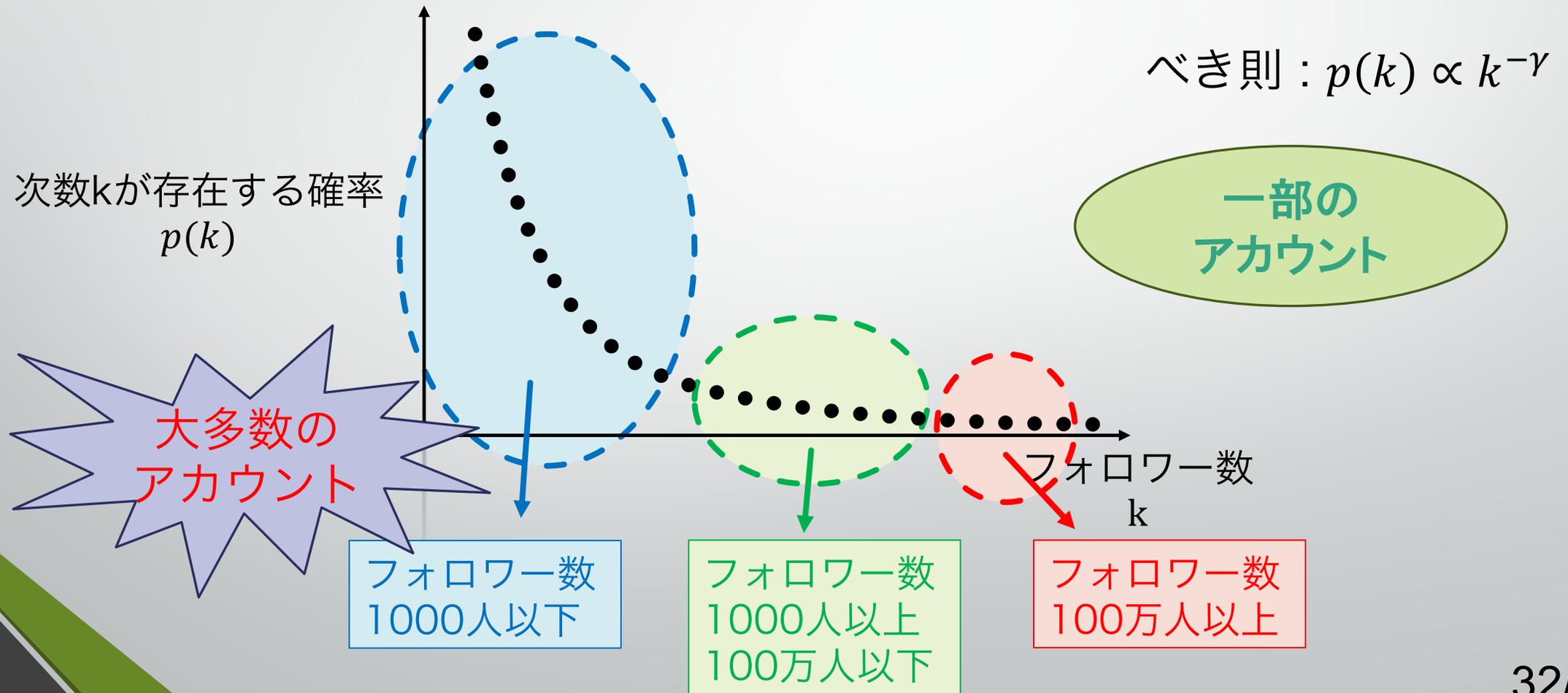
(例). Twitterのフォロワー数とべき則



# 3.修正Configurationモデル

## 3.1 スケールフリー性

(例). Twitterのフォロワー数とべき則



## 3.修正Configurationモデル

### 3.1 スケールフリー性

- スケールフリーネットワーク  
ネットワークの頂点次数の分布がべき則に従うとき,そのネットワークを  
スケールフリーネットワークという
- スケールフリーネットワークでは次数の小さい頂点が多く見られ,  
次数の大きい頂点は少ない

# 3.修正Configurationモデル

## 3.2 Configurationモデル

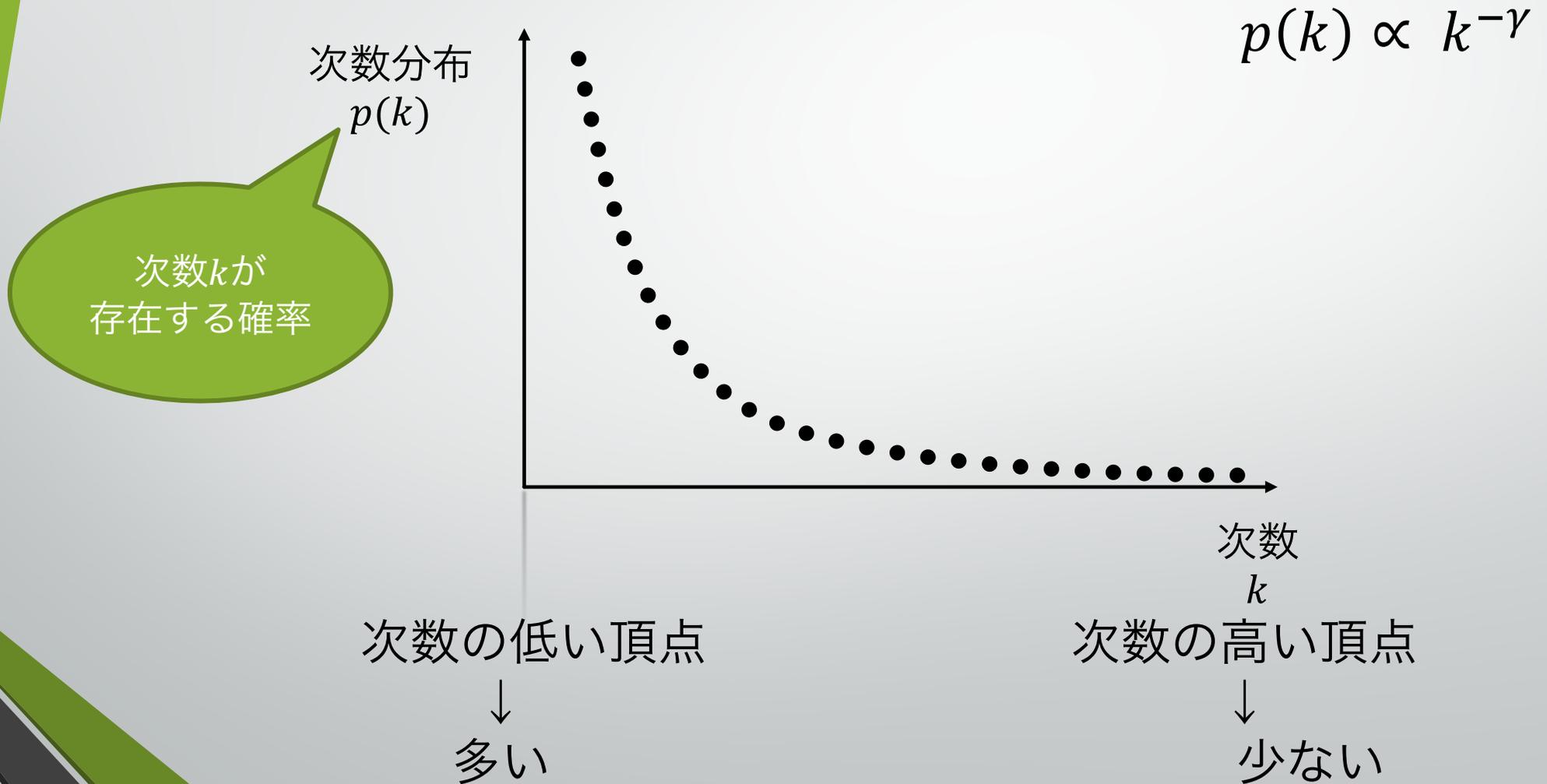
- Configurationモデル

べき指数を指定することで任意のべき則でスケールフリーネットワークを生成できるモデル

「configuration」 : 形状, 地形, 輪郭, 外形

### 3.修正Configurationモデル

## 3.2 Configurationモデル



### 3.修正Configurationモデル

## 3.2 Configurationモデル

- Configurationモデルのアルゴリズム
  1. べき指数 $\gamma$ と頂点数 $N$ を決定する
  2. べき則に従った次数分布で各頂点の次数 $k$ を決める
  3.  $N$ 個の頂点を配置し,各頂点の次数分の「枝の片割れ」を持たせる
  4. 「枝の片割れ」を2つ乱択し,それらを繋いで一つの枝とする
  5. 「枝の片割れ」が全て無くなるまで 4 を繰り返す

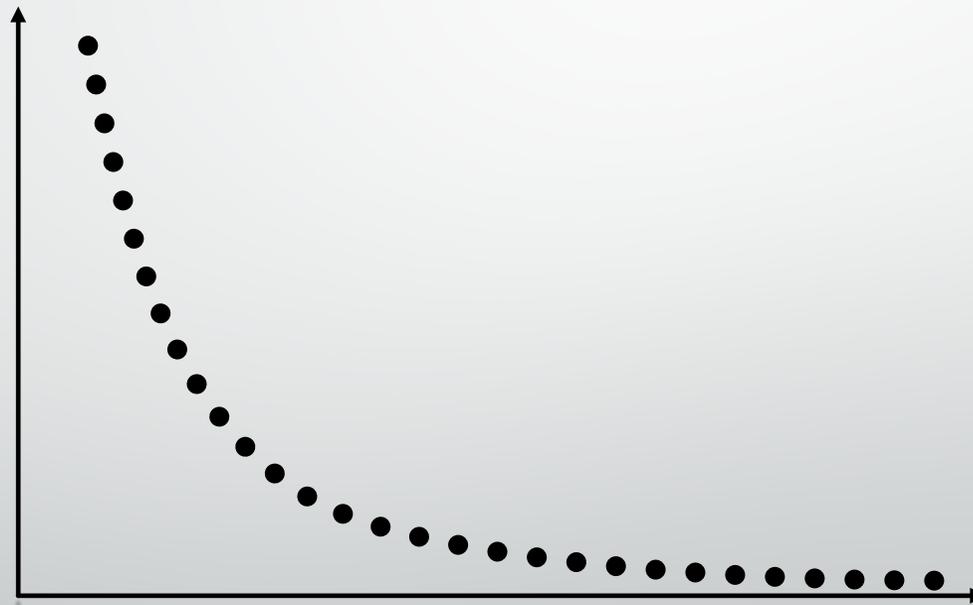
### 3.修正Configurationモデル

## 3.2 Configurationモデル

1.べき指数, 頂点数, 最小次数を決める

ex.  $\gamma = 2.0$ ,  $N = 10000$ ,  $k_{min} = 2$

次数分布  
 $p(k)$



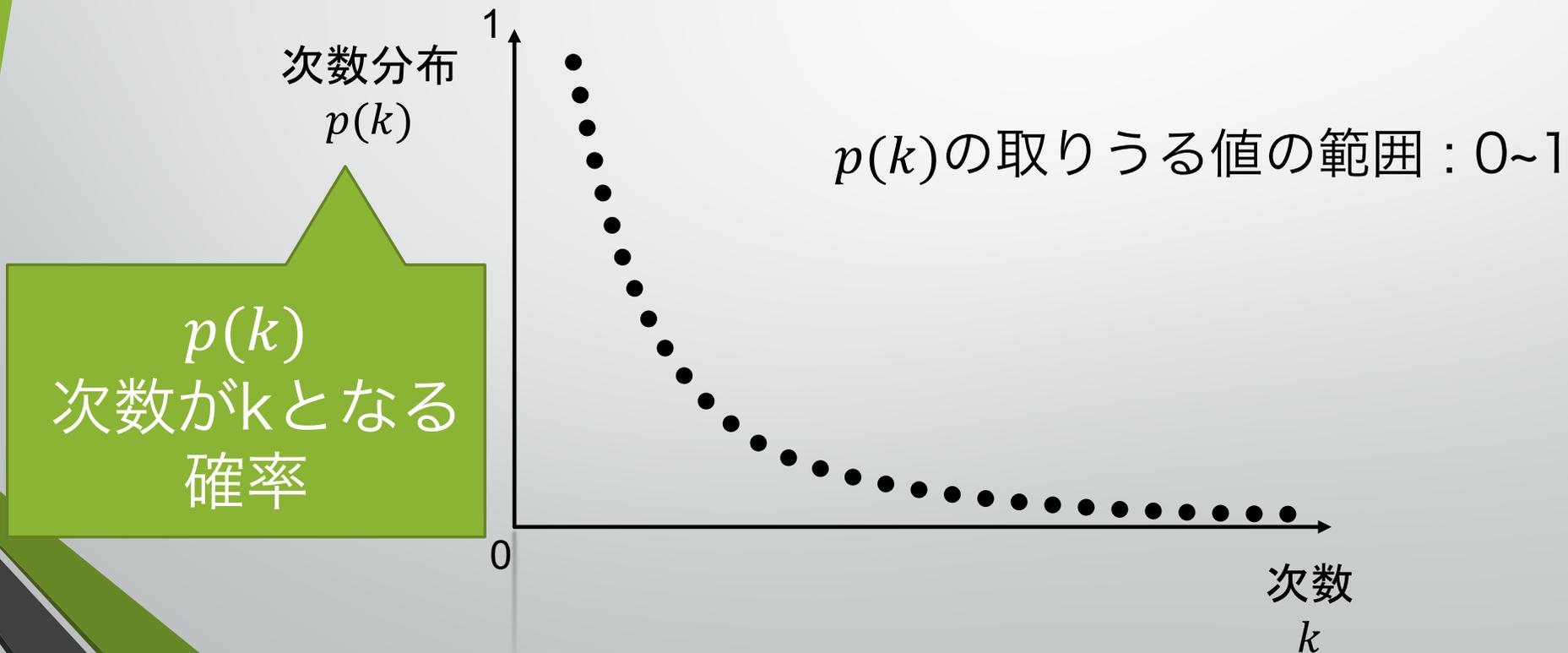
べき則  
 $p(k) \propto k^{-\gamma}$

次数  
 $k$

### 3.修正Configurationモデル

## 3.2 Configurationモデル

2.べき則に従った確率分布で各頂点の次数を決定する



### 3.修正Configurationモデル

## 3.2 Configurationモデル

2.べき則に従った確率分布で各頂点の次数を決定する

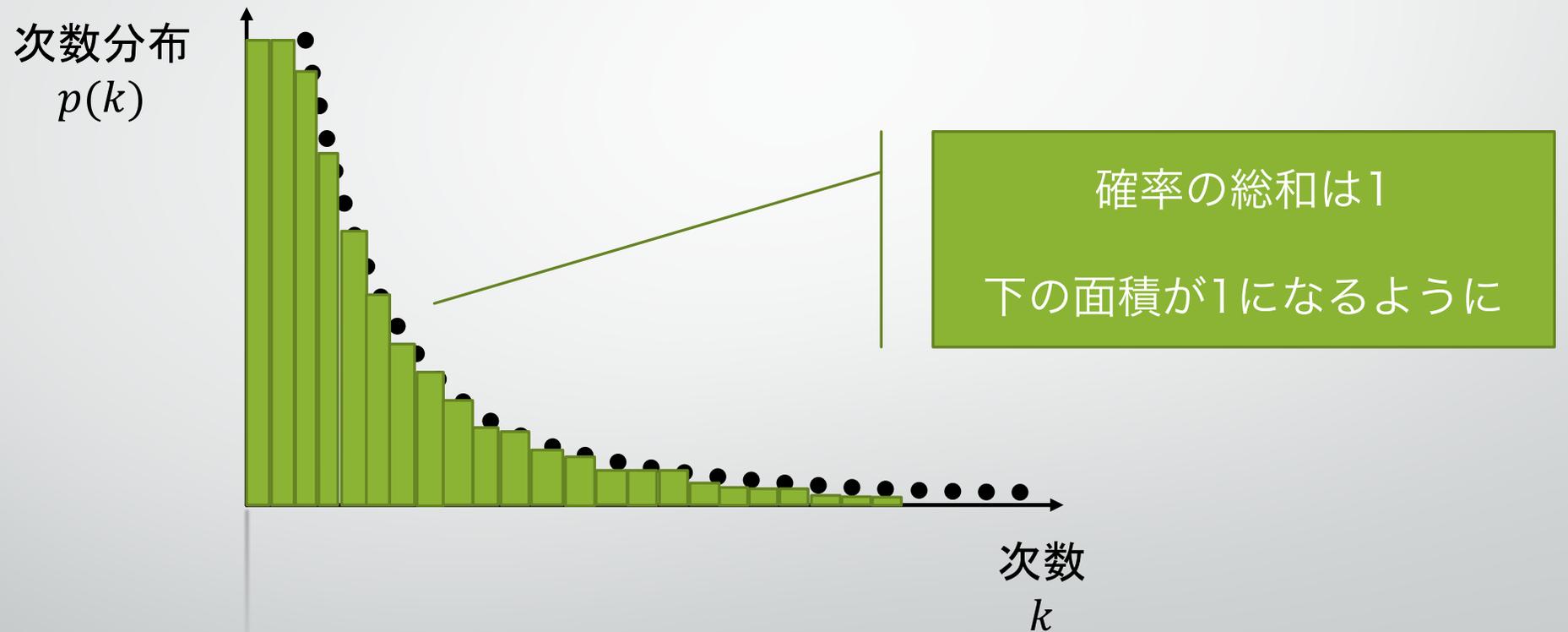
確率の性質より

$$\sum_{i=0}^{\infty} p(i) = 1$$

### 3.修正Configurationモデル

## 3.2 Configurationモデル

2.べき則に従った確率分布で各頂点の次数を決定する



すべての $p(k)$ の合計が1になるように規格化

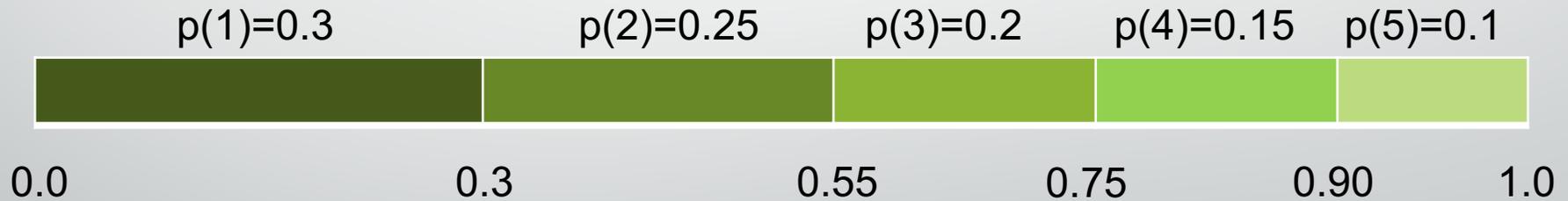
### 3.修正Configurationモデル

## 3.2 Configurationモデル

2.べき則に従った確率分布で各頂点の次数を決定する

規格化した $p(k)$ から累積確率分布を求める

ex.  $p(1)=0.3$ ,  $p(2)=0.25$ ,  $p(3)=0.2$ ,  $p(4)=0.15$ ,  $p(5)=0.1$  とすると

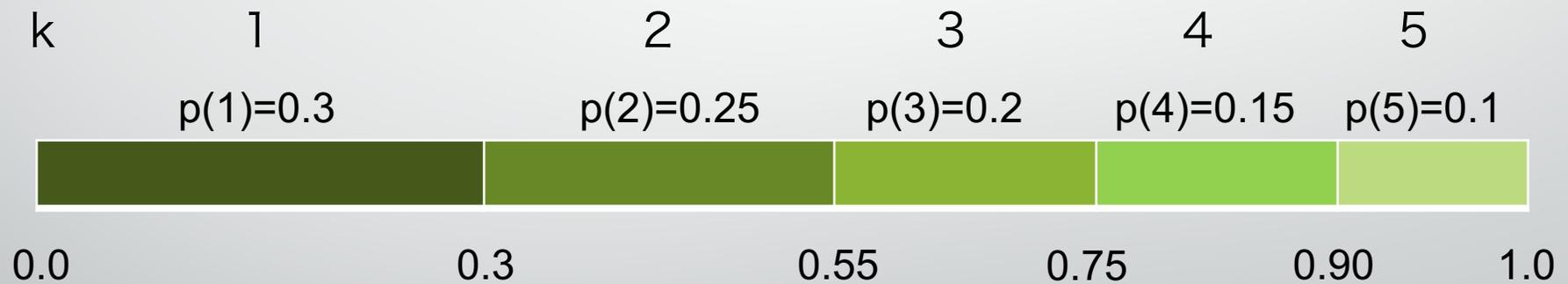


### 3.修正Configurationモデル

## 3.2 Configurationモデル

2.べき則に従った確率分布で各頂点の次数を決定する

0~1の範囲で乱数を生成し,その値を含んだ累積確率が示す次数を頂点の次数とする

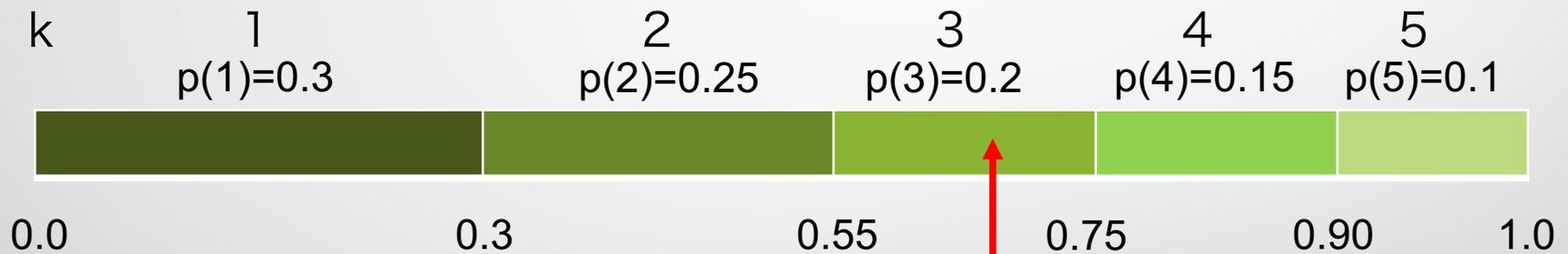


### 3.修正Configurationモデル

## 3.2 Configurationモデル

2.べき則に従った確率分布で各頂点の次数を決定する

ex. 乱数  $r = 0.64$



0.64

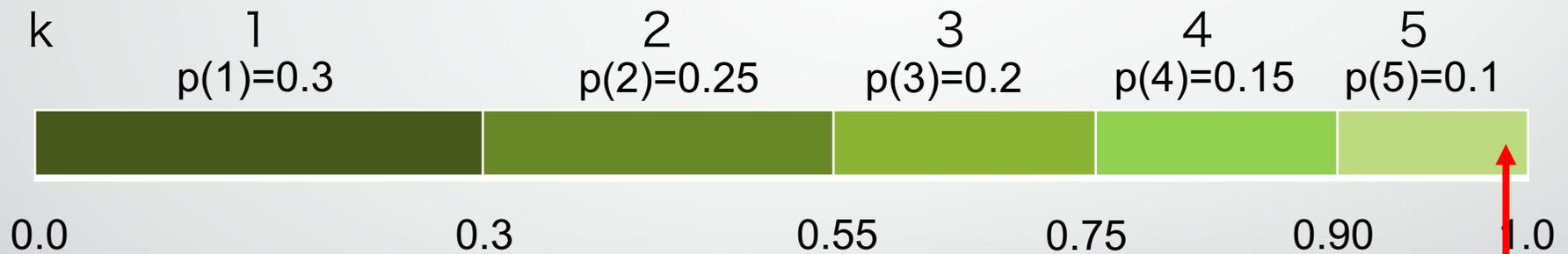
頂点の次数を**3**に決定

### 3.修正Configurationモデル

## 3.2 Configurationモデル

2.べき則に従った確率分布で各頂点の次数を決定する

ex. 乱数  $r = 0.99$



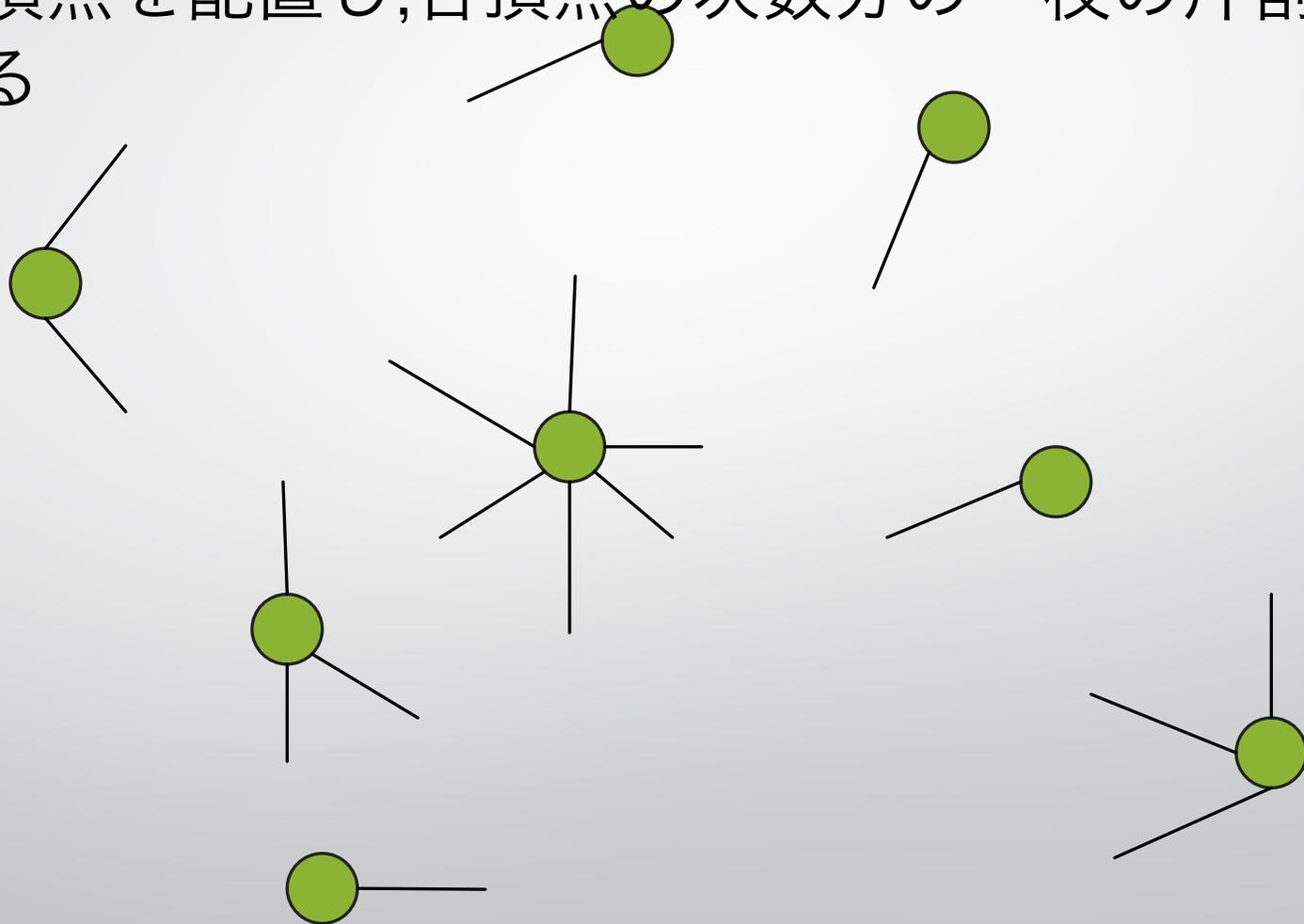
0.99

頂点の次数を**5**に決定

### 3.修正Configurationモデル

## 3.2 Configurationモデル

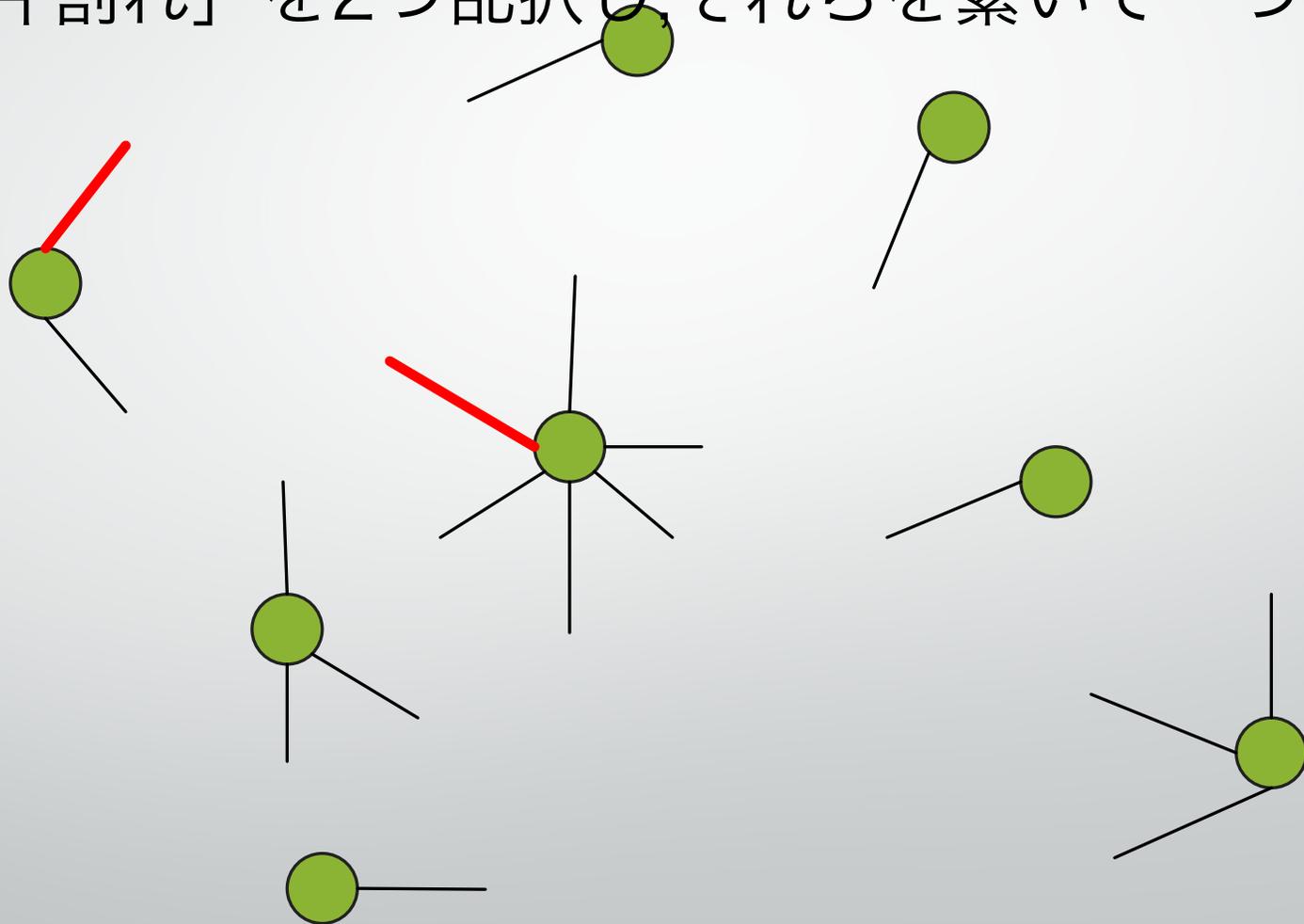
3.  $N$ 個の頂点を配置し,各頂点の次数分の「枝の片割れ」を持たせる



### 3.修正Configurationモデル

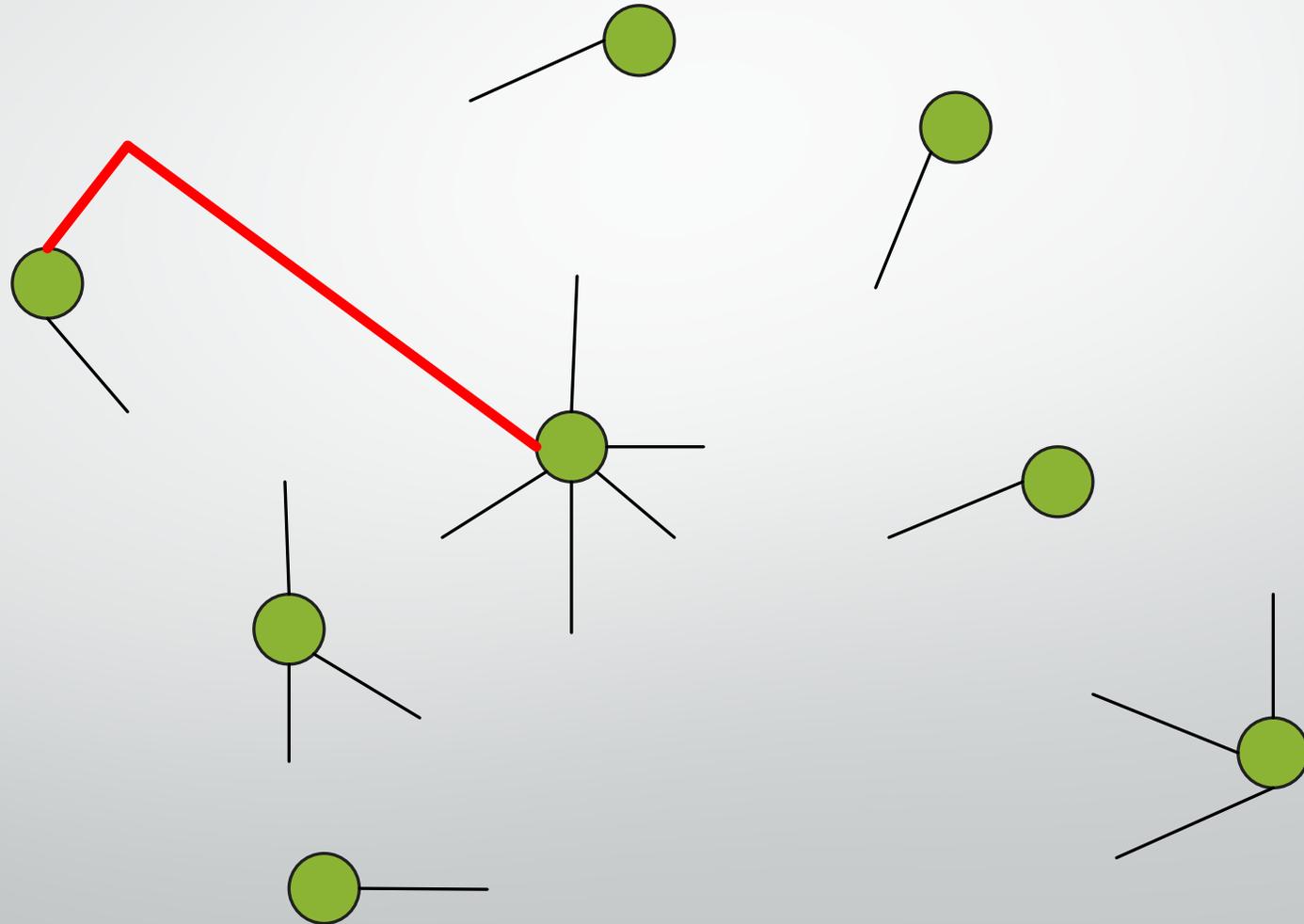
## 3.2 Configurationモデル

4. 「枝の片割れ」を2つ乱択し、それらを繋いで一つの枝とする



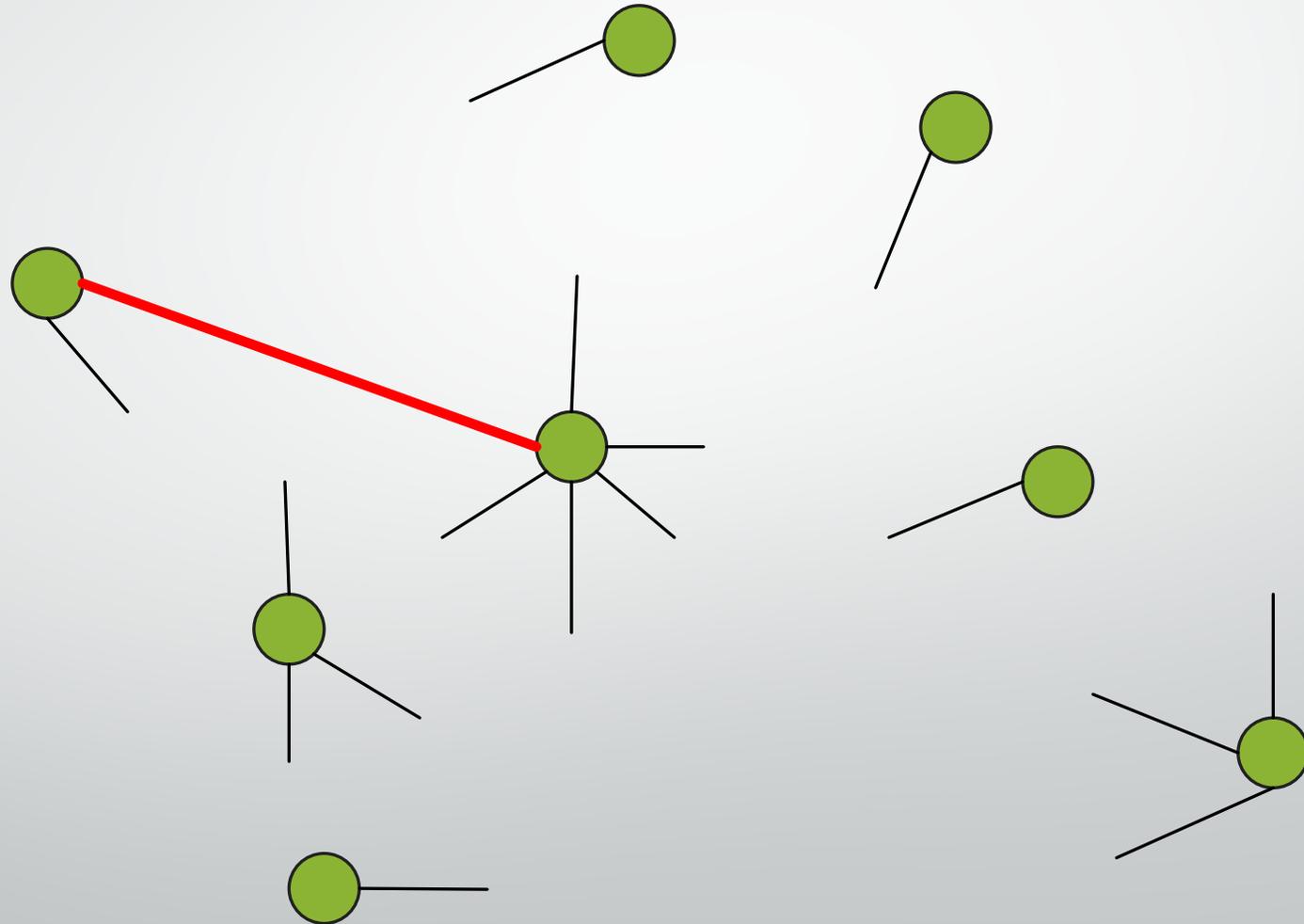
### 3.修正Configurationモデル

## 3.2 Configurationモデル



### 3.修正Configurationモデル

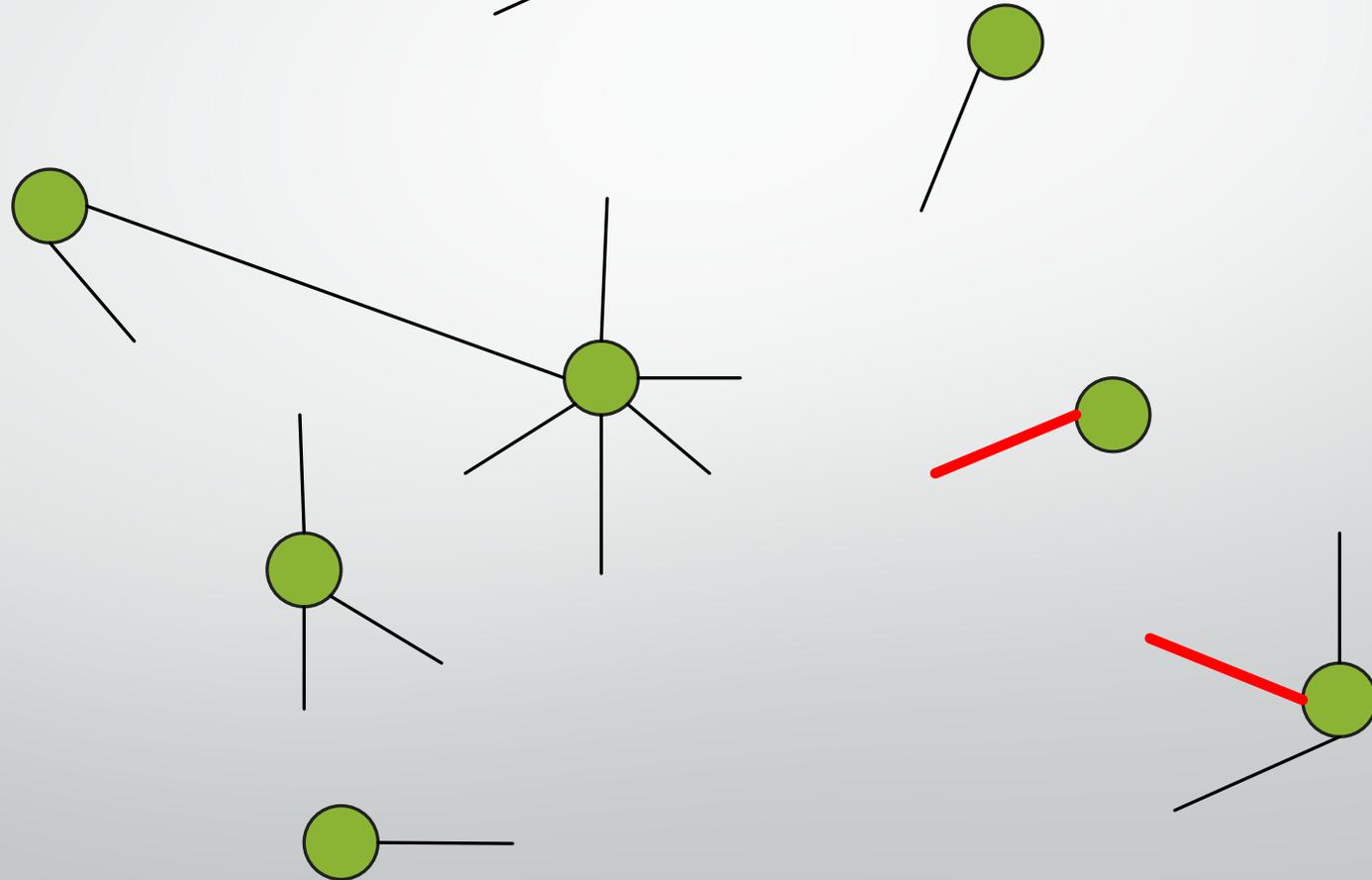
## 3.2 Configurationモデル



### 3.修正Configurationモデル

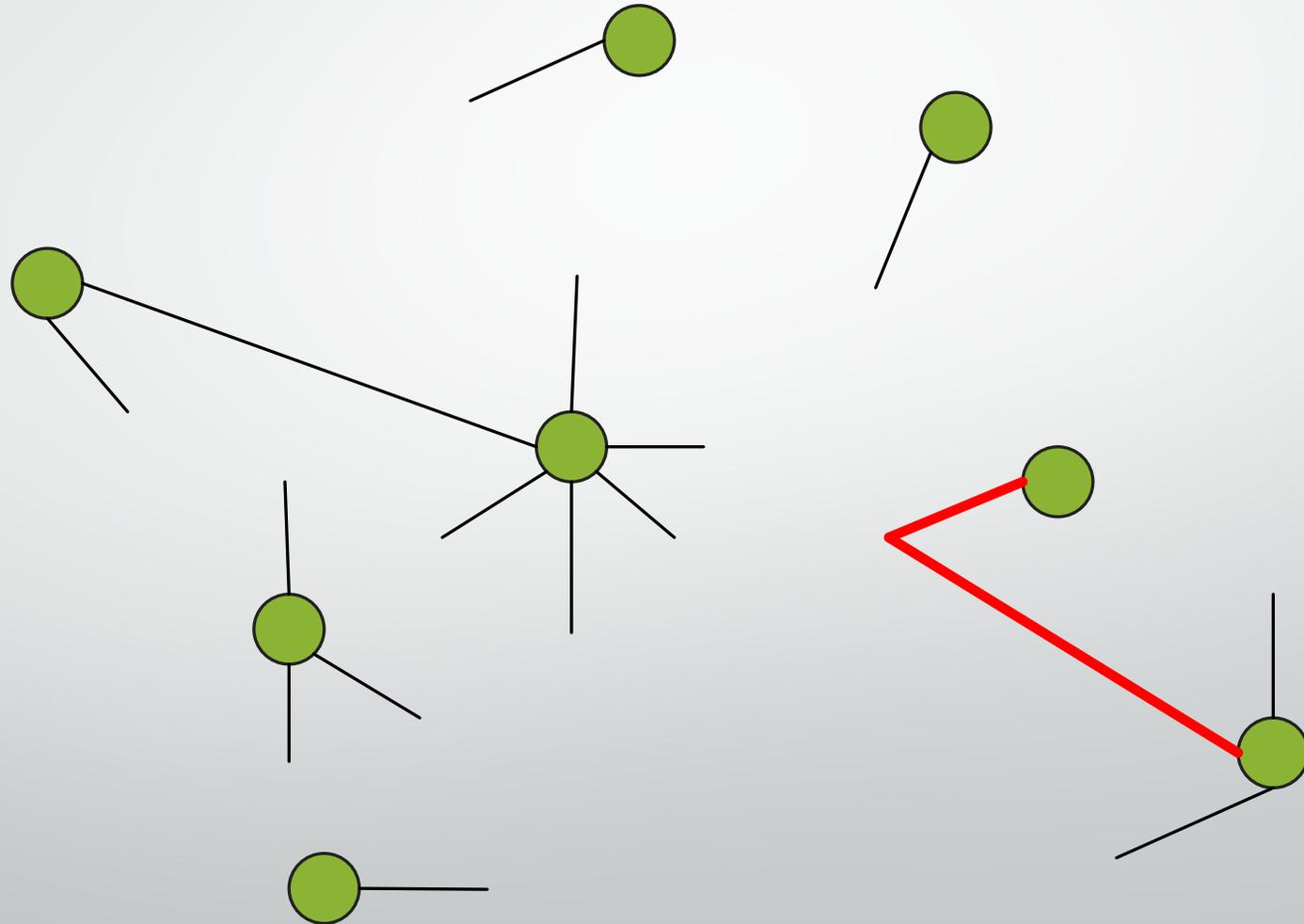
## 3.2 Configurationモデル

5. 「枝の片割れ」が全て無くなるまで 4 を繰り返す



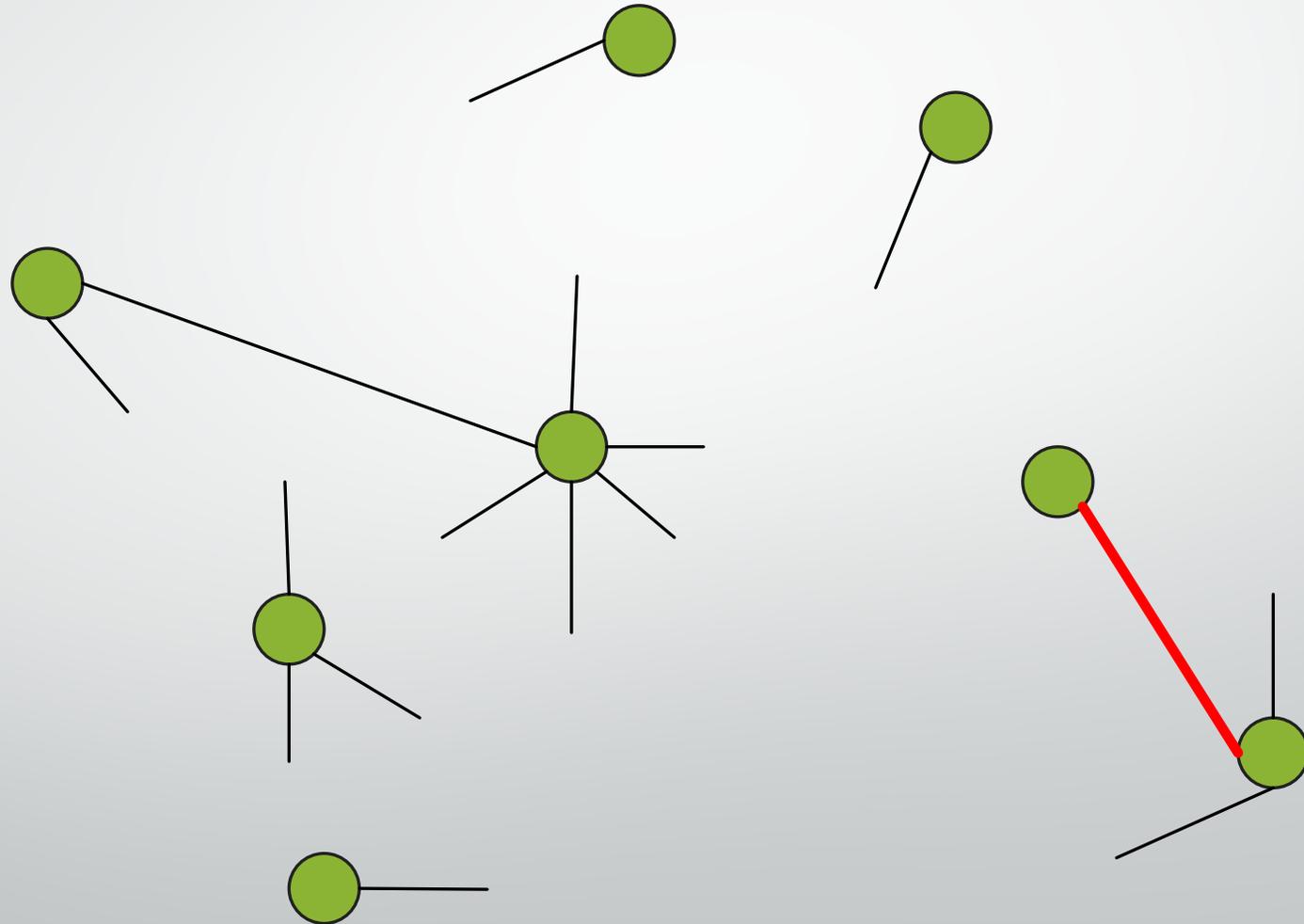
### 3.修正Configurationモデル

## 3.2 Configurationモデル



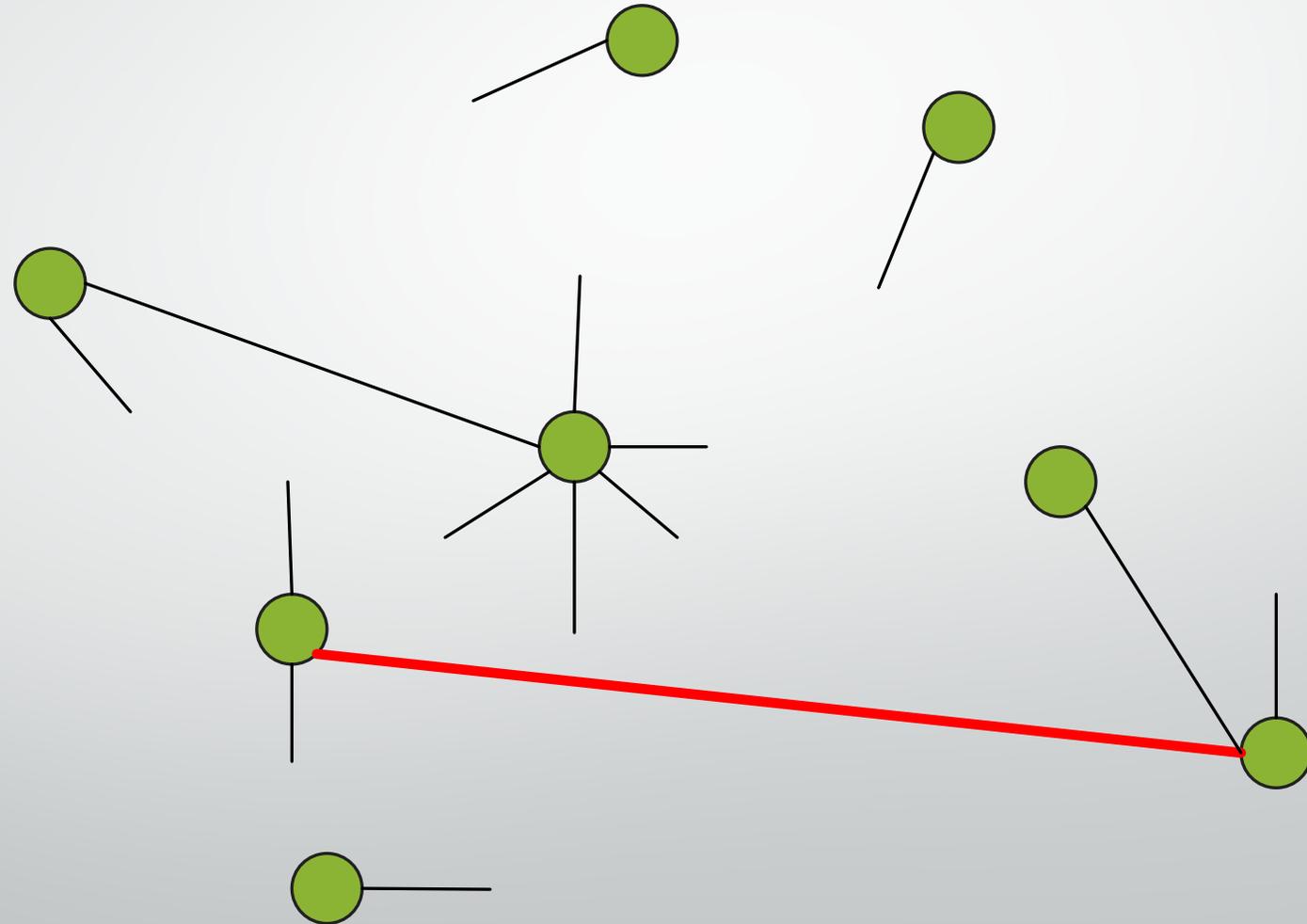
### 3.修正Configurationモデル

## 3.2 Configurationモデル



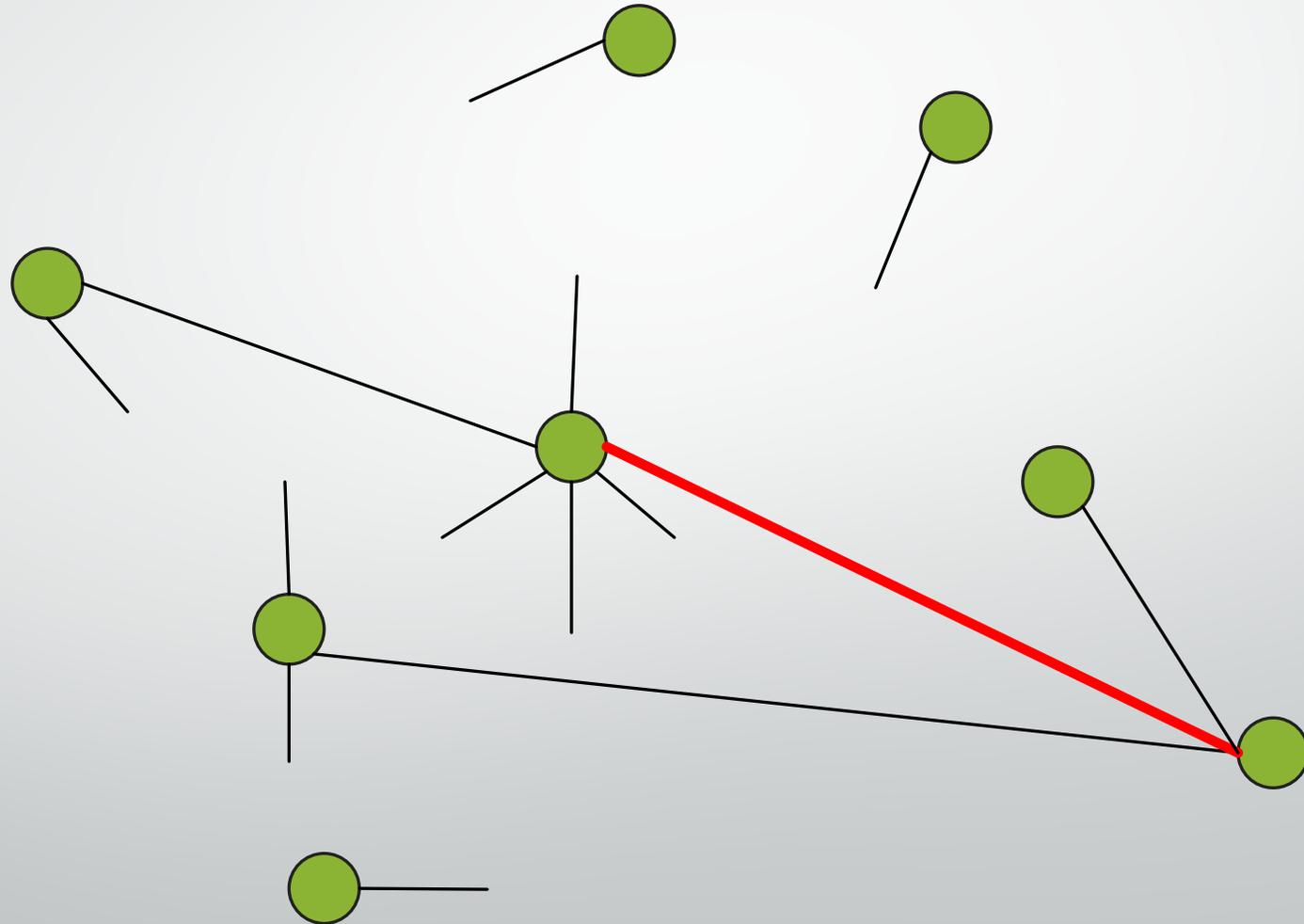
# 3.修正Configurationモデル

## 3.2 Configurationモデル



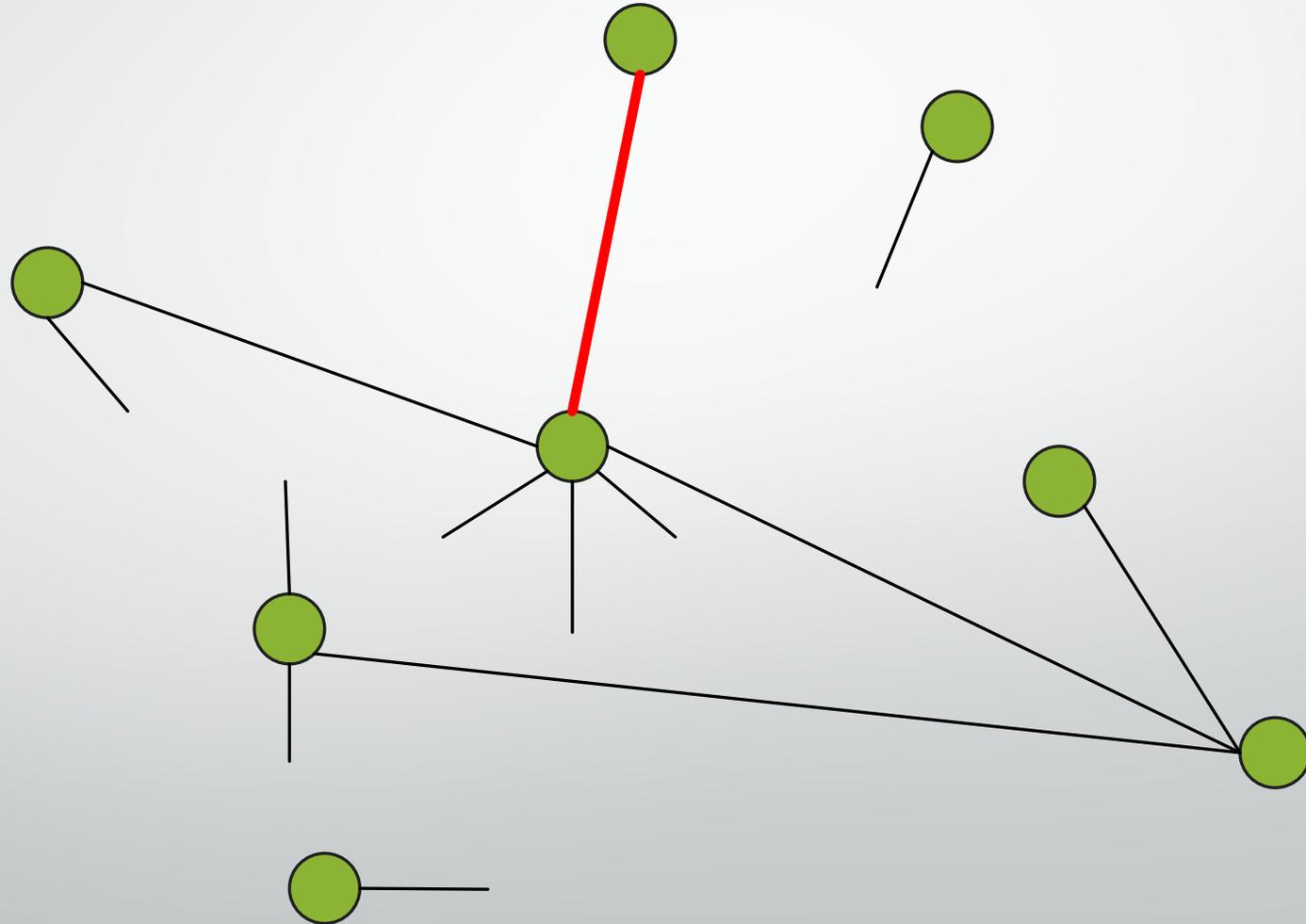
### 3.修正Configurationモデル

## 3.2 Configurationモデル



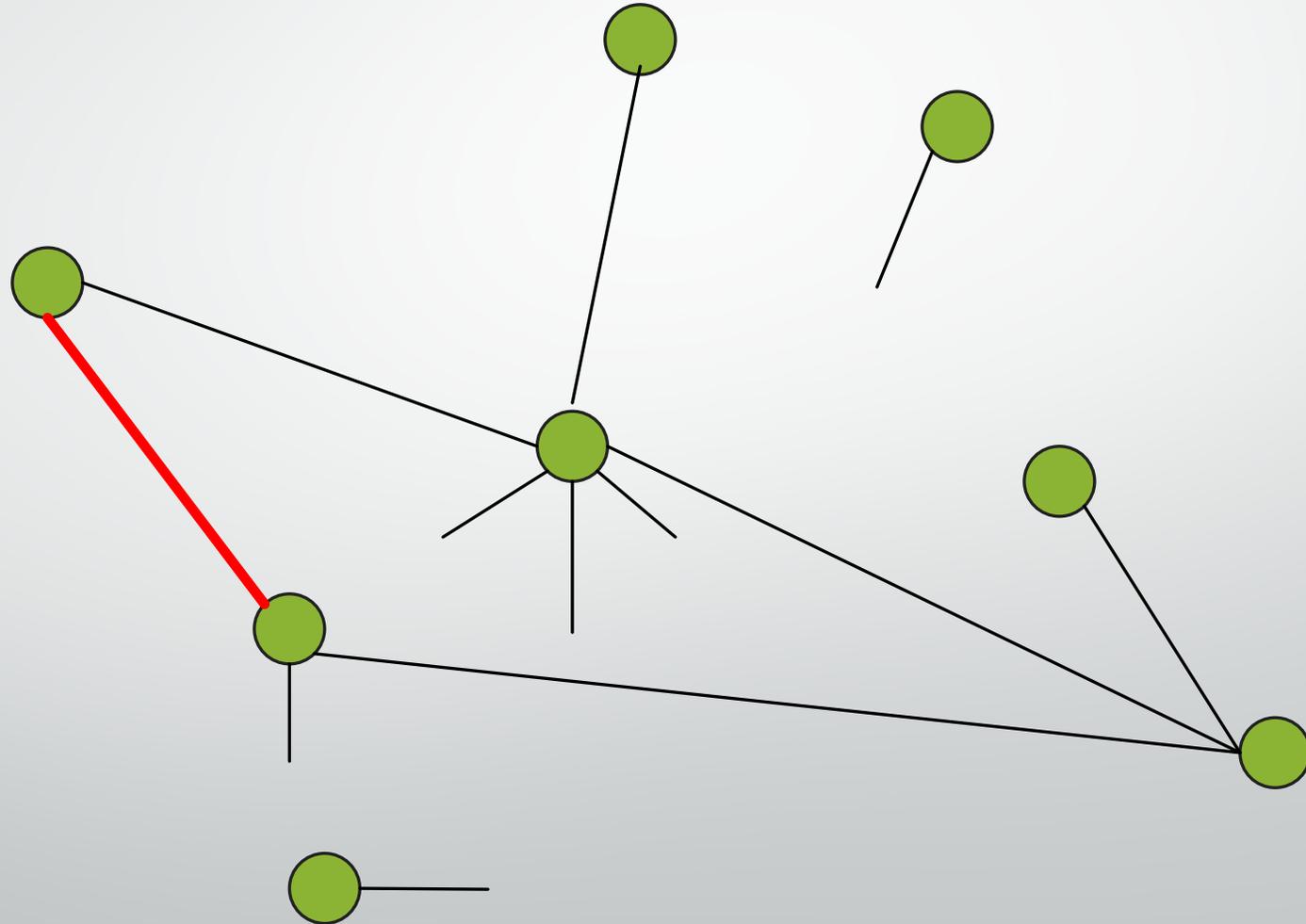
### 3.修正Configurationモデル

## 3.2 Configurationモデル



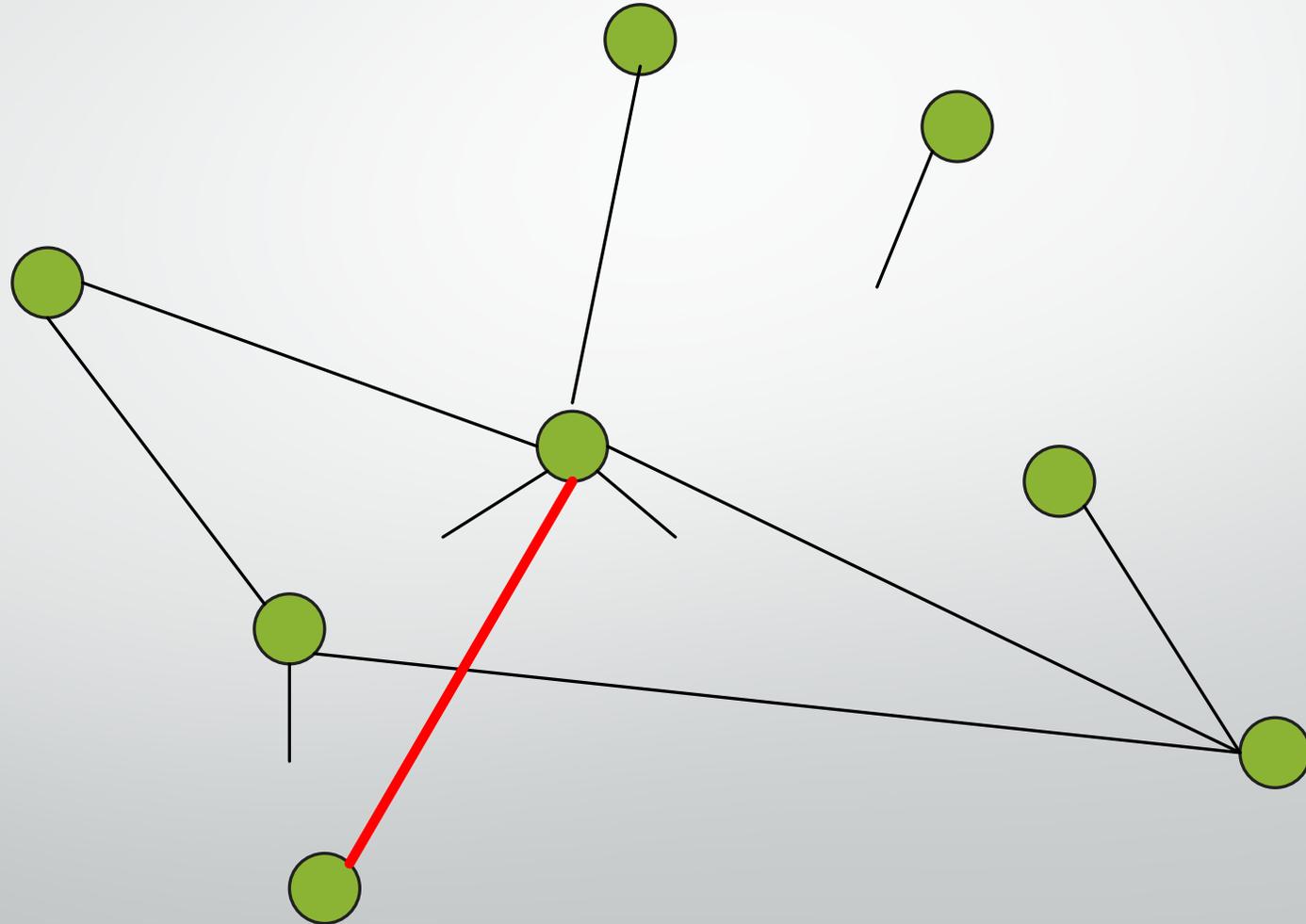
### 3.修正Configurationモデル

## 3.2 Configurationモデル



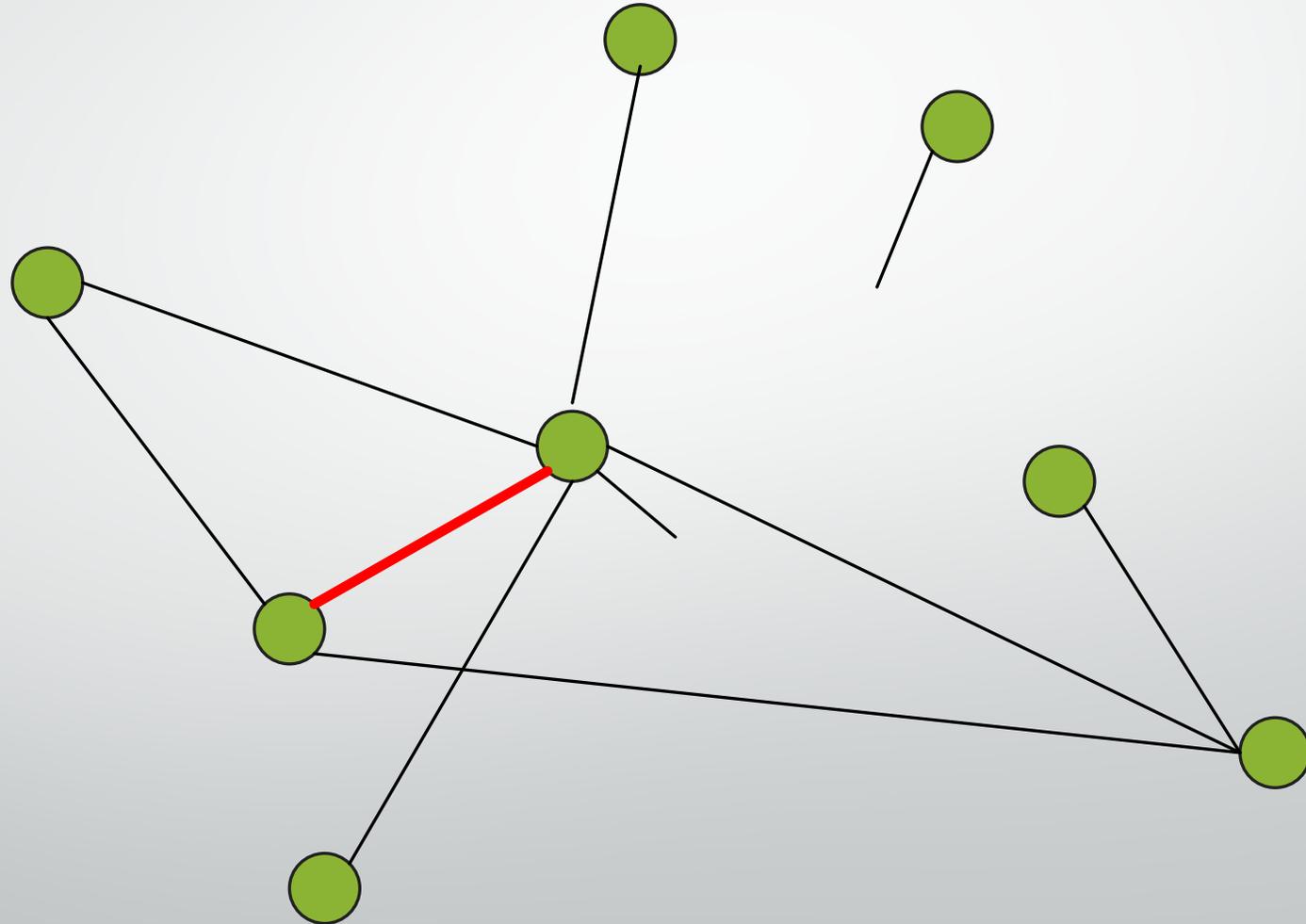
### 3.修正Configurationモデル

## 3.2 Configurationモデル



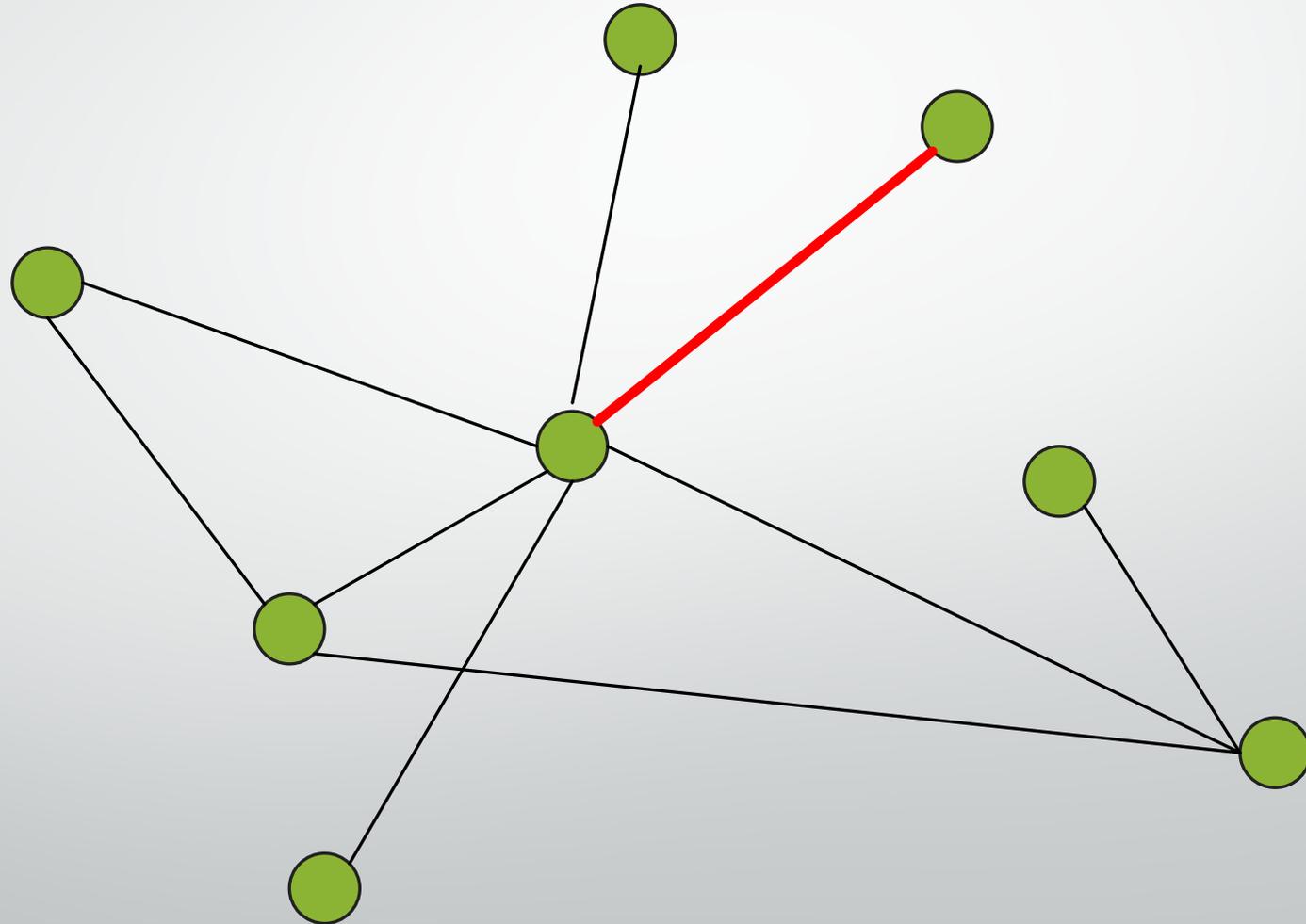
### 3.修正Configurationモデル

## 3.2 Configurationモデル



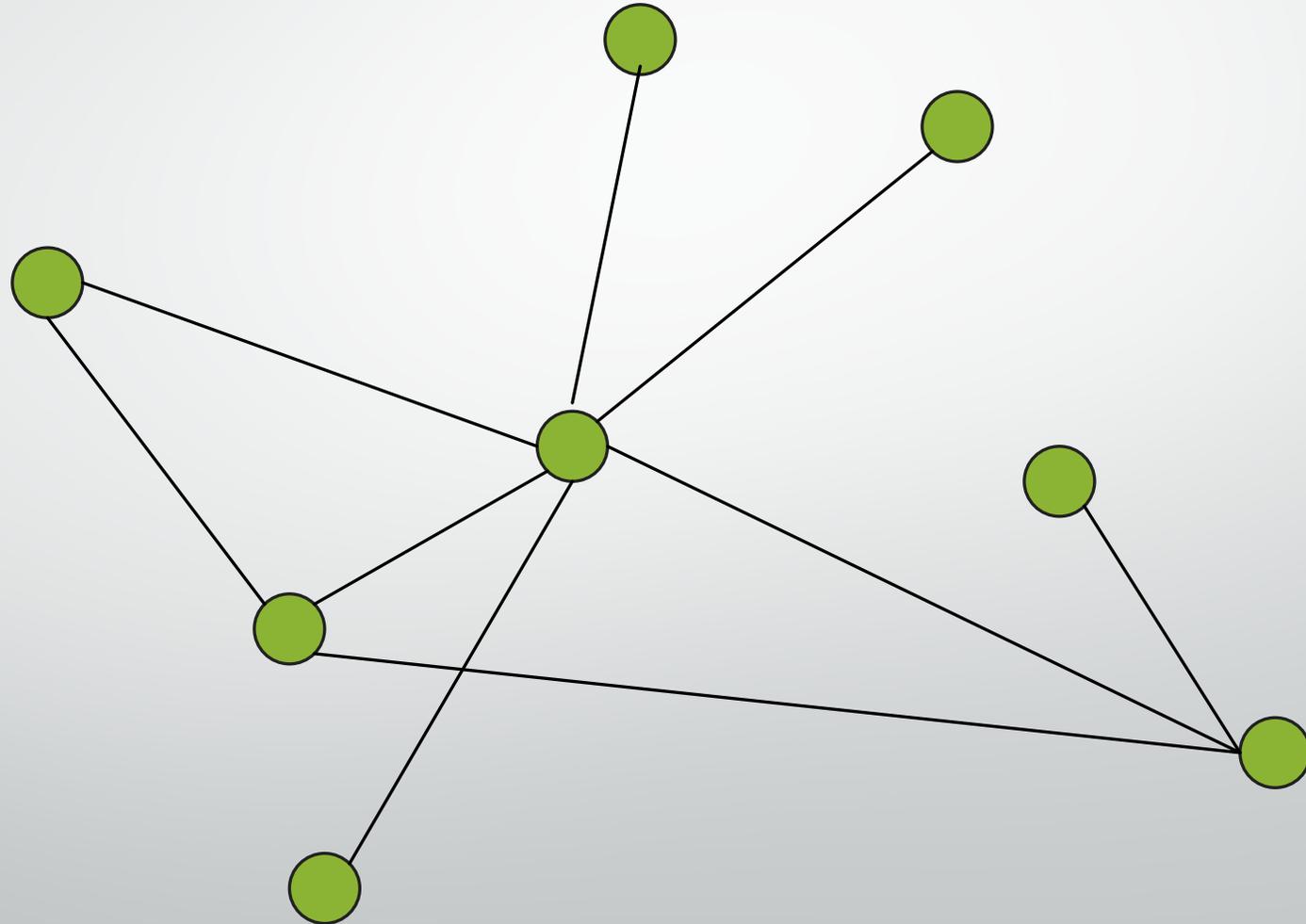
### 3.修正Configurationモデル

## 3.2 Configurationモデル



### 3.修正Configurationモデル

## 3.2 Configurationモデル



## 3.修正Configurationモデル

# 3.2 Configurationモデル

### 特徴

- 次数相関がほとんどない
- 自己ループや多重辺が発生する可能性
- 非連結となる可能性

# 3.修正Configurationモデル

## 3.2 Configurationモデル

- 自己ループや多重辺が発生する可能性
  - ✓ 自己ループとなる原因
    - 同一頂点の枝の片割れが選ばれた時
  - ✓ 多重辺となる原因
    - 辺が張られている頂点のペアが再び選ばれた時

### 3.修正Configurationモデル

## 3.2 Configurationモデル

- 自己ループが発生する場合

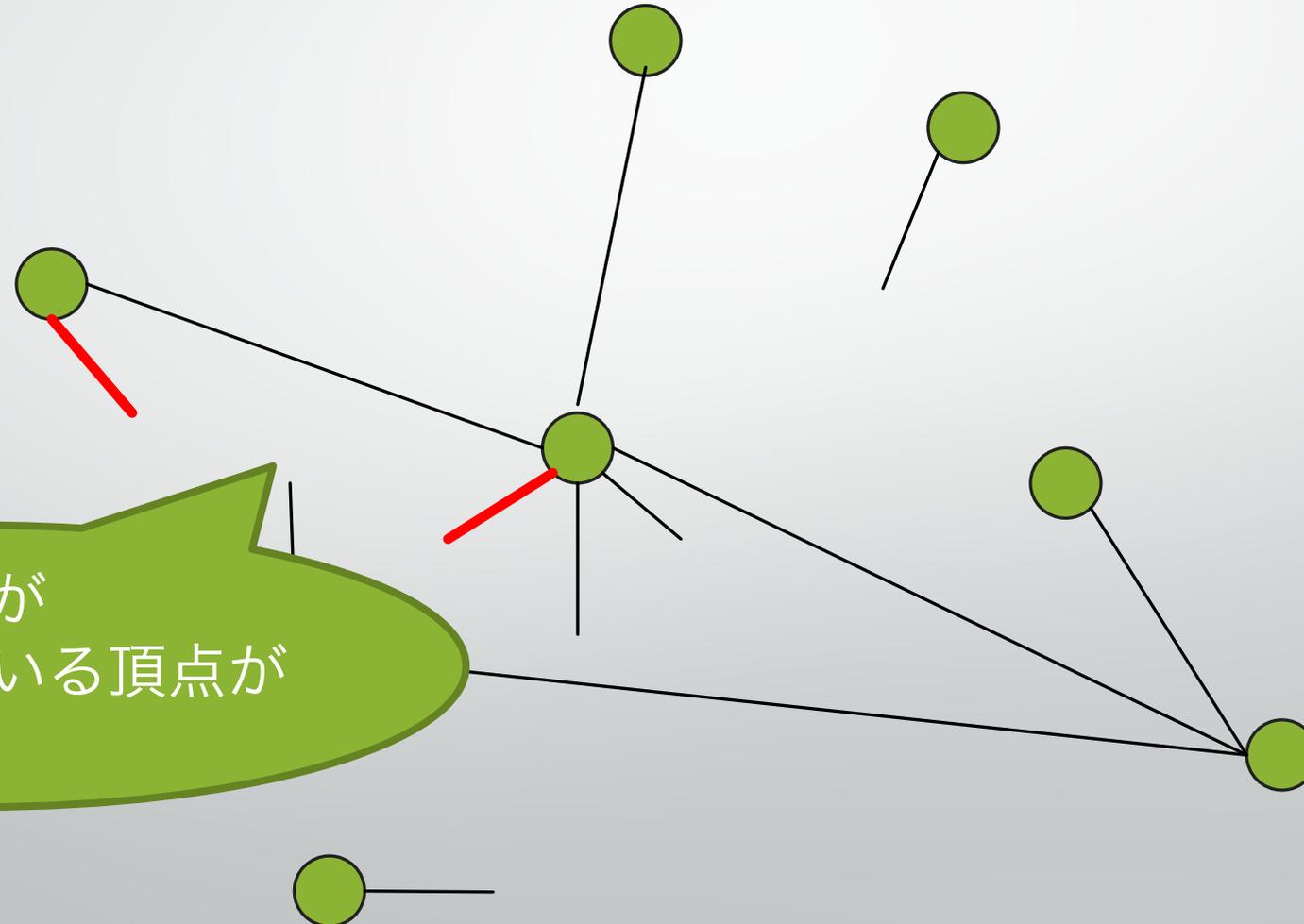




### 3.修正Configurationモデル

## 3.2 Configurationモデル

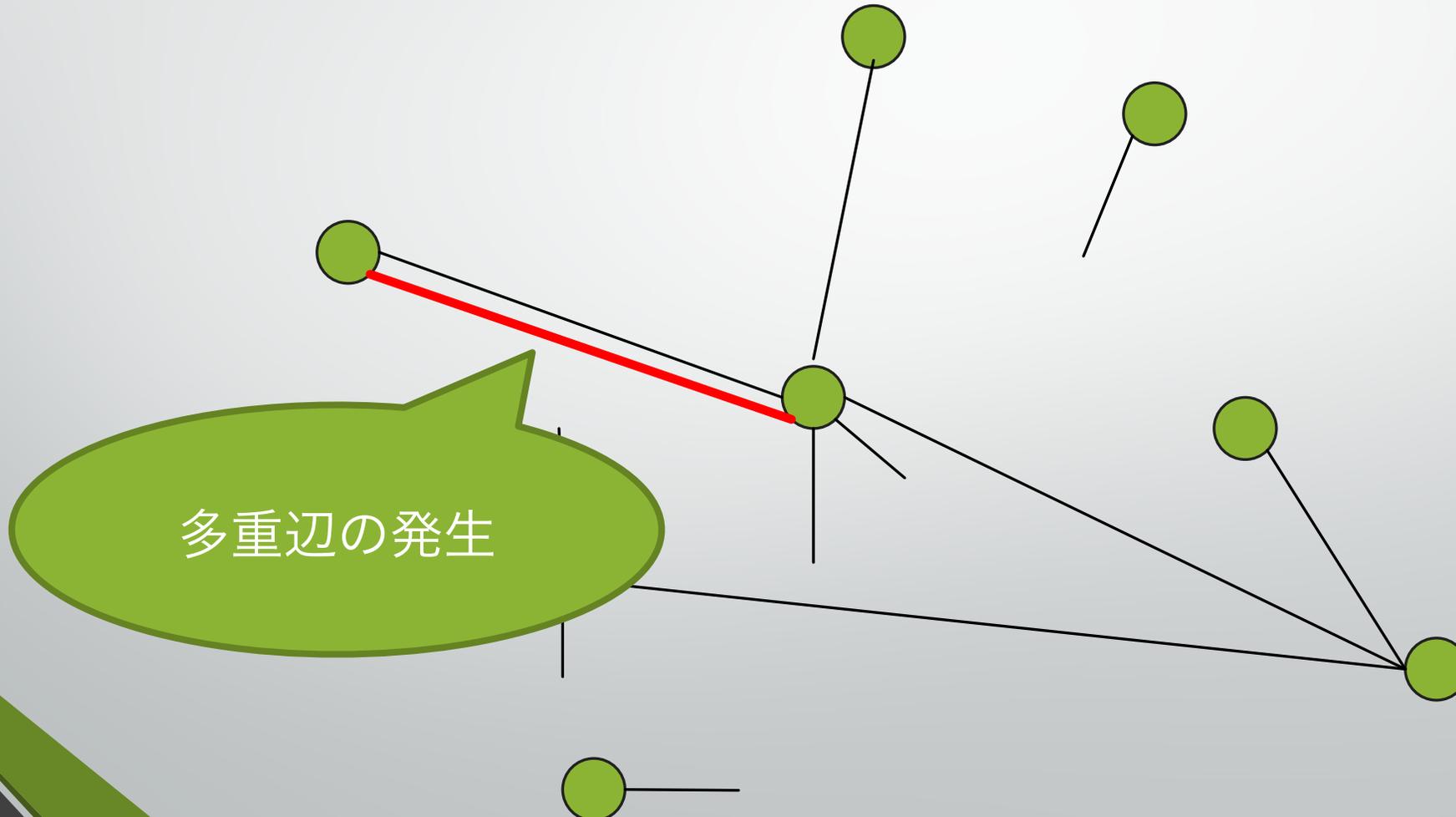
- 多重辺が発生する場合



### 3.修正Configurationモデル

## 3.2 Configurationモデル

- 多重辺が発生する場合



### 3.修正Configurationモデル

## 3.2 Configurationモデル

自己ループや多重辺,非連結となる可能性を排除すると生成したネットワークのUncorrelated性が損なわれる

## 3.修正Configurationモデル

### 3.3 Uncorrelated性

- Uncorrelated性

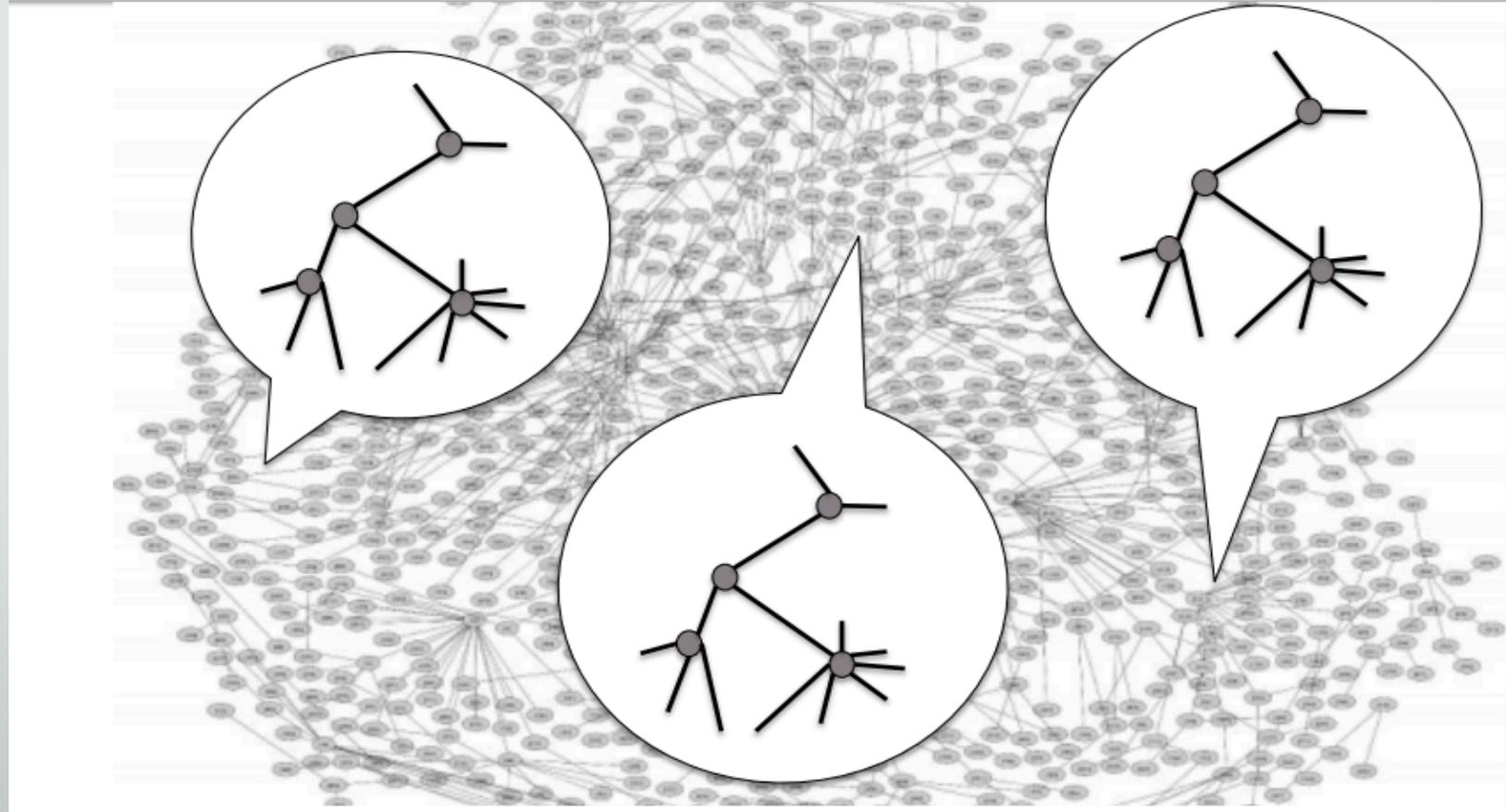
複雑ネットワークの研究で用いられるネットワークの性質

局所性がなく,ネットワーク全体で同じような構造が見られる

ある次数 $k$ をもつ頂点に隣接する頂点の次数分布はどれも似ている

# 3.修正Configurationモデル

## 3.3 Uncorrelated性



## 3.修正Configurationモデル

# 3.4 修正Configurationモデル

- 修正Configurationモデル

以下の論文で提案されたモデル

- ・ Michele Catanzaro, Marian Boguna, Romualdo Pastor-Satorras

「Generation of uncorrelated random scale-free networks」

<http://journals.aps.org/pre/abstract/10.1103/PhysRevE.71.027103>

- ・ 論文内での名称は, Uncorrelated Configuration Model (UCM)

- ・ 真のUncorrelatedネットワークを生成できるかは  
数学的に証明されていない(実験的に検証)

### 3.修正Configurationモデル

## 3.4 修正Configurationモデル

- 修正Configurationモデル

自己ループや多重辺を持たないConfigurationモデルを改良して、よりUncorrelated性を持つようにしたもの

頂点の最大次数を制限してUncorrelated性の高いネットワークを再現している

### 3.修正Configurationモデル

## 3.4 修正Configurationモデル

- 修正Configurationモデルでは,頂点の最大次数を制限する
  - Configurationモデル :  $k_{\min} < k < N-1$  ( $k_{\min}$  は最小次数)
  - 修正Configurationモデル :  $k_{\min} < k < N^{\frac{1}{2}}$

カットオフ

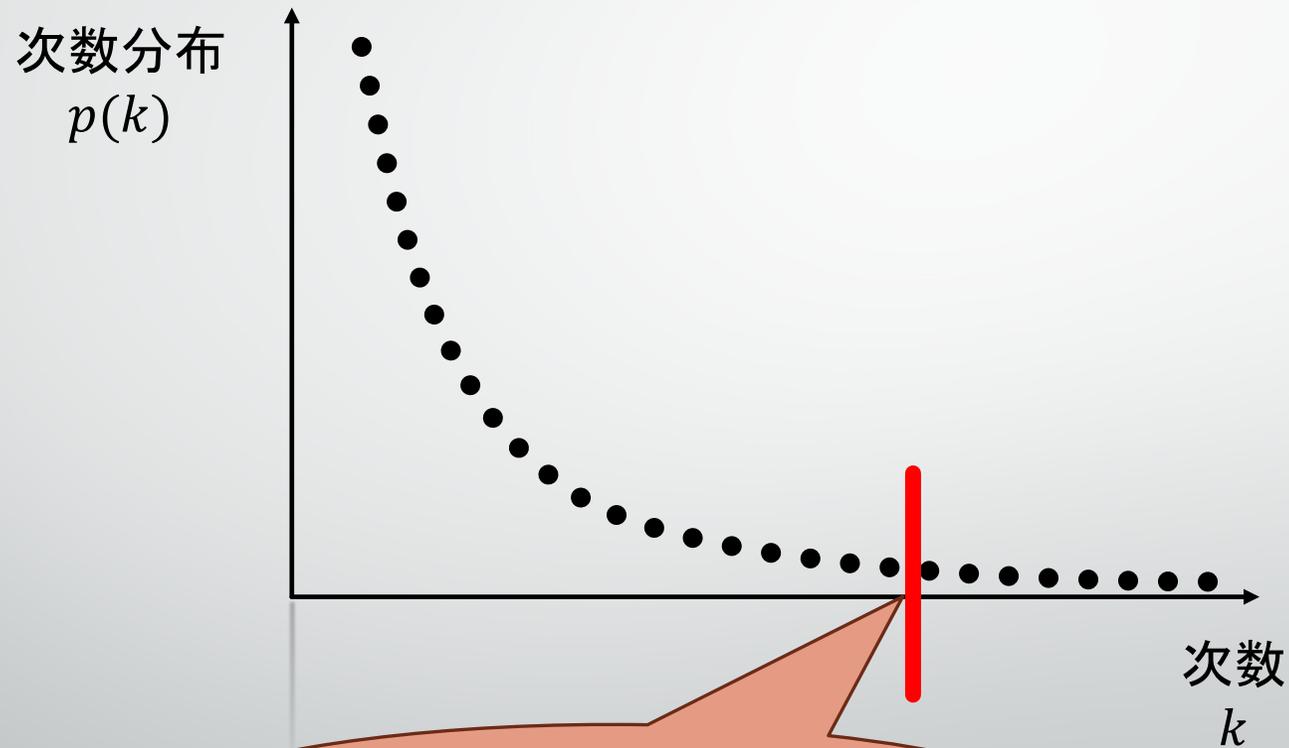
### 3.修正Configurationモデル

## 3.4 修正Configurationモデル

- 頂点数が10000 → 次数は最大で9999
- 現実ではそのようなことは起こりにくい
- 最大次数をある程度制限してネットワークを生成する

### 3.修正Configurationモデル

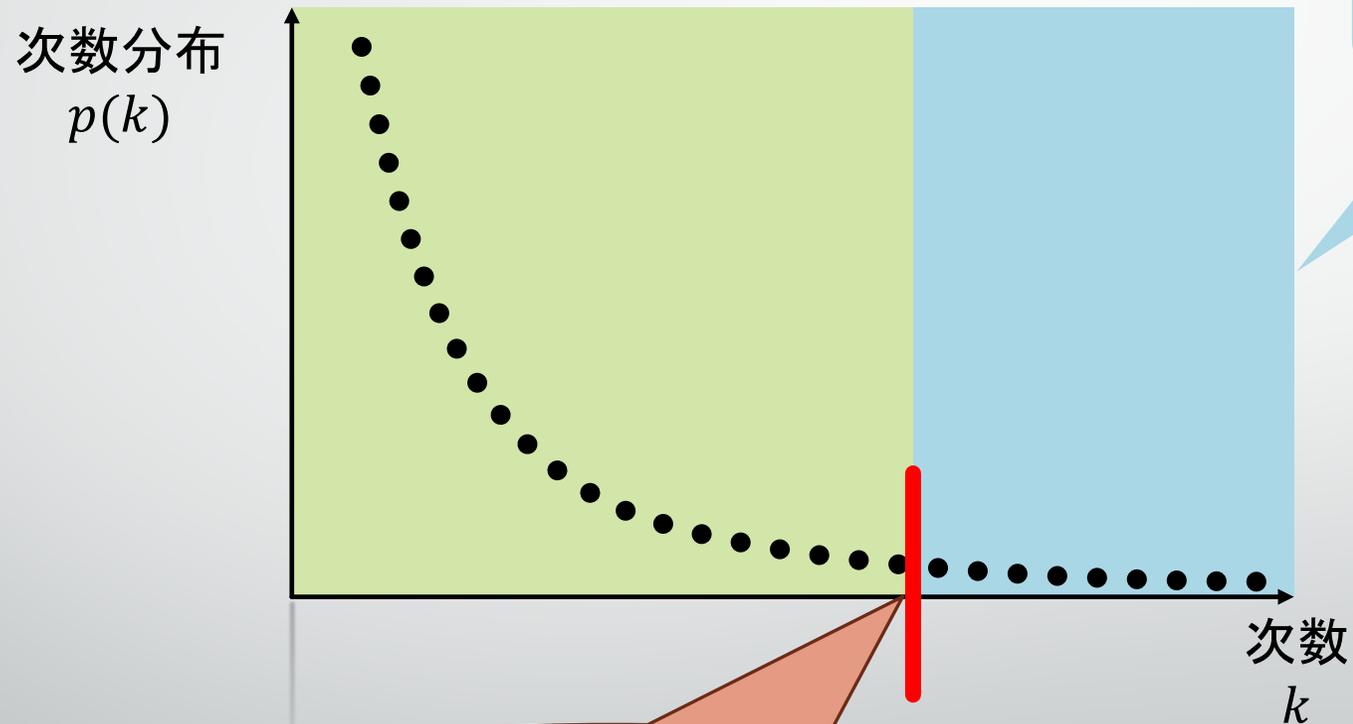
## 3.4 修正Configurationモデル



最大次数として設定

### 3.修正Configurationモデル

## 3.4 修正Configurationモデル

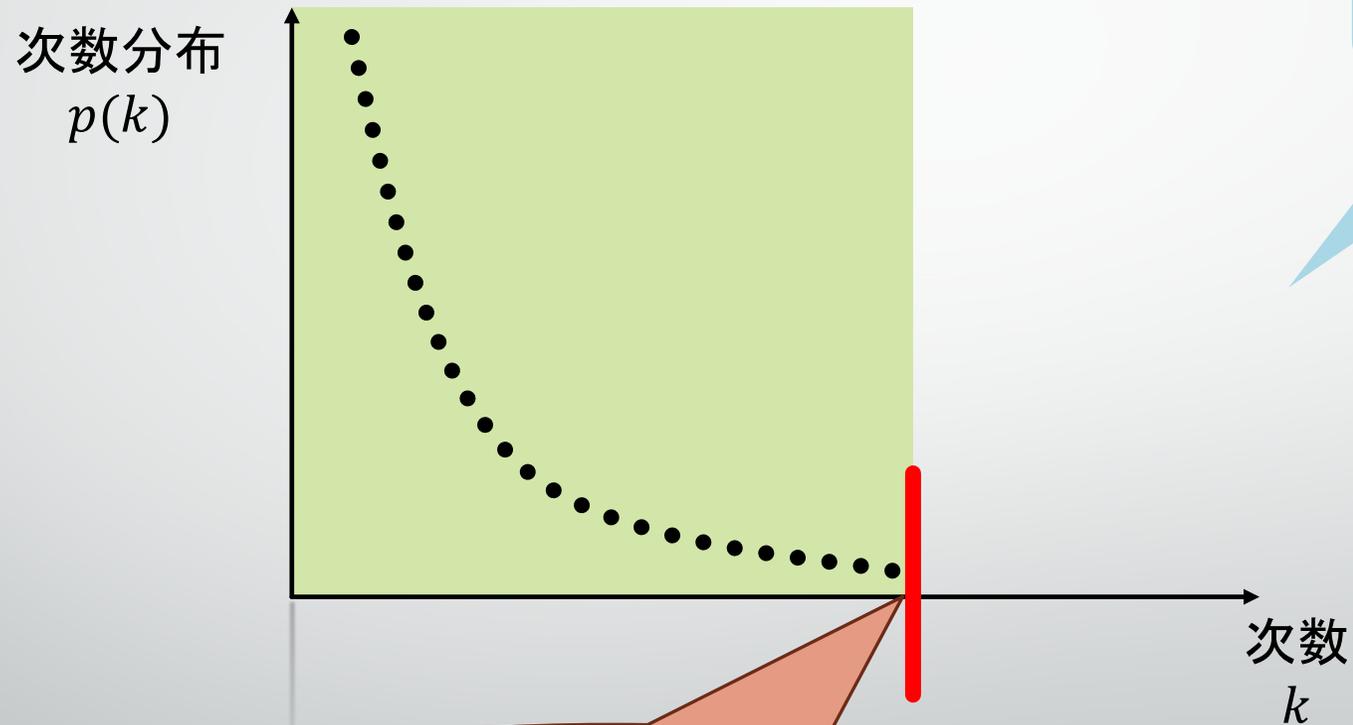


カットオフ

最大次数として設定

### 3.修正Configurationモデル

## 3.4 修正Configurationモデル



カットオフ

最大次数として設定

### 3.修正Configurationモデル

## 3.4 修正Configurationモデル

例 「Twitterのフォロワー数」

現在Twitterの全ユーザー数：3億1000万人 (2016年6月)[1]

- 世界で最もフォロワーが多いアカウント [2]  
「KATY PERRY」 フォロワー数：9600万人(全ユーザーの約31%)
- 日本で最もフォロワーが多いアカウント [3]  
「有吉弘行」 フォロワー数：640万人(全ユーザーの約2%)



最大次数をカットオフしても現実的には問題無い場合がある

## 3.修正Configurationモデル

# 3.4 修正Configurationモデル

### 参照ウェブページ

- [1] 【最新版】 2017年2月更新！ 11のソーシャルメディア最新動向データまとめ  
<http://gaiax-socialmedialab.jp/post-30833/>  
2017/01/30 参照
- [2] Twitter フォロワー数の世界ランキング【世界一フォローされている人】  
<https://matome.naver.jp/odai/2139806539969747601>  
2017/01/30 参照
- [3] Twitter日本 フォロワー数 総合ランキング 1-50位  
[http://meyou.jp/ranking/follower\\_allcat](http://meyou.jp/ranking/follower_allcat)  
2017/01/30 参照



## 4. 特徵量

4.1 直径 · 半径

4.2 中心性

## 4.特徴量

# 4.1 直径・半径

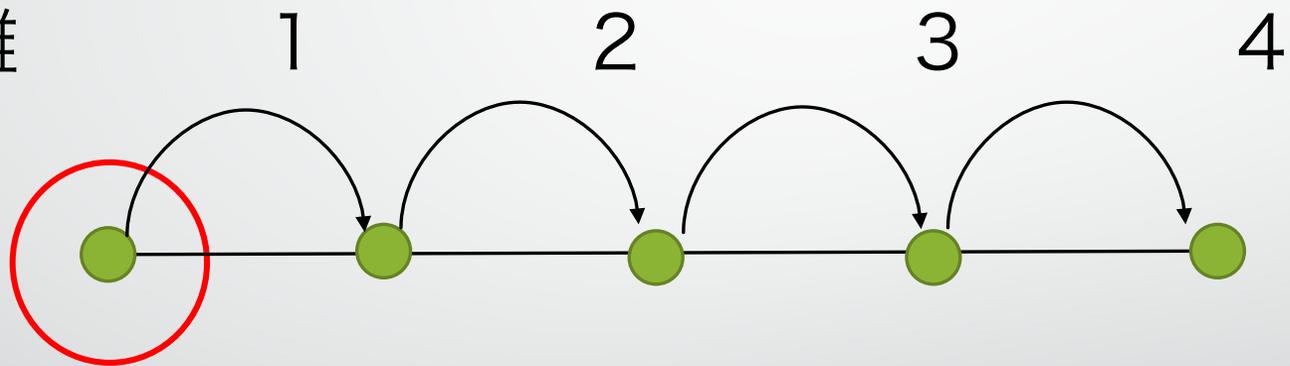
直径・半径を求めるに当たって

- Eccentricity値
  - ・ 各頂点に一つ定まる値
  - ・ 任意の頂点とその頂点から一番遠い頂点との距離で定まる

# 4. 特徵量

## 4.1 直徑 · 半徑

距離

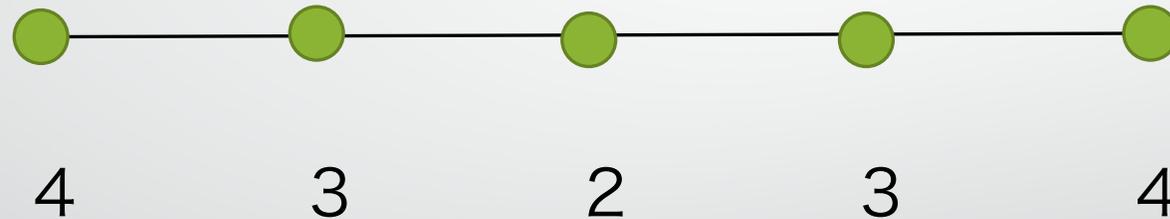


eccentricity = 4

## 4. 特徴量

### 4.1 直径・半径

- すべての頂点のeccentricity値を求める



## 4.特徴量

# 4.1 直径・半径

- 直径(diameter)

ある頂点から各頂点までの最短経路のうち距離が最大のもの

- 半径(radius)

ある頂点から各頂点までの最短経路のうち距離が最小のもの

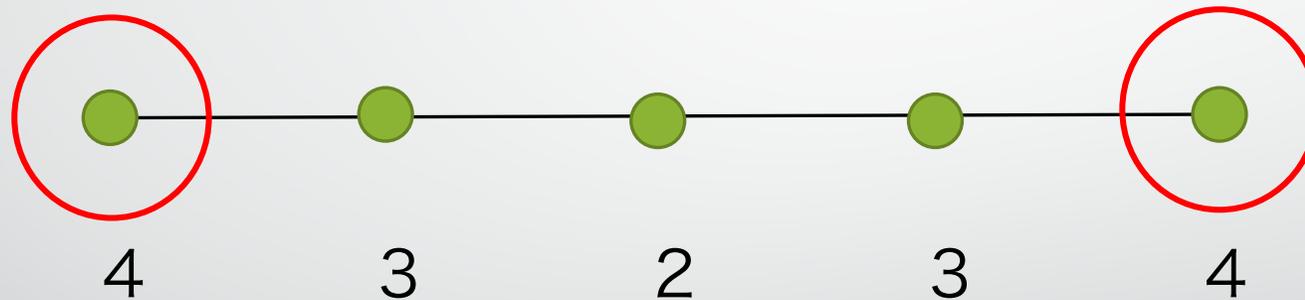
ある頂点から各頂点までの最短経路 = eccentricity値

## 4. 特徴量

### 4.1 直径・半径

直径：

eccentricity値のうち最大のものの

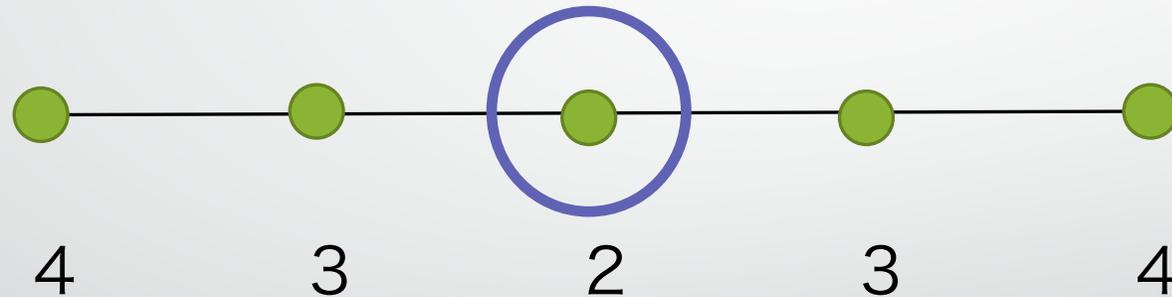


## 4.特徴量

### 4.1 直径・半径

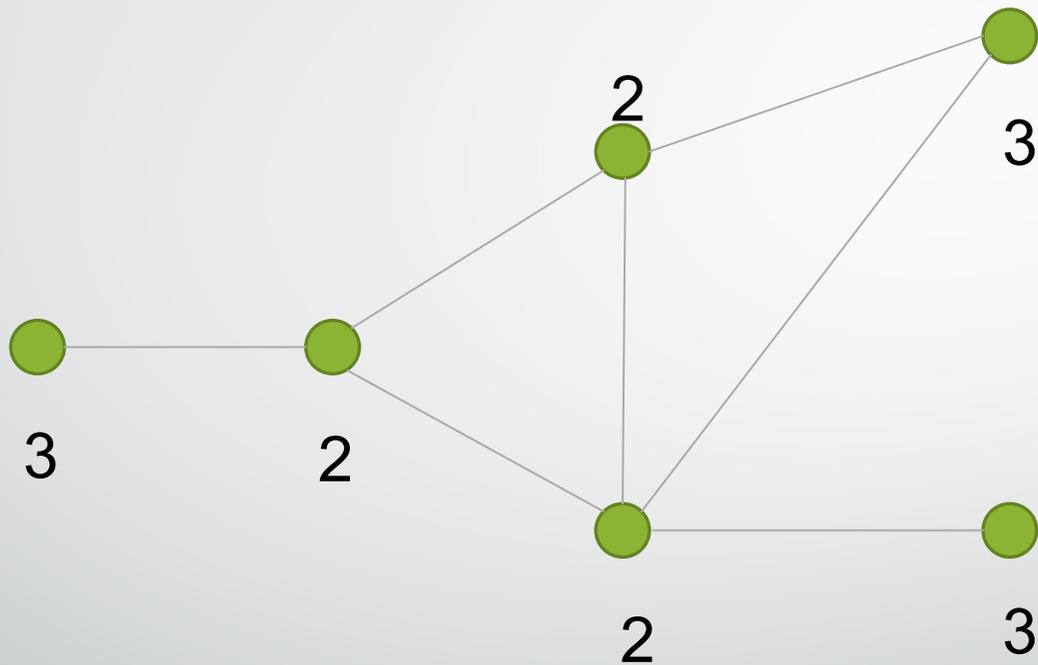
半径：

eccentricity値のうち最小のもの



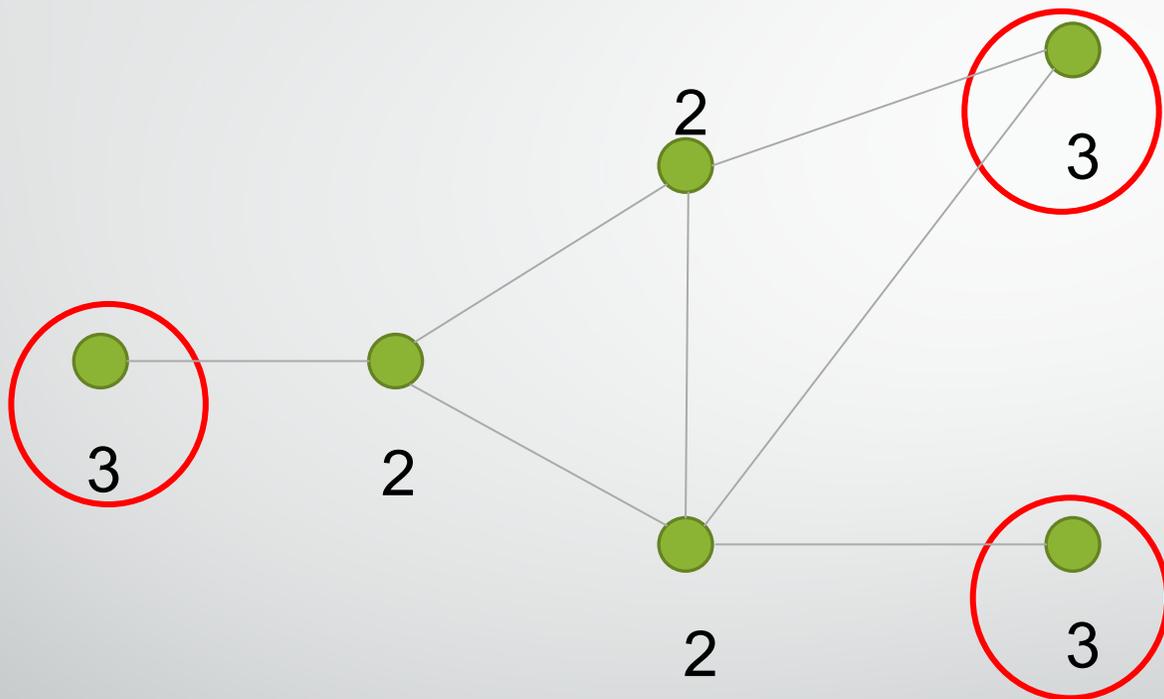
## 4. 特徴量

### 4.1 直径・半径



## 4. 特徴量

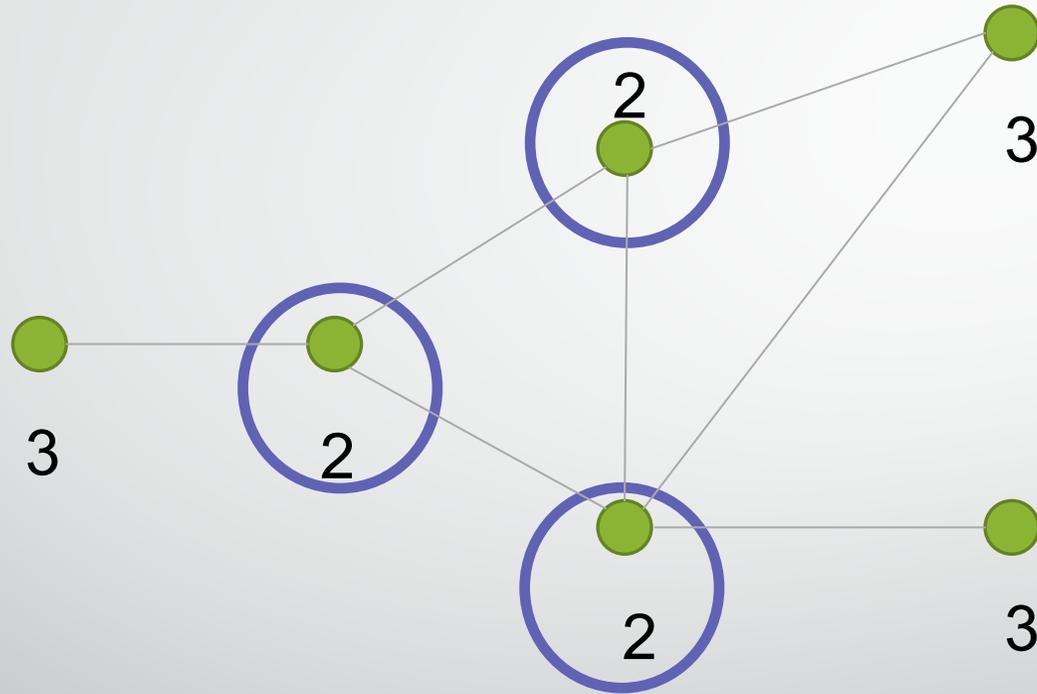
### 4.1 直径・半径



直径：  
eccentricity値のうち最大のもの

## 4.特徴量

### 4.1 直径・半径



半径：  
eccentricity値のうち最小のもの

## 4.特徴量

### 4.2 中心性

- 中心性
  - ある頂点をネットワーク全体から見たとき重要かどうかを評価する尺度
- ネットワークの中心性
  - 次数中心性
  - 近接中心性
  - 媒介中心性
  - 固有ベクトル中心性

## 4. 特徴量

### 4.2 中心性

- 近接中心性

ある頂点が他の頂点から平均的にどれくらい近いかどうかの指標

[定義]

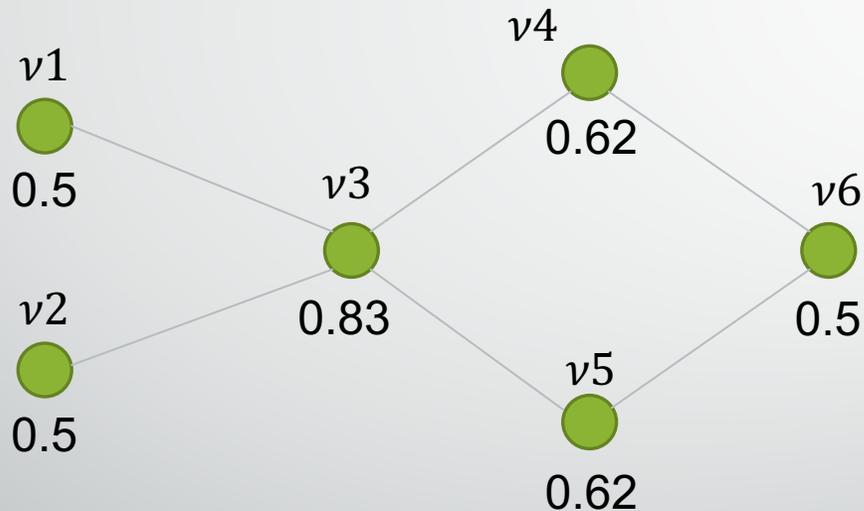
$$C_c(v) = \frac{N-1}{\sum_{t \in V} d(v,t)} \quad d(v,t): v, t間の最短距離$$

## 4. 特徴量

### 4.2 中心性

$$C_c(v_3) = \frac{6 - 1}{1 + 1 + 0 + 1 + 1 + 2} = \frac{5}{6} = 0.83 \dots$$

$$C_c(v) = \frac{N - 1}{\sum_{t \in V} d(v, t)}$$



ある頂点の近接中心性は、  
他のN-1個のすべての頂点への距離の平均を求めている

## 4.特徴量

### 4.2 中心性

- 媒介中心性

ある2頂点間の最短経路内にその頂点がどのくらい関わっているのかどうかを表す指標

[定義]

$$C_B(v) = \sum_{\substack{s,v,t \in V \\ s \neq v \neq t}} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

$\sigma_{st}$ : s, t間の最短経路数

$\sigma_{st}(v)$ : vを通るs, t間の最短経路数

# 4. 特徴量

## 4.2 中心性

● 例



$\sigma_{st}$ :  $s, t$ 間の最短経路数

$\sigma_{st}(v)$ :  $v$ を通る $s, t$ 間の最短経路数

$$C_B(v) = \sum_{\substack{s,v,t \in V \\ s \neq v \neq t}} \frac{\sigma_{st}(v)}{\sigma_{st}}$$



# 5. 実験方法

5.1 実験の方針

5.2 CMで生成するネットワークの条件

5.3 生成されたネットワークの比較方法

## 2.背景・演習の目的 (再掲)2.3 演習の目的

### 本演習の目的

乱数生成時に使用するseedを変更し、  
実験結果がseedに依存しないか検証する

## 5.実験方法

### 5.1 実験の方針

- seedを以下の4種類で定めてConfigurationモデルでそれぞれネットワークを生成する
  - 1.時刻 $T_0, T_1, \dots, T_{99}$
  - 2.時刻 $T'_0, T'_1, \dots, T'_{99}$
  - 3.id(00~99)の整数値
  - 4.id(17000~17099)の整数値
- 生成したネットワークの特徴量を計算・比較し,seedの違うネットワークの特徴量に大きな差がないかを検証する

## 5.実験方法

# 5.2 生成するネットワークの条件

- 生成するネットワークの詳細

頂点数 : 10000

最小次数 : 2

べき指数 : 5種類 {2.0, 2.25, 2.50, 2.75, 3.0}

seed : 4パターン {時刻群1(100回), 時刻群2(100回)  
id(00~99), id(17000~17099)}

乱数生成器 : メルセンヌ・ツイスタ

## 5.実験方法

# 5.2 生成するネットワークの条件

- ネットワーク生成方法と生成数

以下のモデルでそれぞれネットワークを**100個**生成

1. Configurationモデル(自己ループ・多重辺あり)
2. Configurationモデル(自己ループ・多重辺なし)
3. 修正Configurationモデル(自己ループ・多重辺あり)
4. 修正Configurationモデル(自己ループ・多重辺なし)

(注) 以後実験結果では下記のように略記

Configurationモデル → CM

修正Configurationモデル → UCM

(自己ループ・多重辺あり) → (al)

## 5.実験方法

# 5.2 生成するネットワークの条件

- 生成するネットワーク総数

生成モデル: 4種類

べき指数の種類: 5種類

seedの取り方: 4パターン

生成数: 100個

8000個

## 5.実験方法

### 5.3 ネットワークの比較

- 生成したネットワークに対して以下の特徴量を計算
  1. 直径・半径
  2. 近接中心性・媒介中心性
- 計算した特徴量をseedの取り方の異なる4グループ間で比較

## 5.実験方法

### 5.3 ネットワークの比較

- 近接中心性・媒介中心性

→ 「中心」の概念から、ネットワークに存在する中心は一つ(少数)

→ 近接中心性・媒介中心性の**最大値**を計算し、ネットワーク同士で比較

## 5.実験方法

### 5.3 ネットワークの比較

seed = 時刻群1の  
ネットワークの特徴量

seed = 時刻群2の  
ネットワークの特徴量

seed = 00~99の  
ネットワークの特徴量

seed = 17000~17099の  
ネットワークの特徴量

オープンソースデータ分析ツール「R」で  
グラフ描画し結果を視覚化

## 5.実験方法

# 5.3 ネットワークの比較

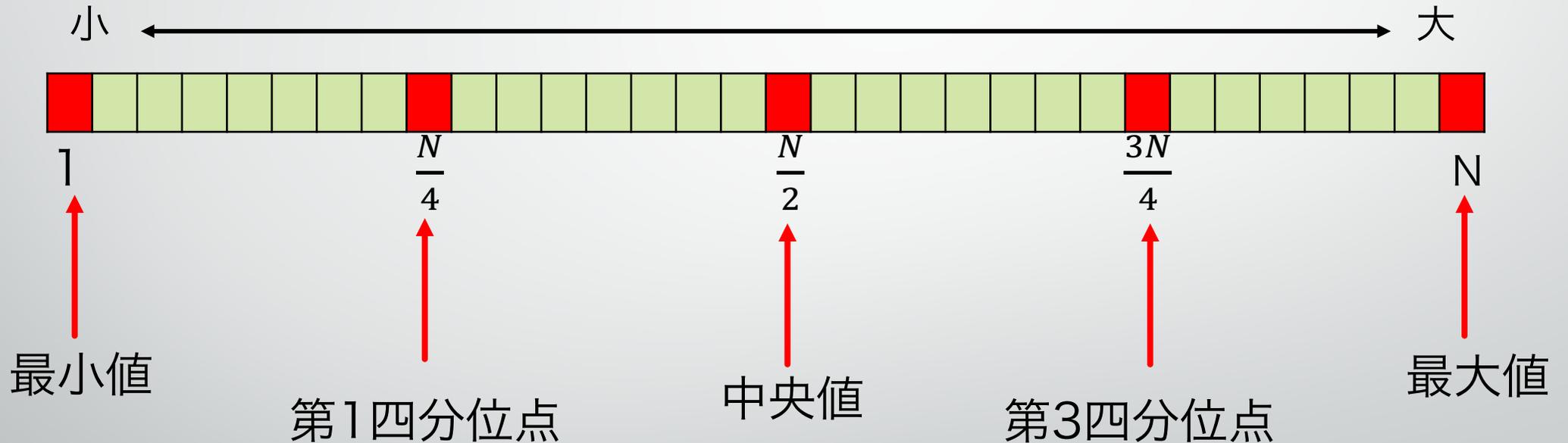
- ネットワークの特徴量を箱ひげ図を用いて描画  
直径, 半径, 近接中心性, 媒介中心性
- seedの取り方の異なるネットワークのクラスタリング  
K-medoid法によるクラスタリング  
最適クラスタ数の計算

## 5.実験方法

# 5.3 ネットワークの比較

- 箱ひげ図の読み方(言葉の説明)

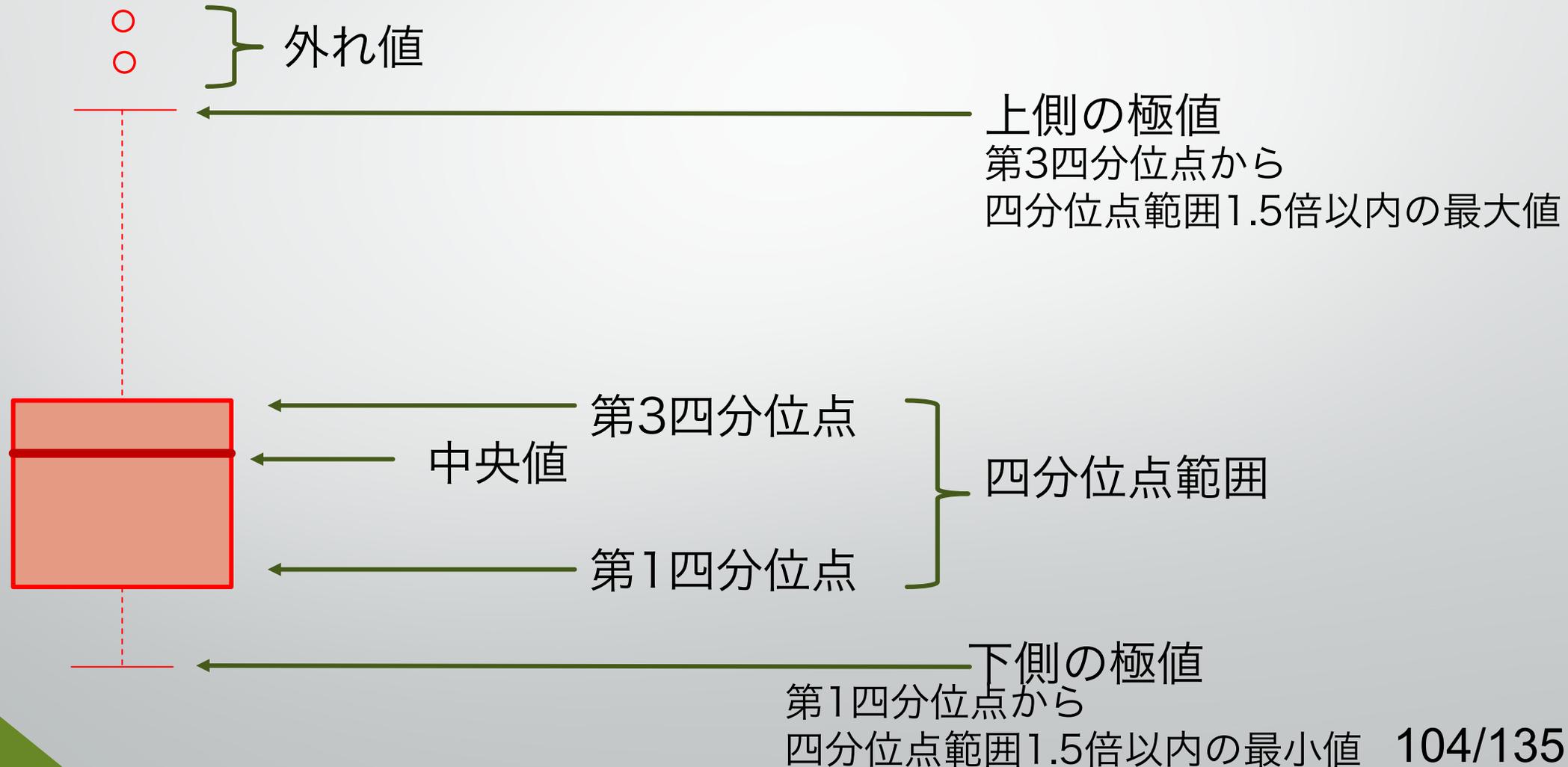
使用するN個のデータを昇順に並べて,四分分割する



## 5.実験方法

### 5.3 ネットワークの比較

- Rの箱ひげ図(box\_plot)の読み方



## 5.実験方法

# 5.3 ネットワークの比較

- ネットワークのクラスタリング

生成手法,べき指数を統一し,seedの取り方が異なる4グループのネットワークに対し,各ネットワークの特徴量を属性値としてクラスタリング

予想→生成結果がseedに依存していなければクラスタ数は少数では？

## 5.3 ネットワークの比較

### 各ネットワークの属性値

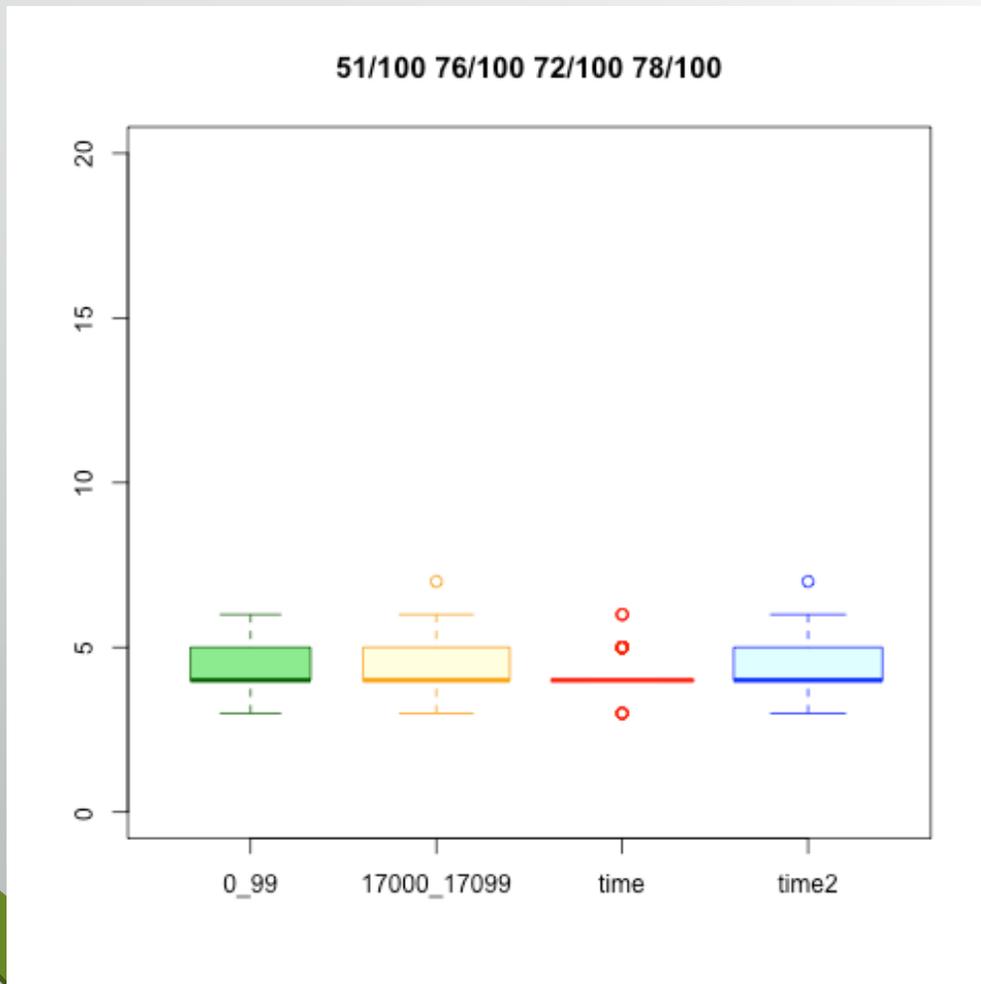
- 直径
- 半径
- 平均距離
- 各頂点の次数(10000個)
- 近接中心性の値の大きいものから100個
- 媒介中心性の値の大きいものから100個



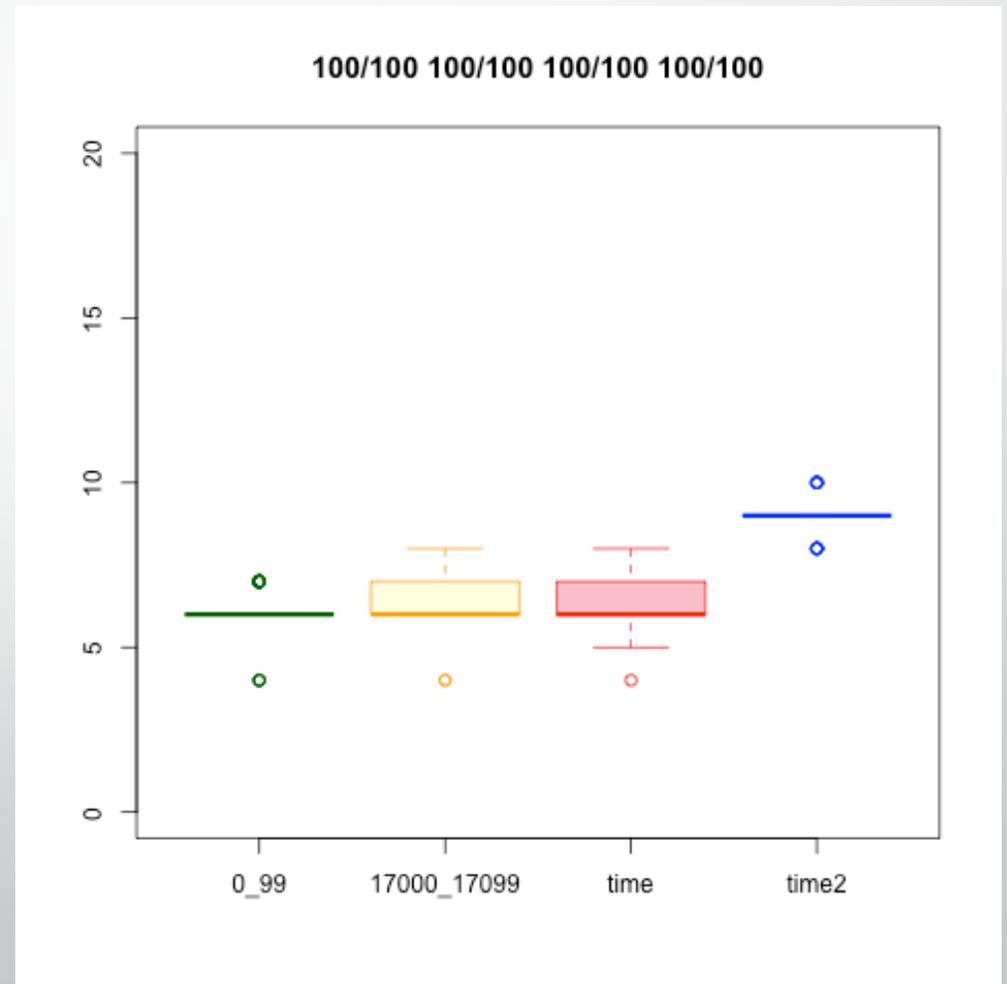
## 6. 実験結果

# 6. 実験結果

CM  $\gamma = 2.0$  直径

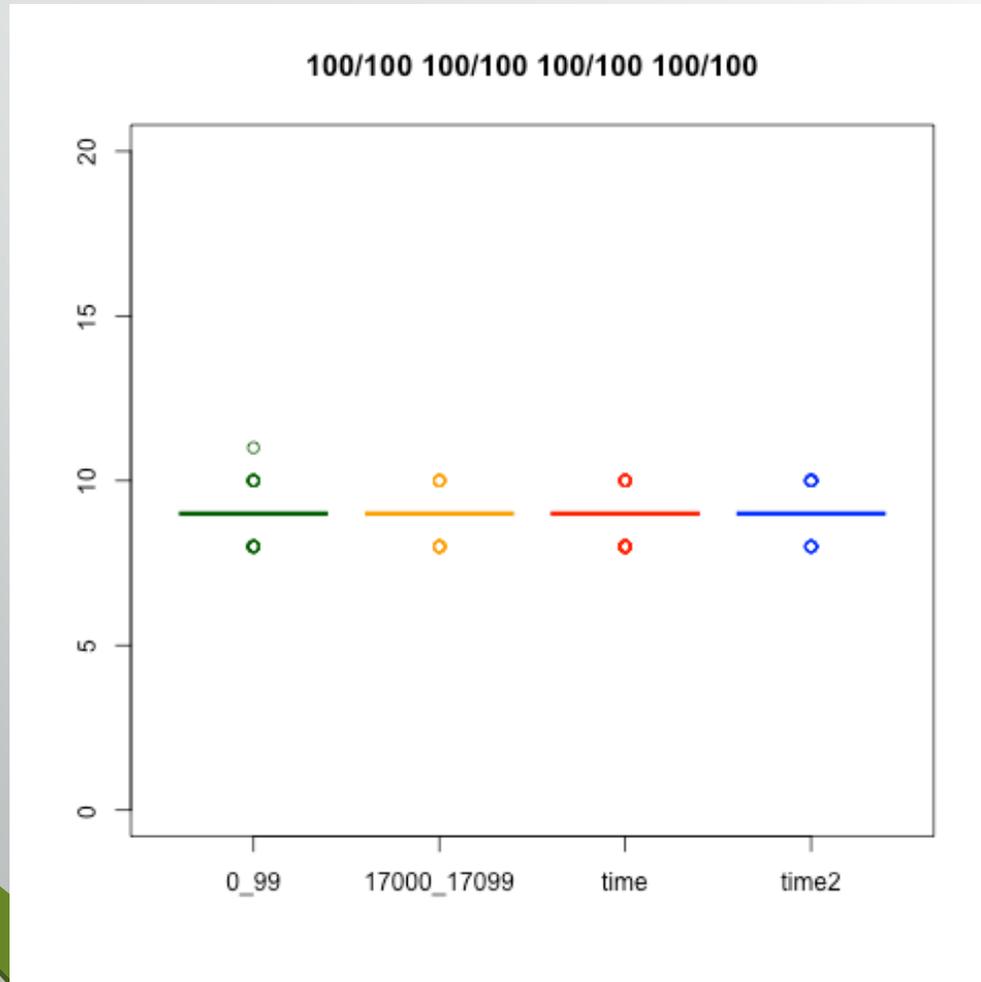


CM(al)  $\gamma = 2.0$  直径

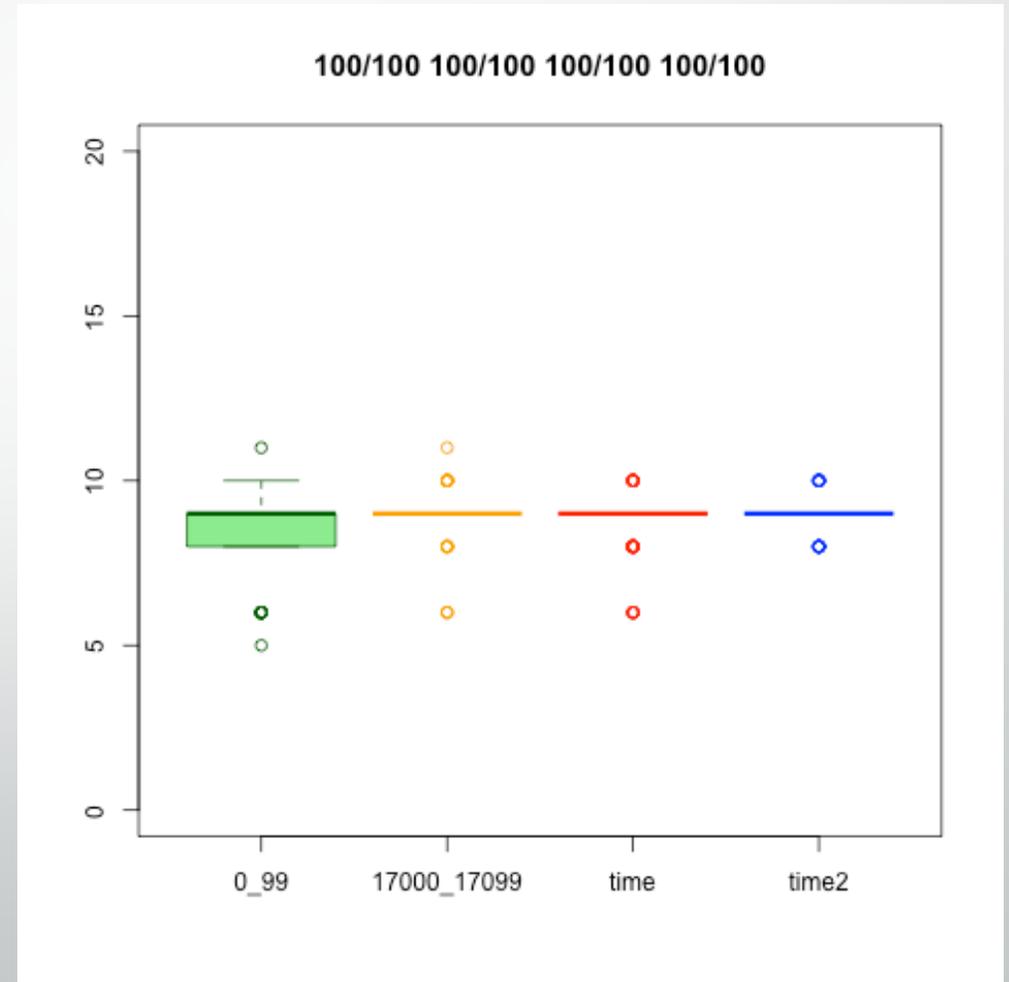


# 6. 実験結果

UCM  $\gamma = 2.0$  直径

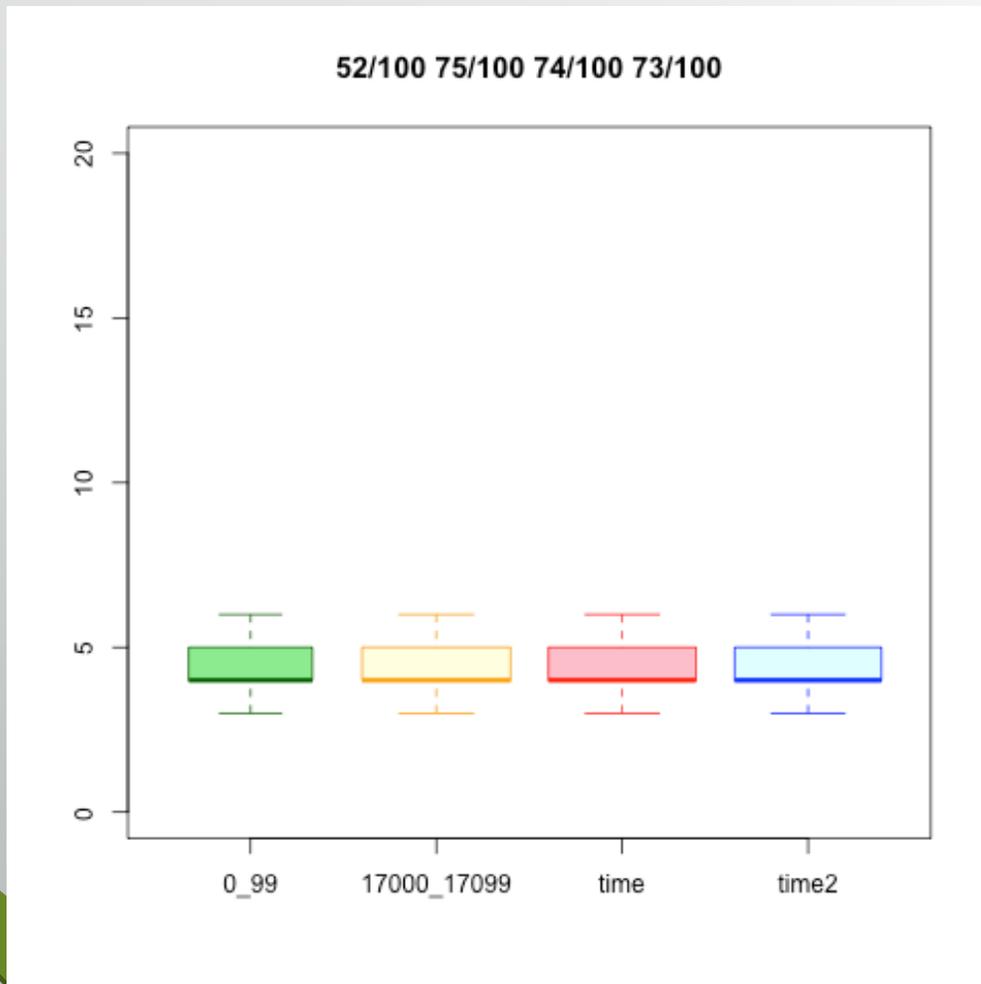


UCM(al)  $\gamma = 2.0$  直径

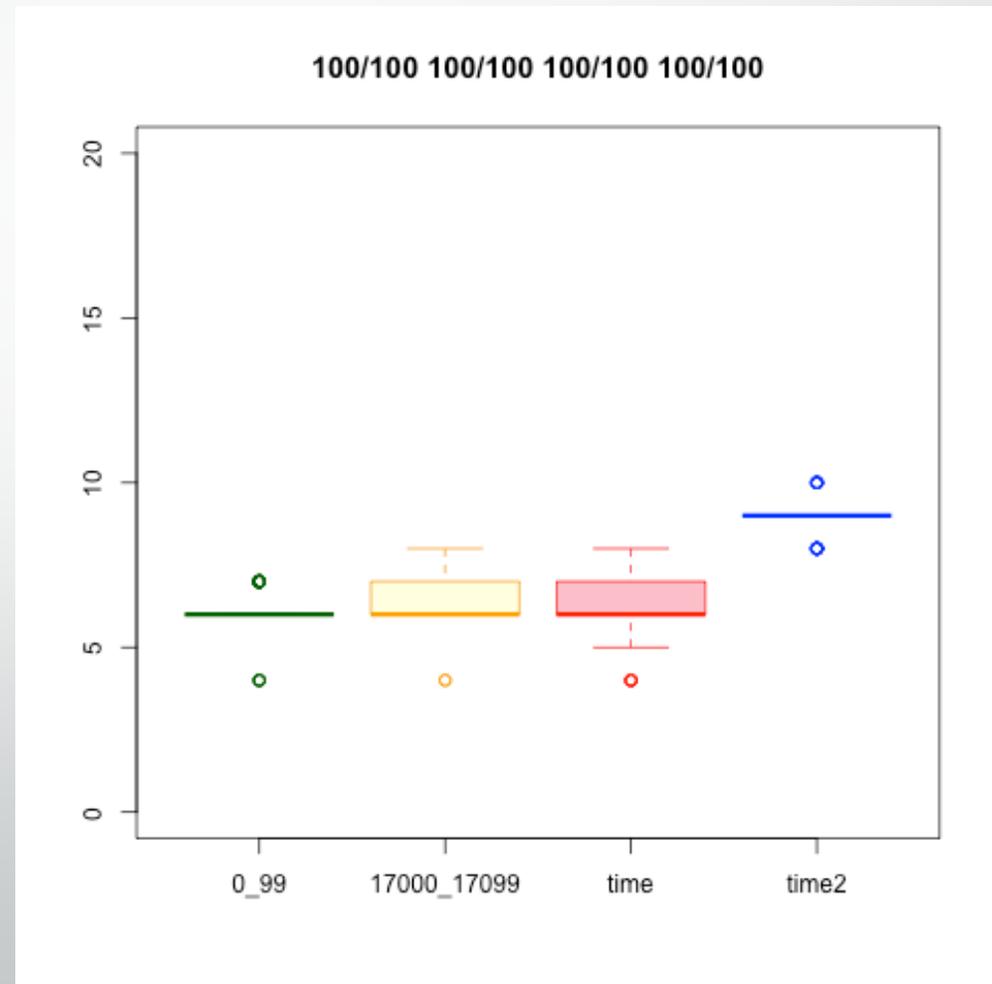


# 6. 実験結果

CM  $\gamma = 2.25$  直径

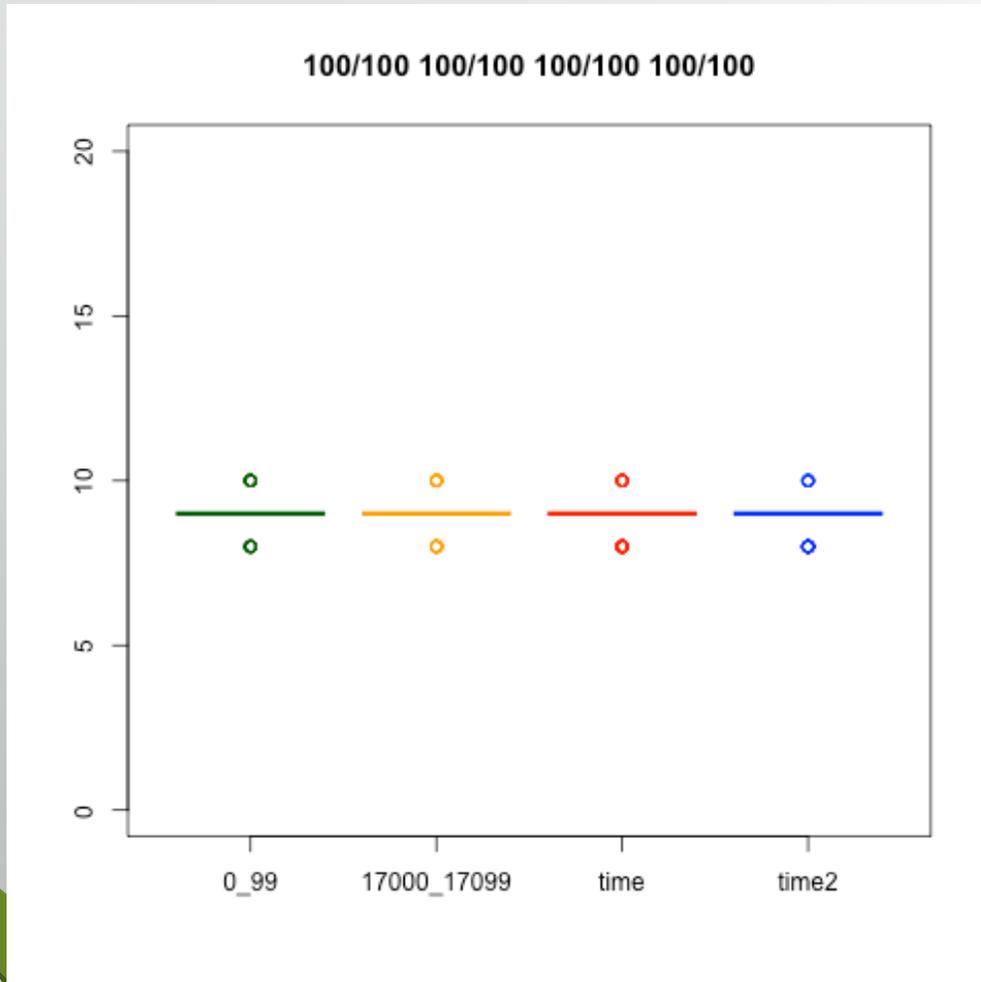


CM(al)  $\gamma = 2.25$  直径

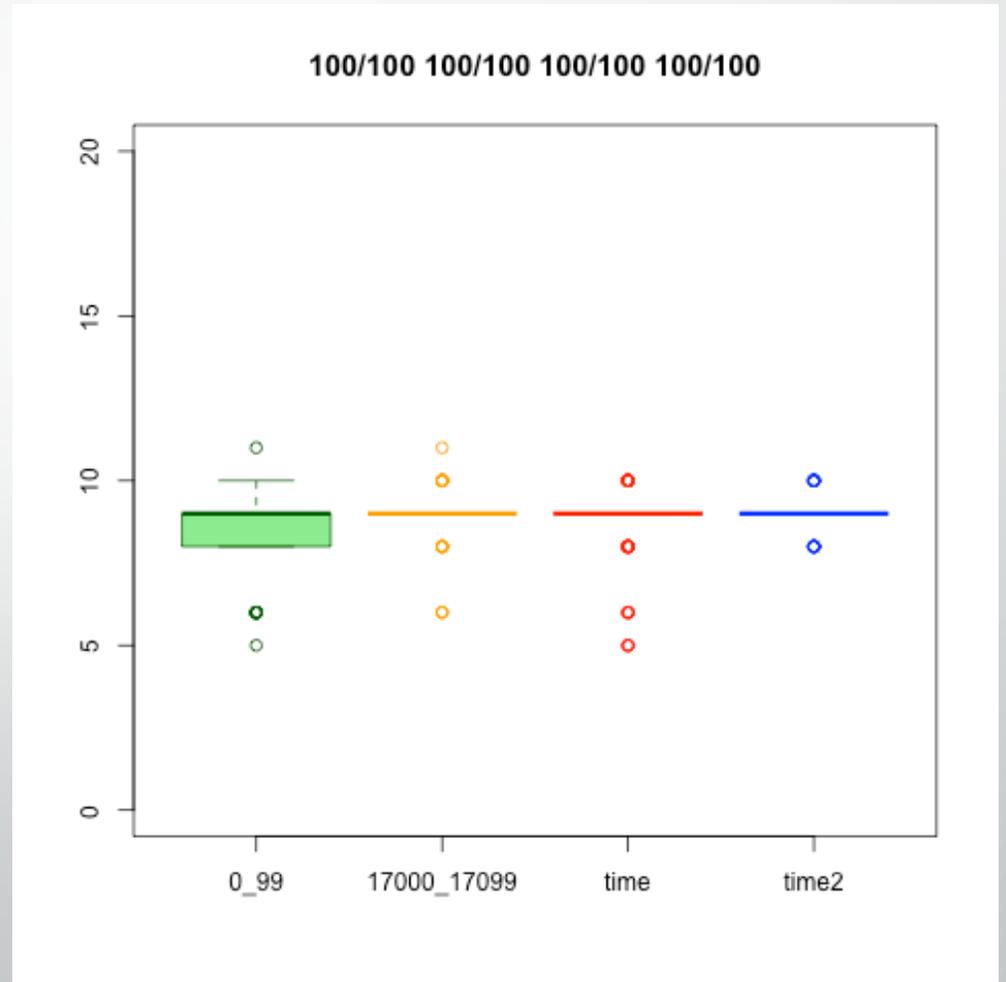


# 6. 実験結果

UCM  $\gamma = 2.25$  直径

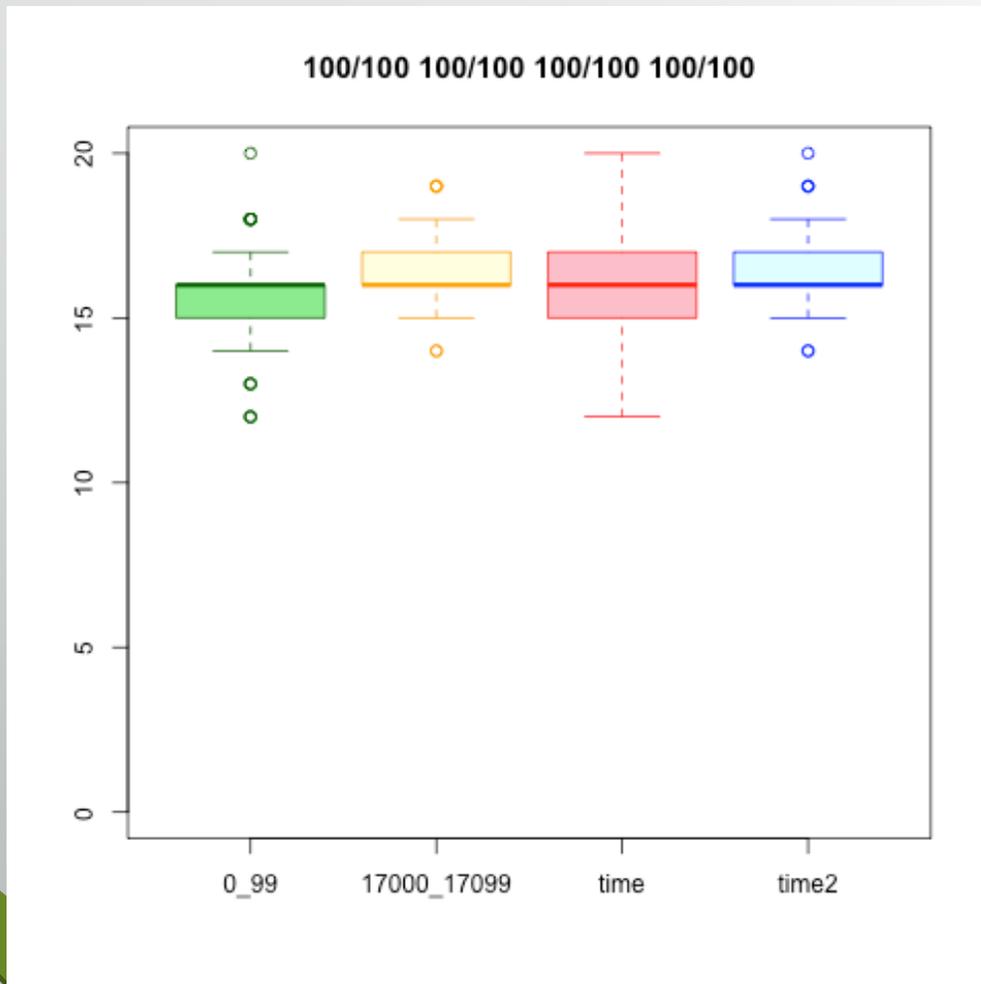


UCM(al)  $\gamma = 2.25$  直径

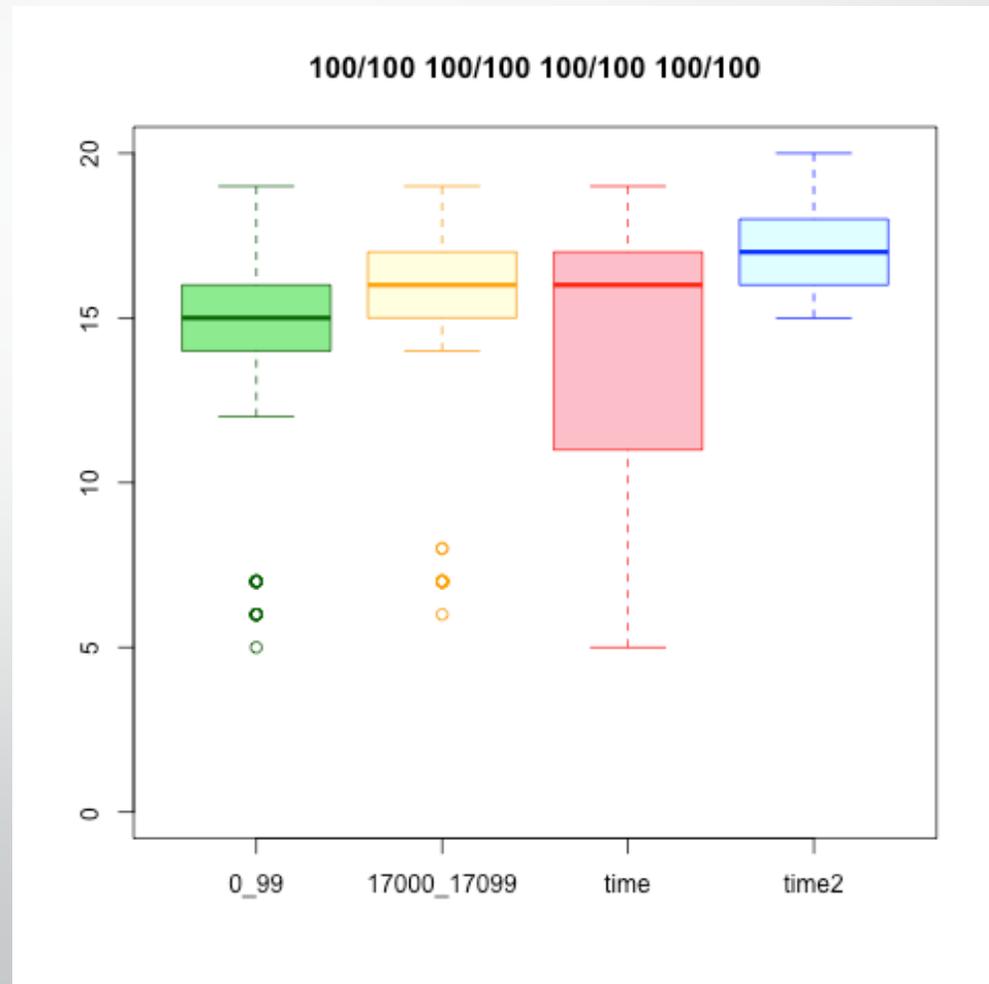


# 6. 実験結果

CM  $\gamma = 3.0$  直径

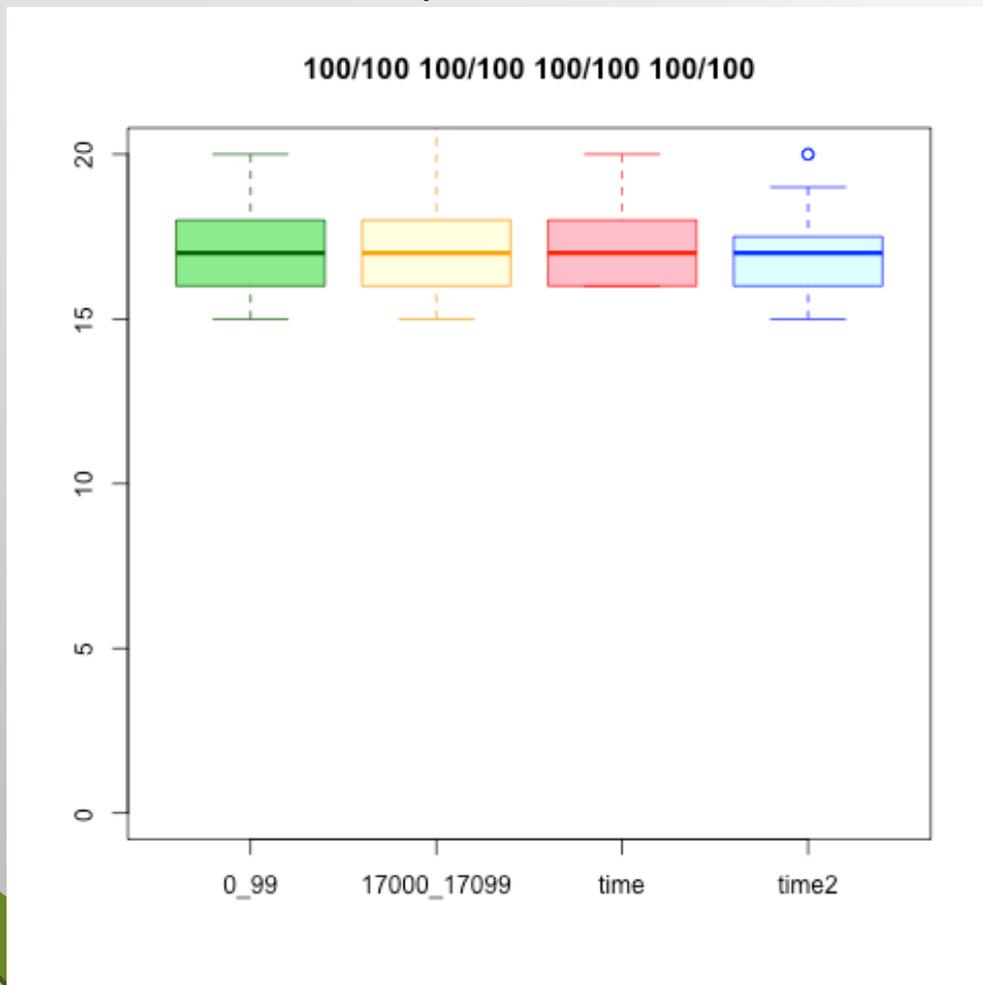


CM(al)  $\gamma = 3.0$  直径

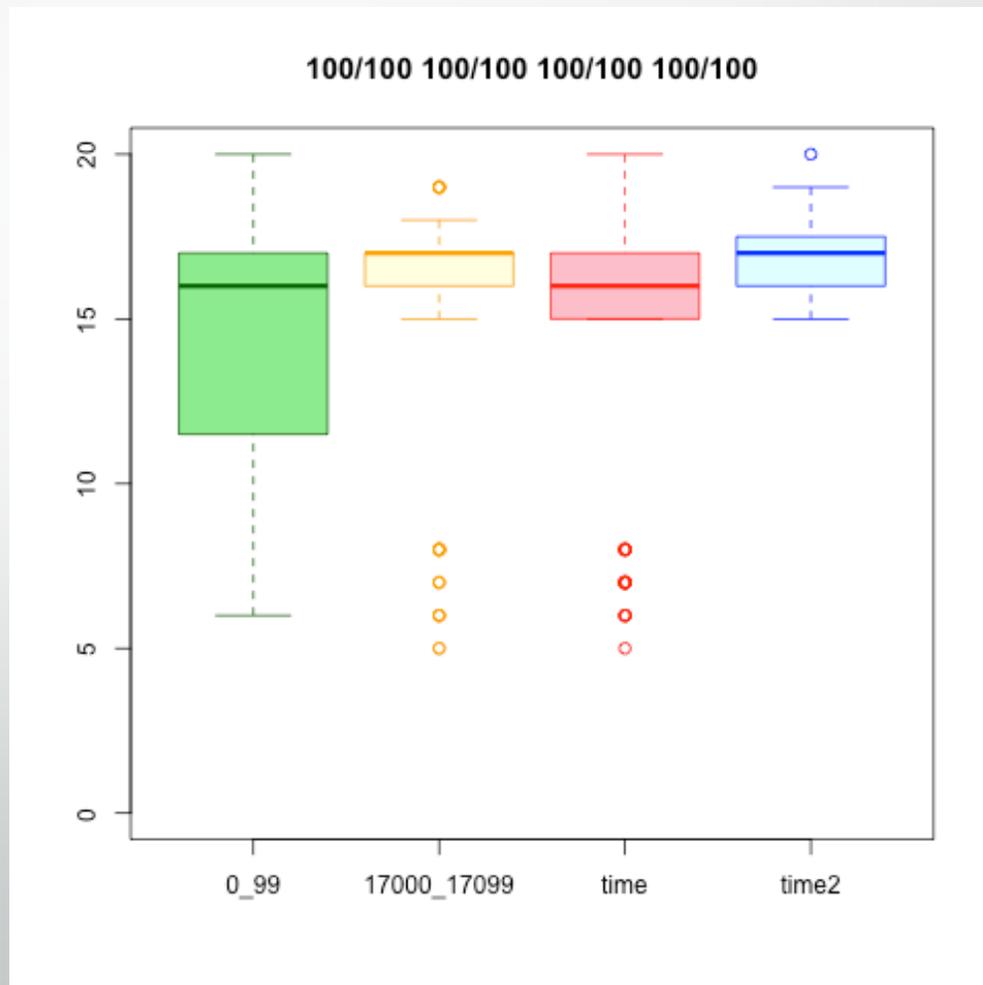


# 6. 実験結果

UCM  $\gamma = 3.0$  直径

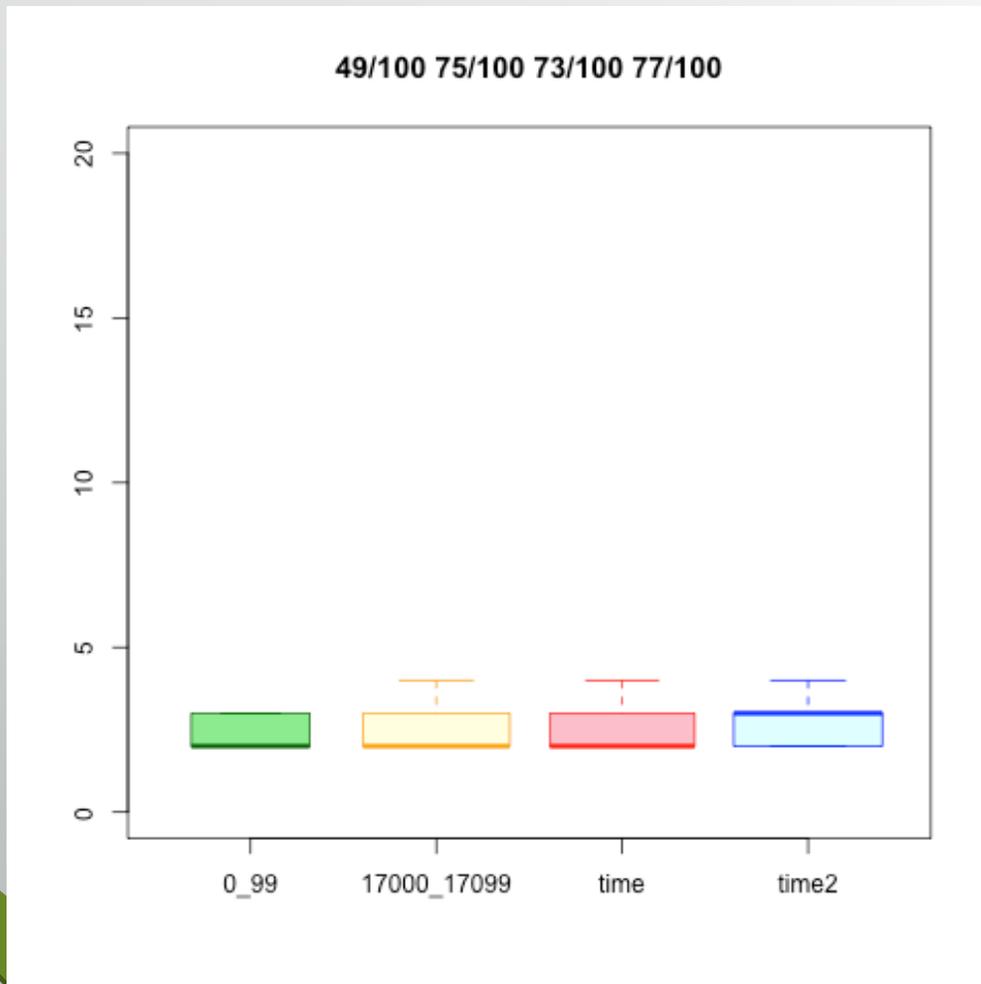


UCM(al)  $\gamma = 3.0$  直径

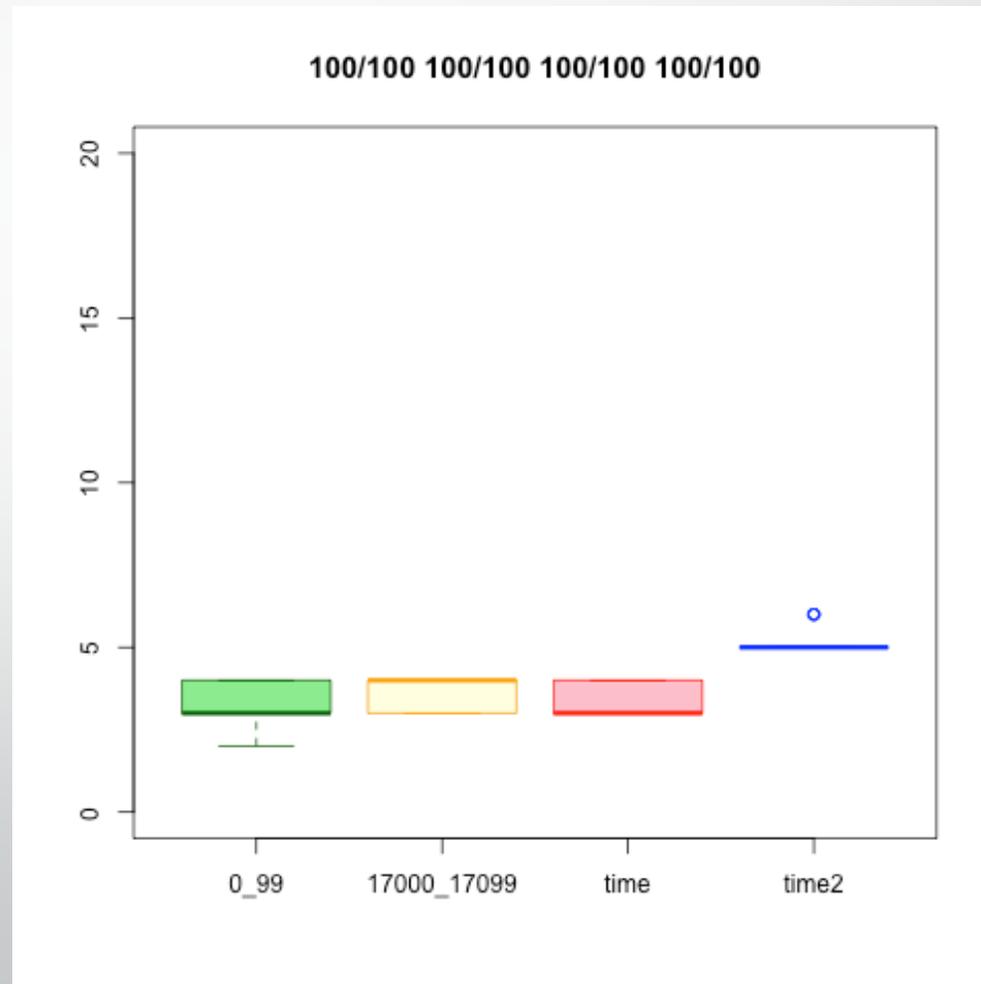


# 6. 実験結果

CM  $\gamma = 2.75$  半径

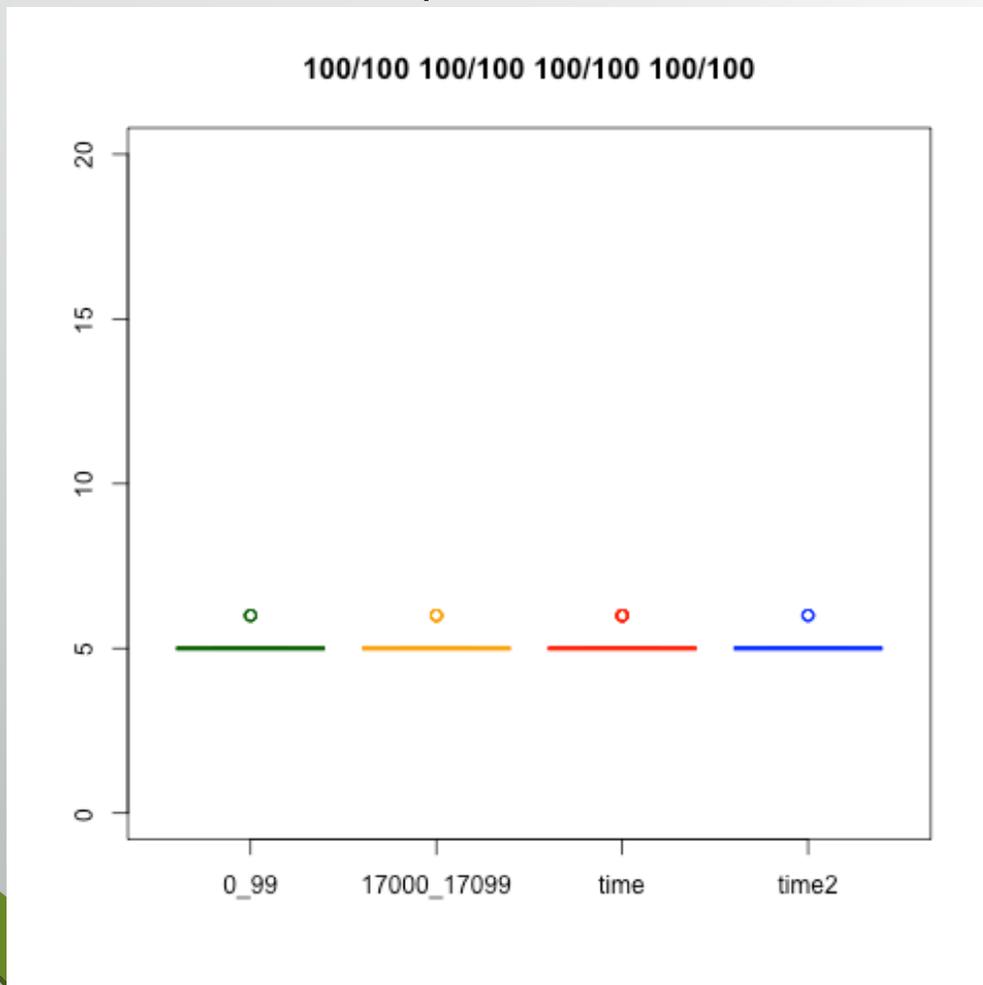


CM(al)  $\gamma = 2.75$  半径

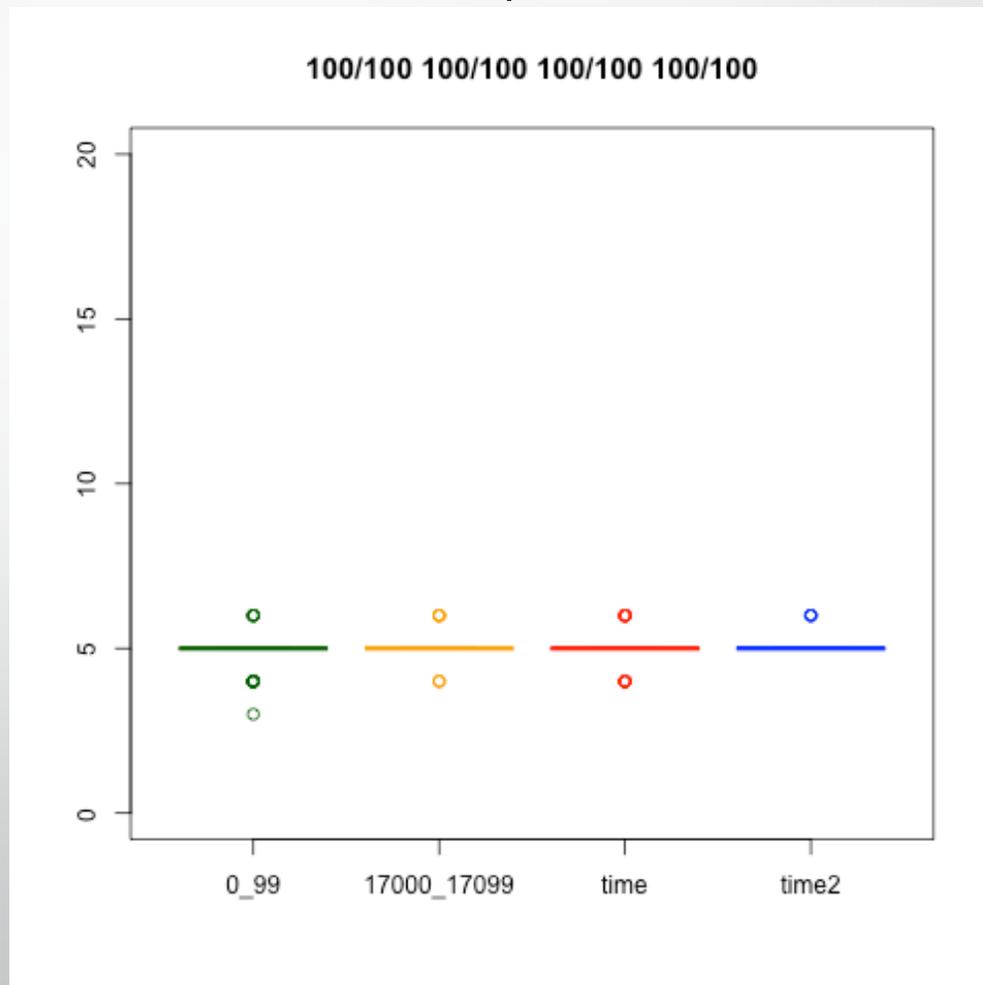


# 6. 実験結果

UCM  $\gamma = 2.75$  半径

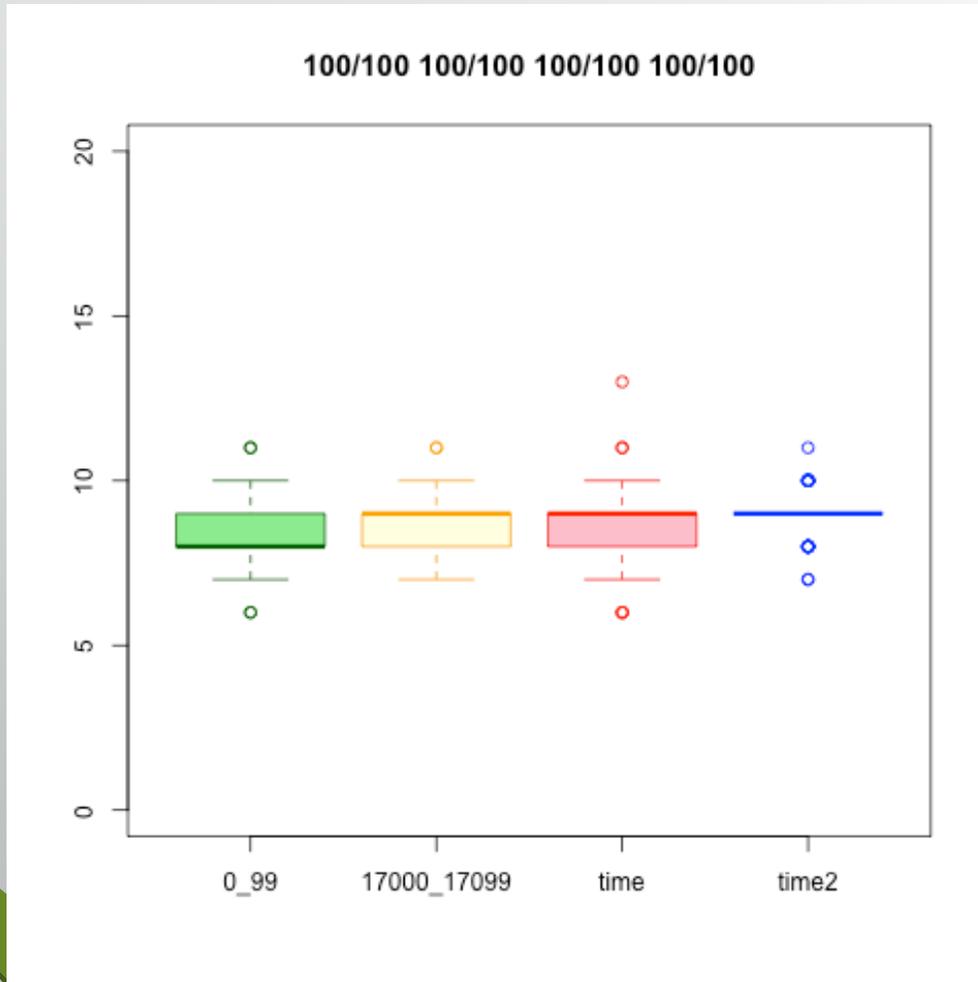


UCM(al)  $\gamma = 2.75$  半径

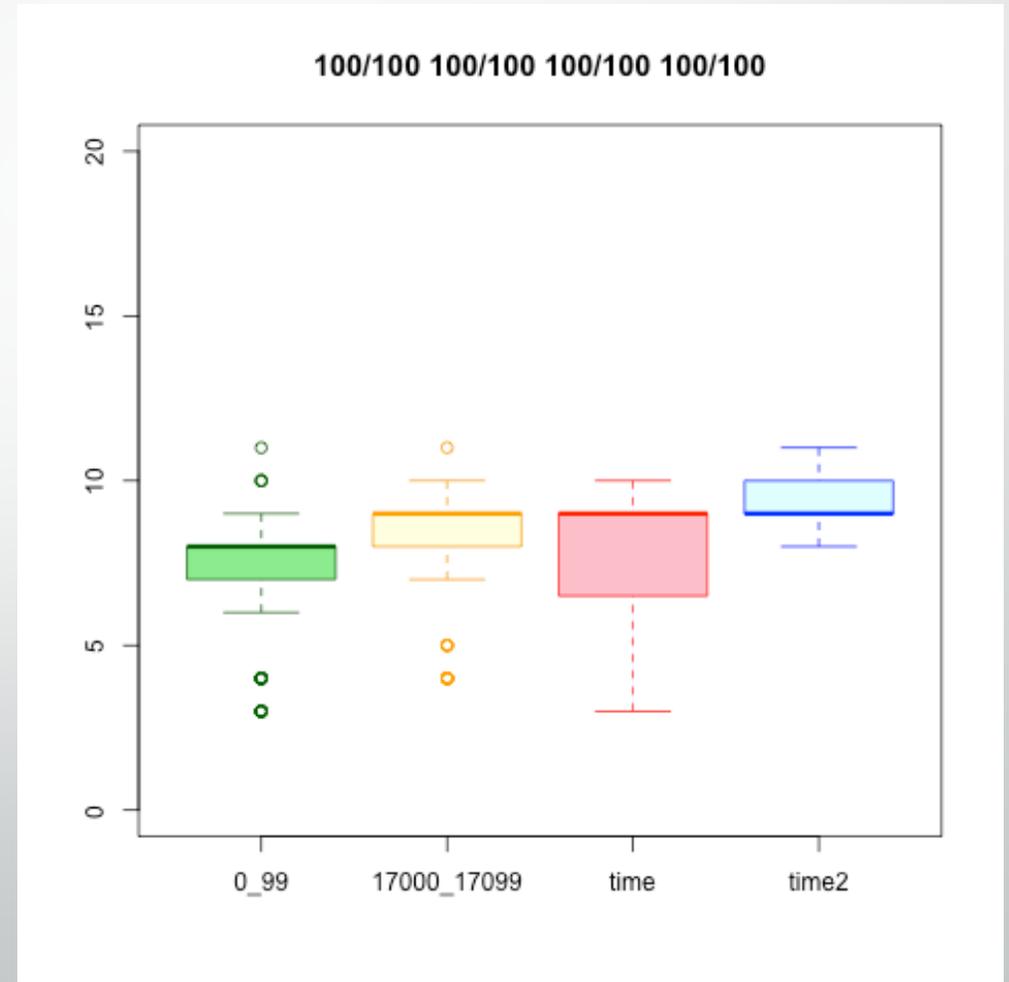


# 6. 実験結果

CM  $\gamma = 3.0$  半径

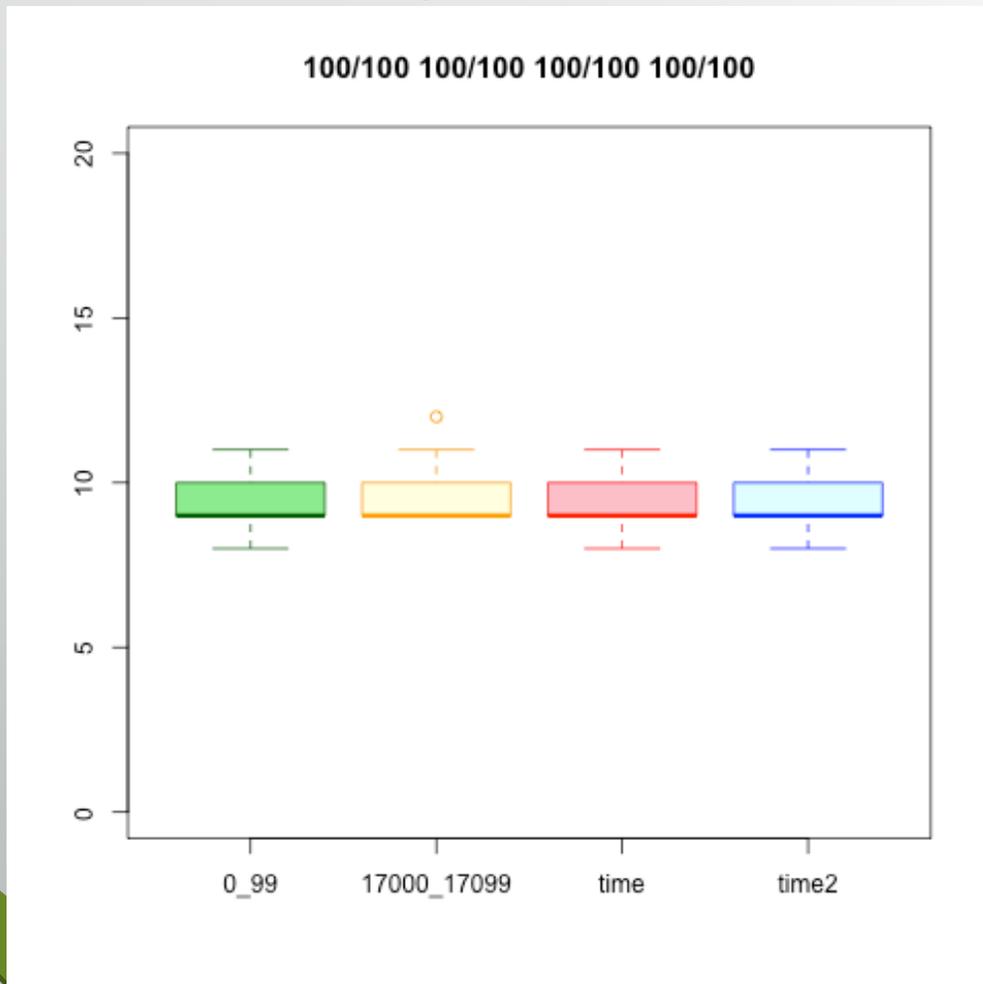


CM(al)  $\gamma = 3.0$  半径

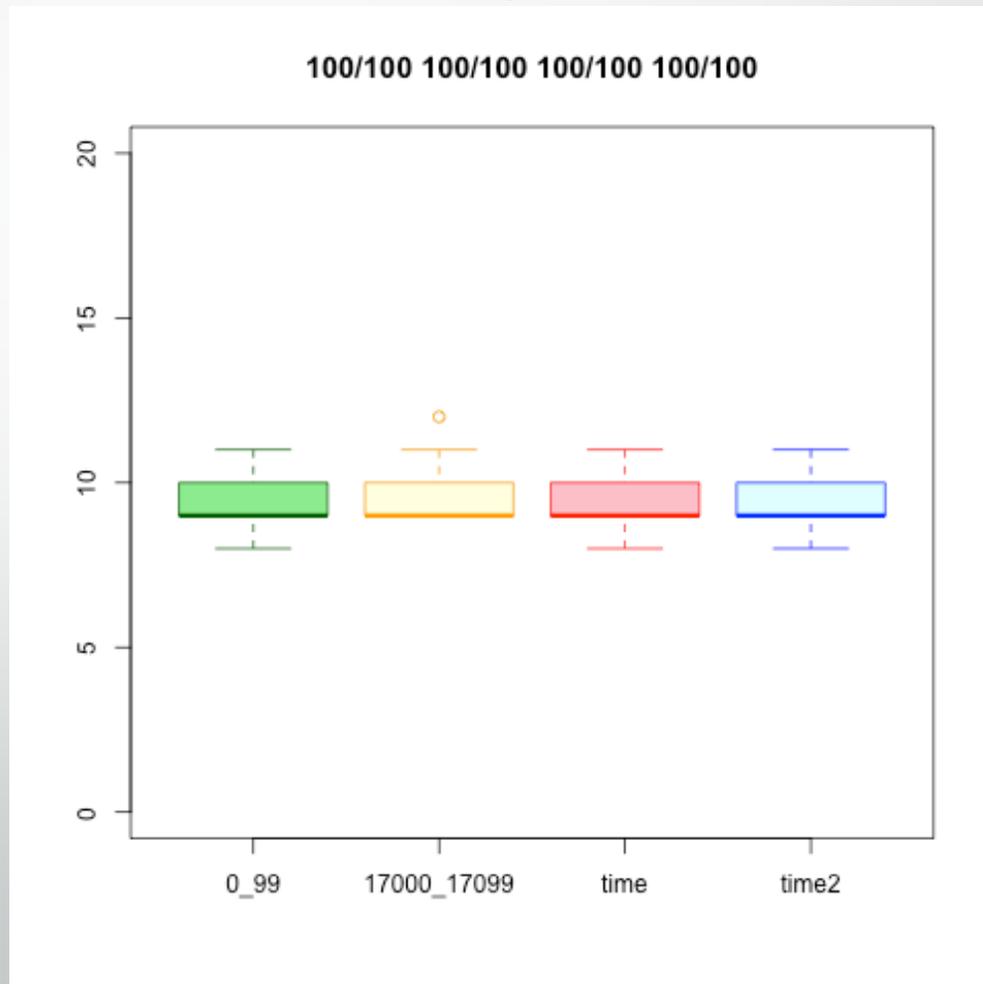


# 6. 実験結果

UCM  $\gamma = 3.0$  半径

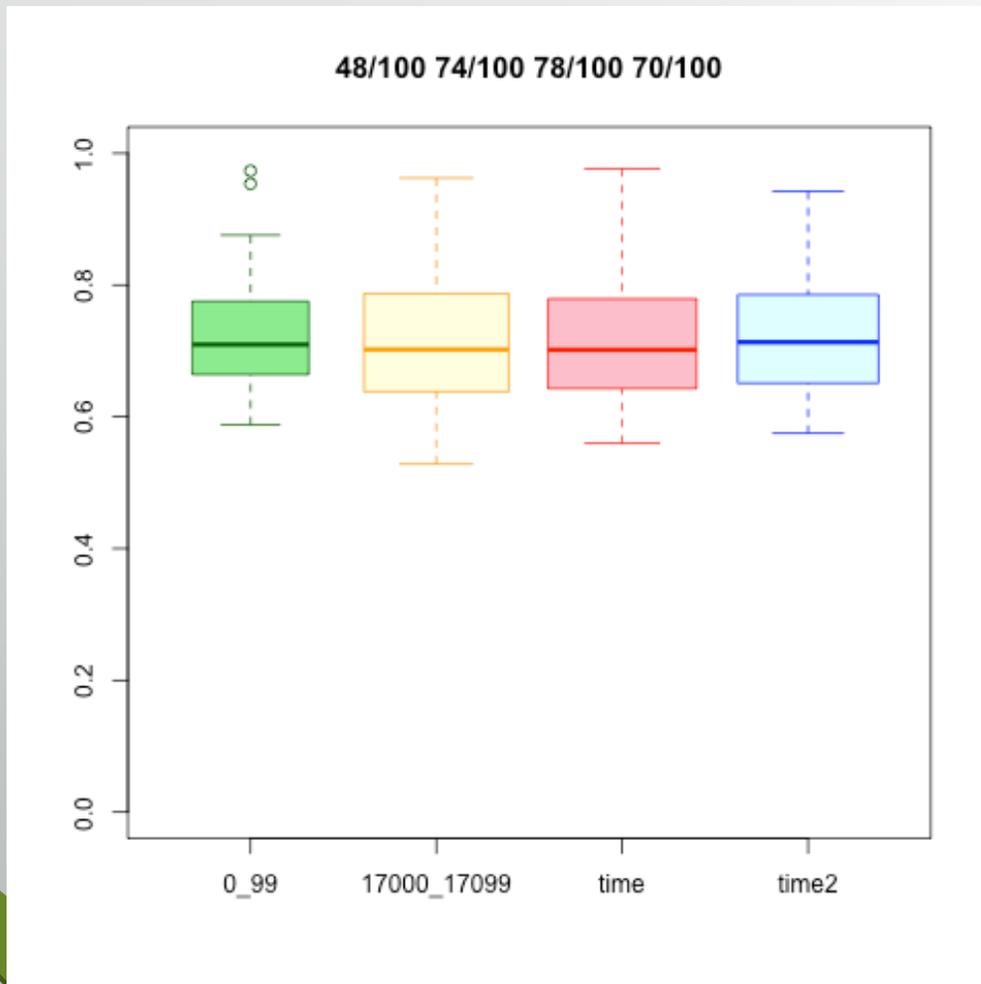


UCM(al)  $\gamma = 3.0$  半径

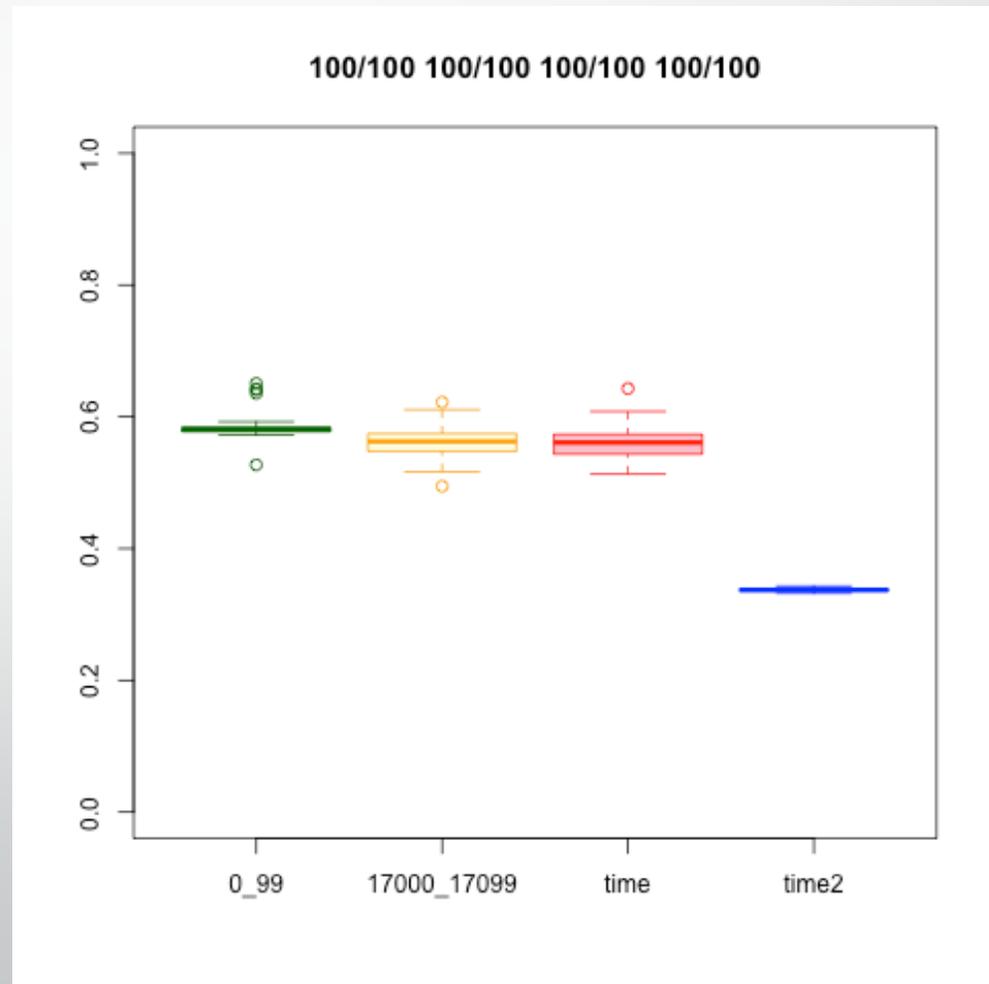


# 6. 実験結果

CM  $\gamma = 2.50$  近接

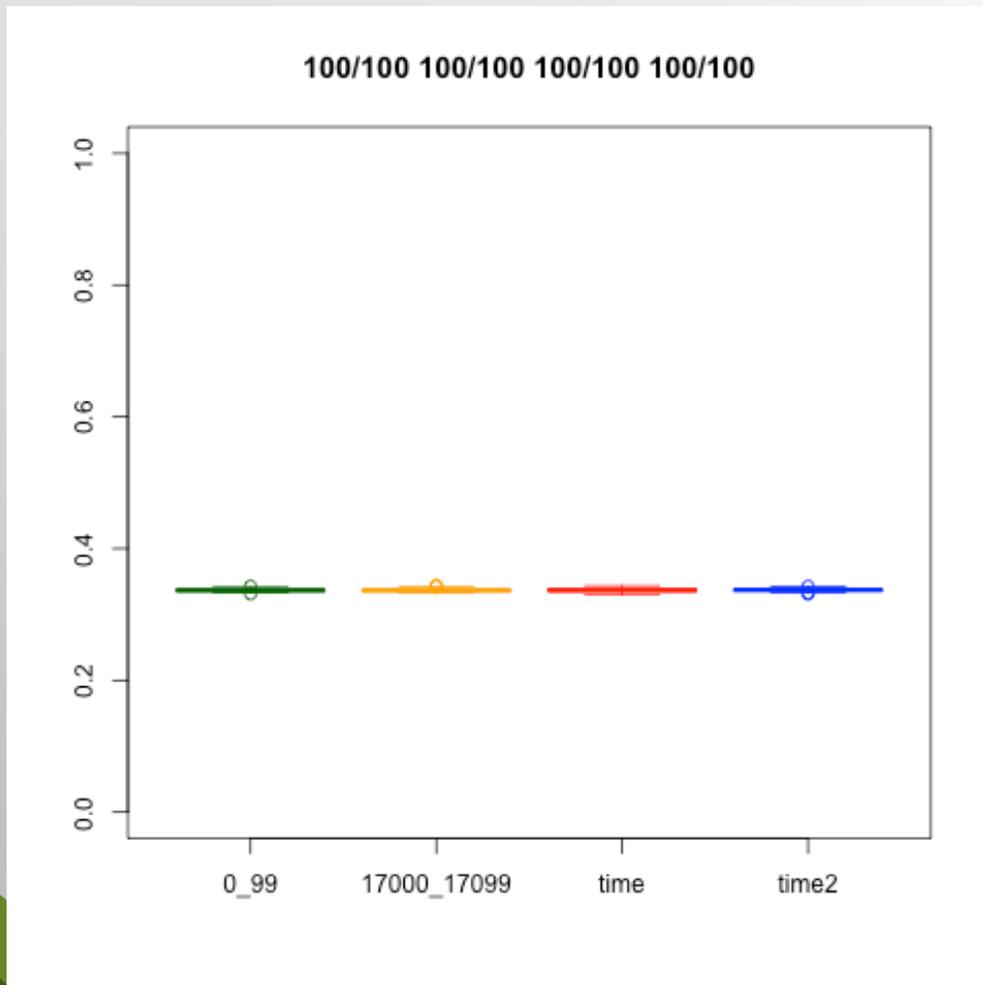


CM(al)  $\gamma = 2.50$  近接

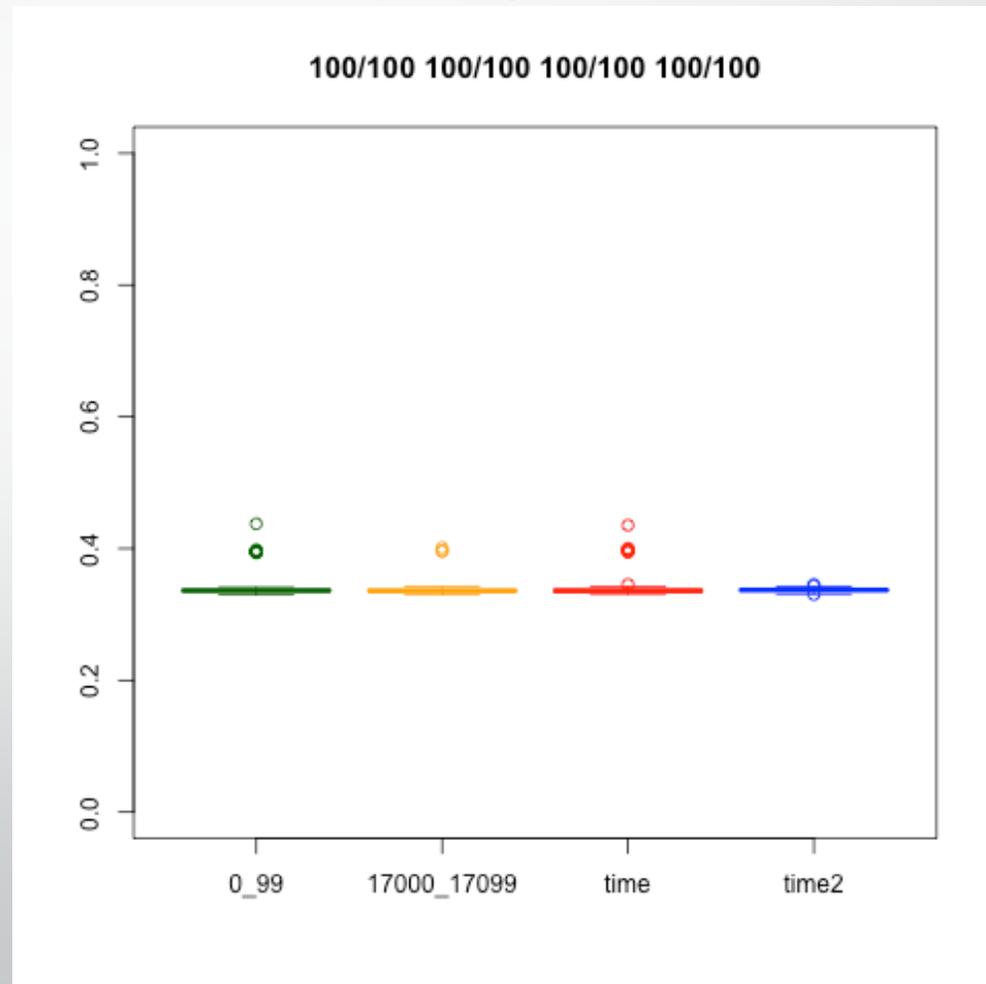


# 6. 実験結果

UCM  $\gamma = 2.50$  近接

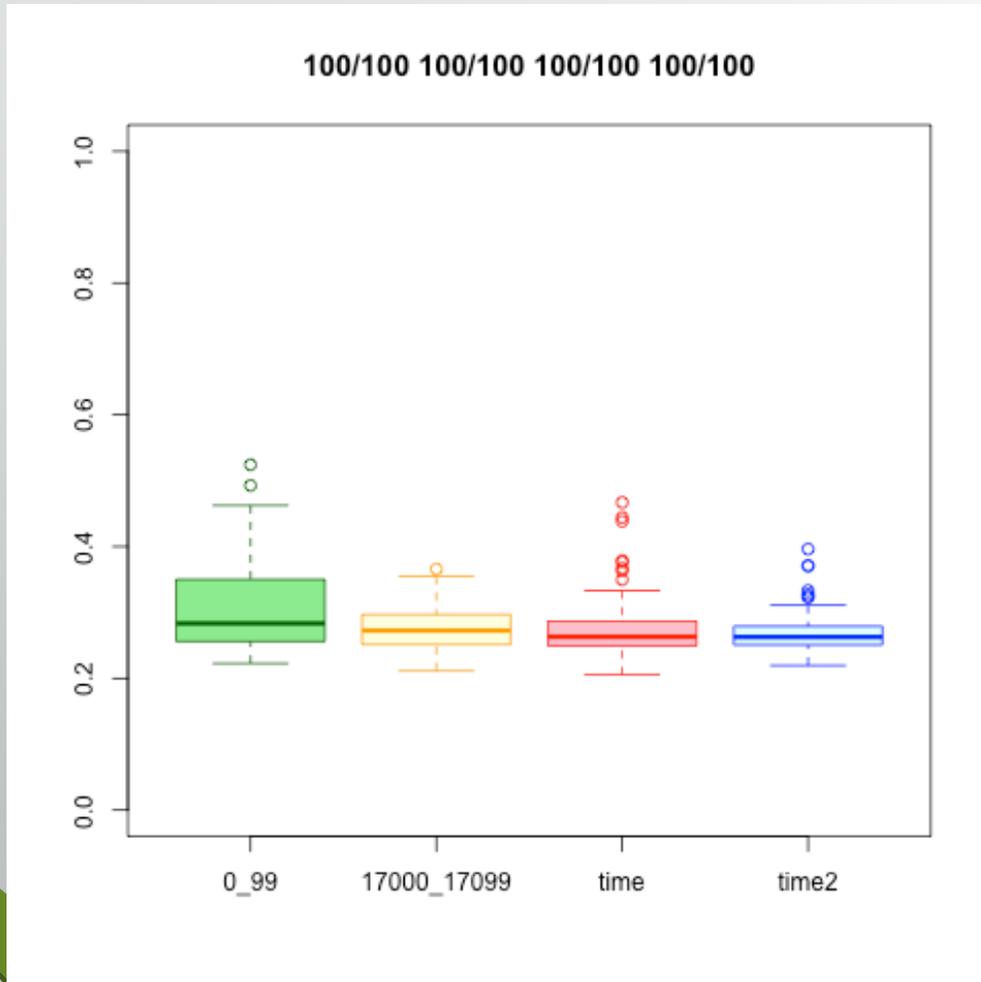


UCM(al)  $\gamma = 2.50$  近接

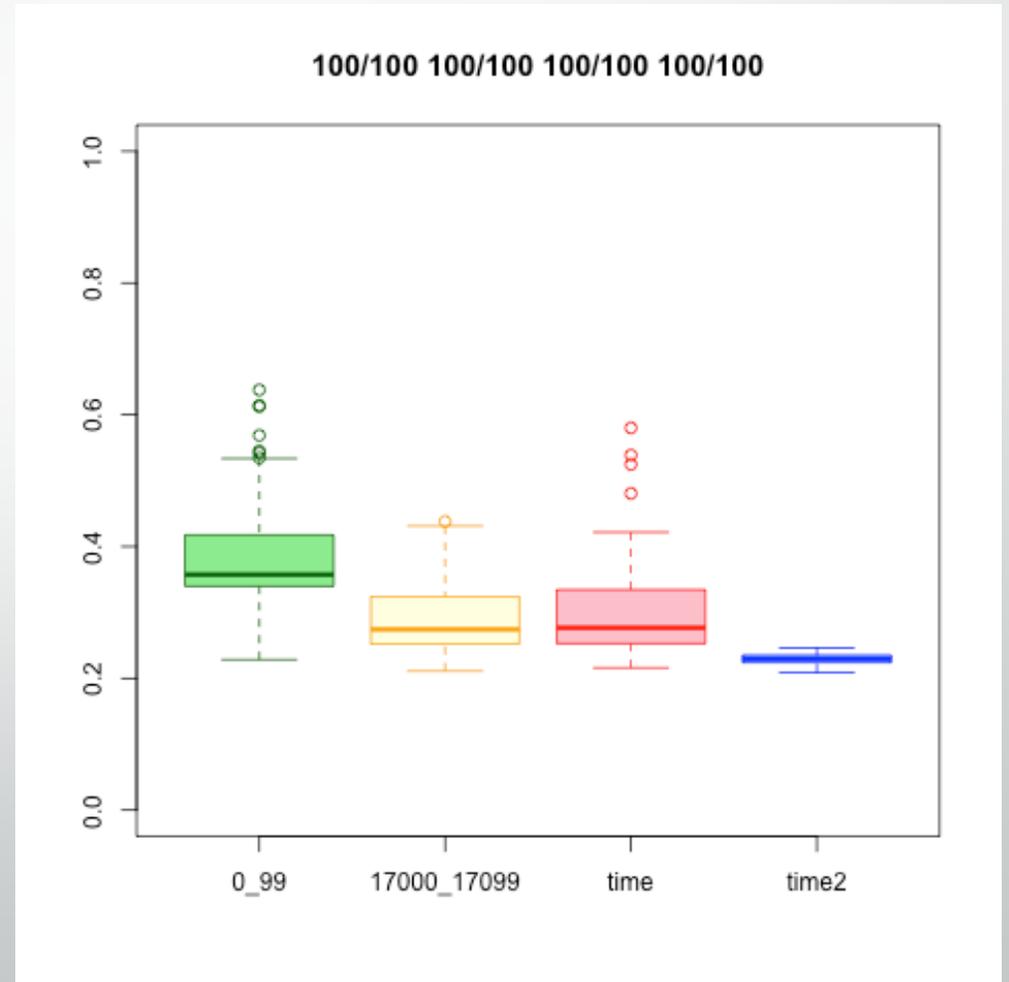


# 6. 実験結果

CM  $\gamma = 3.0$  近接

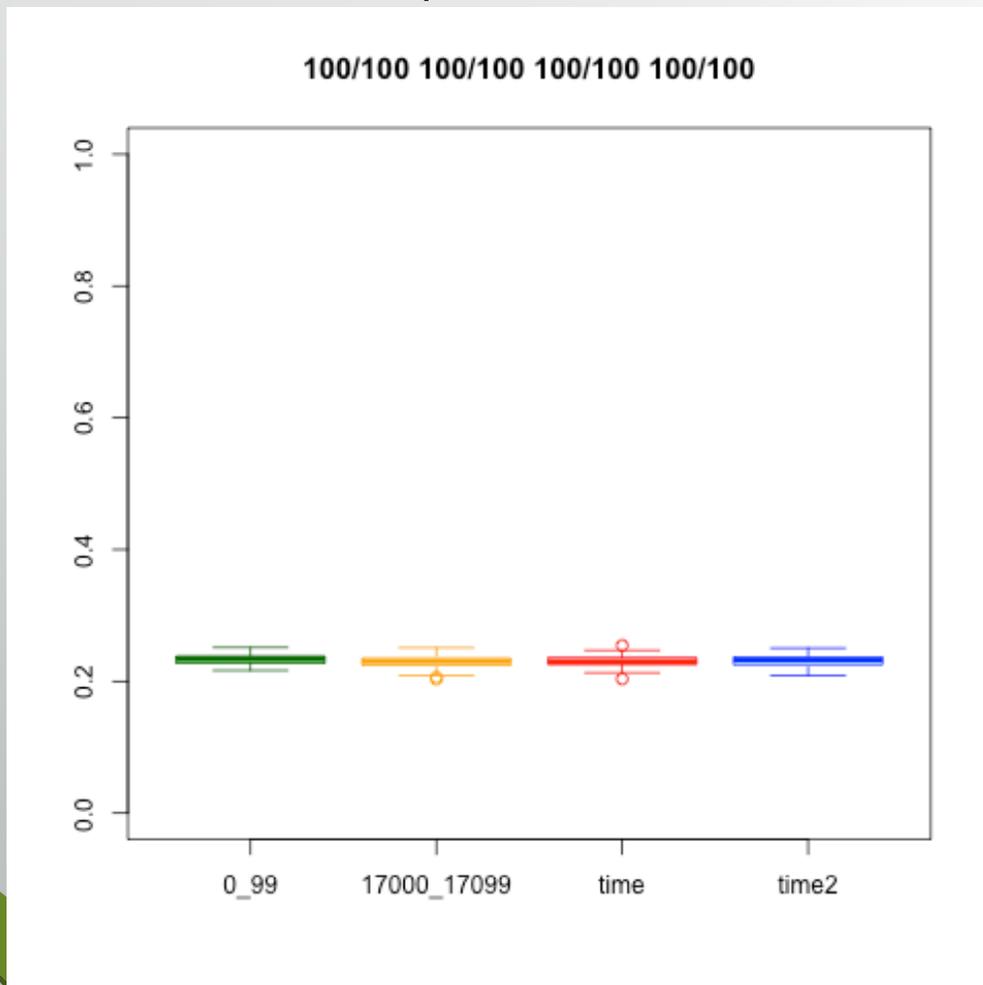


CM(al)  $\gamma = 3.0$  近接

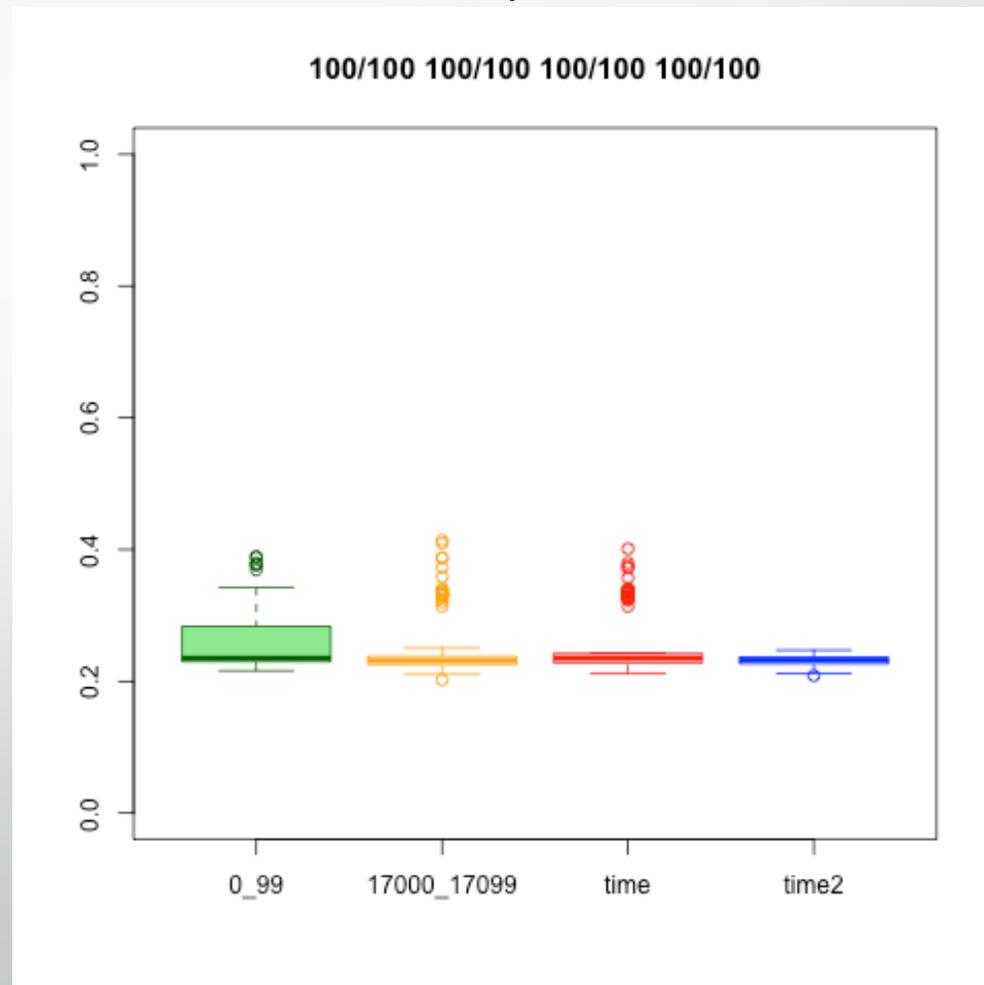


# 6. 実験結果

UCM  $\gamma = 3.0$  近接

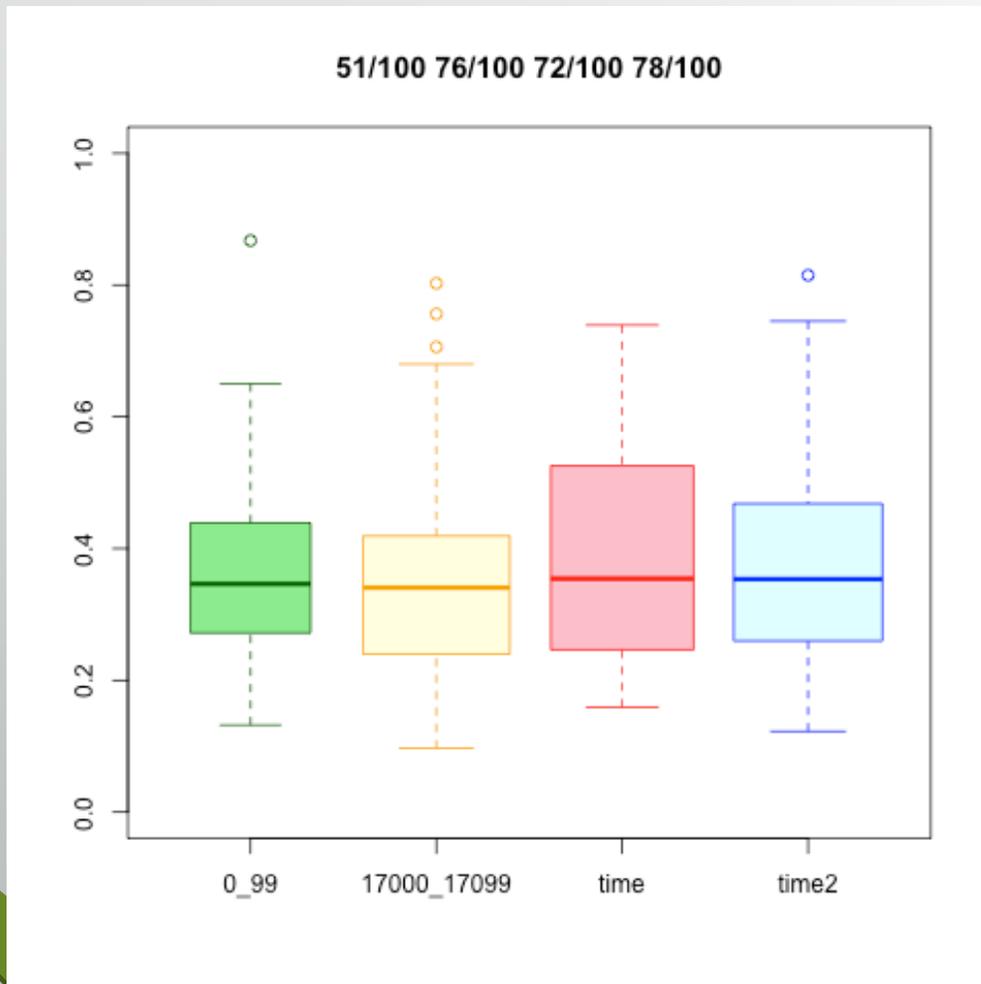


UCM(al)  $\gamma = 3.0$  近接

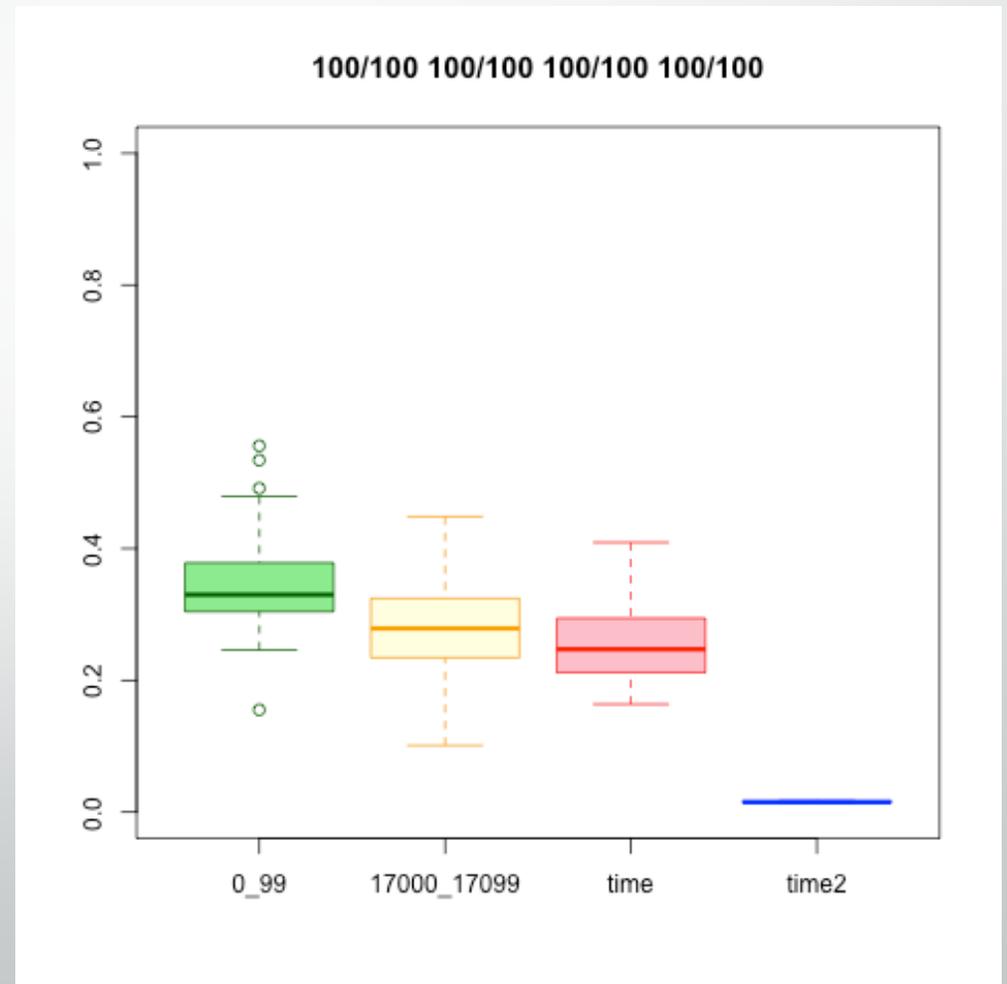


# 6. 実験結果

CM  $\gamma = 2.0$  媒介

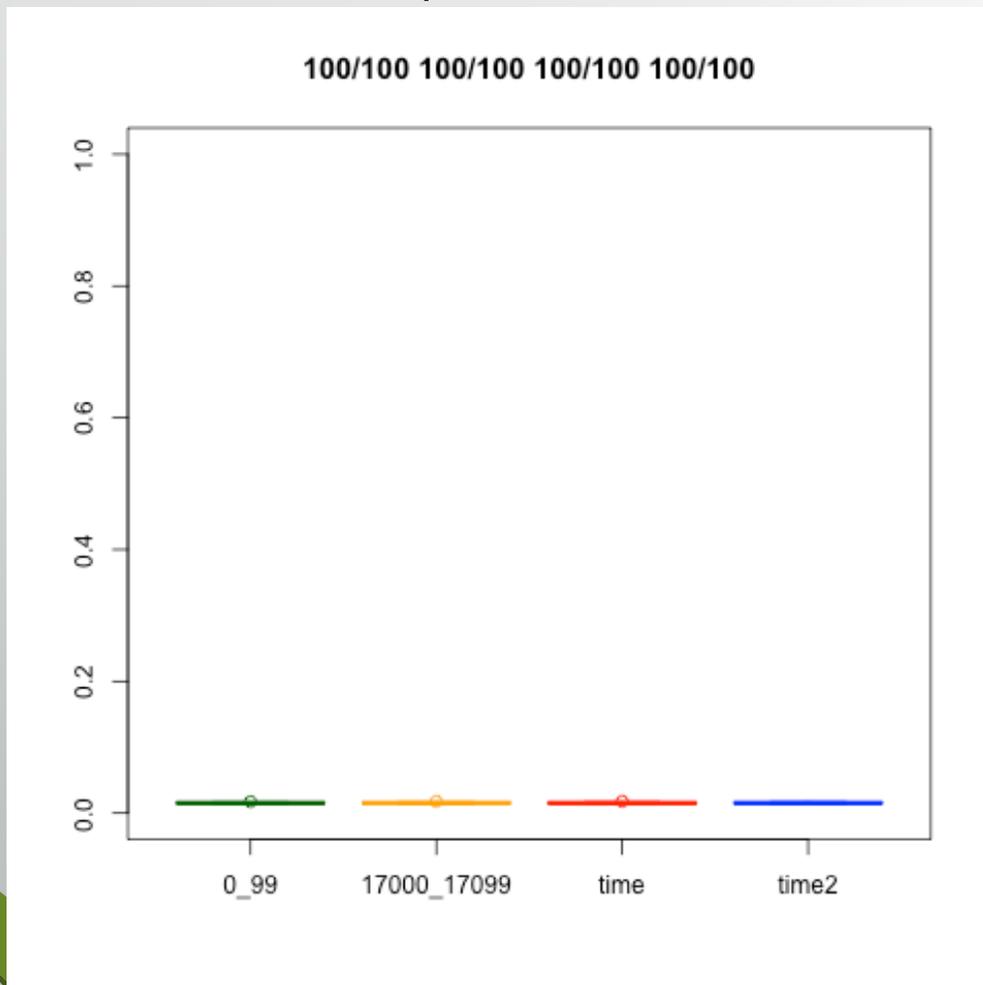


CM(al)  $\gamma = 2.0$  媒介

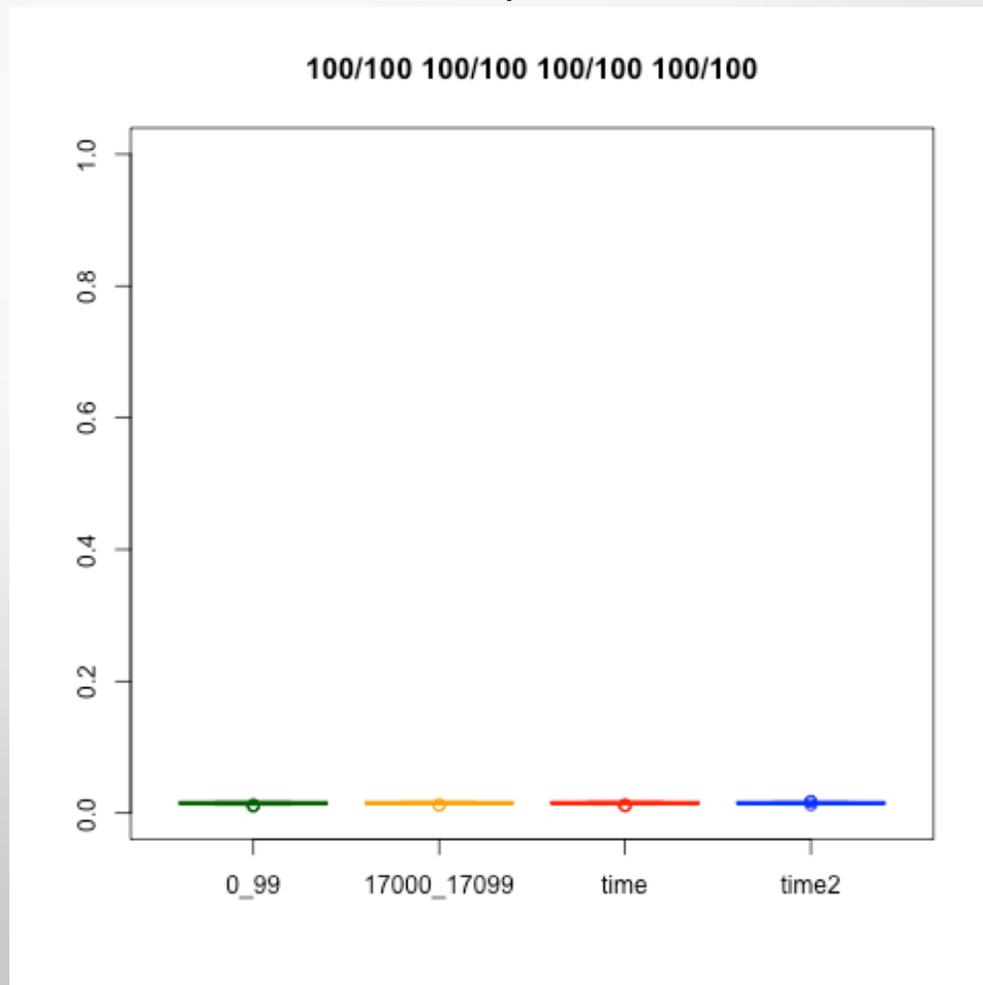


# 6. 実験結果

UCM  $\gamma = 2.0$  媒介

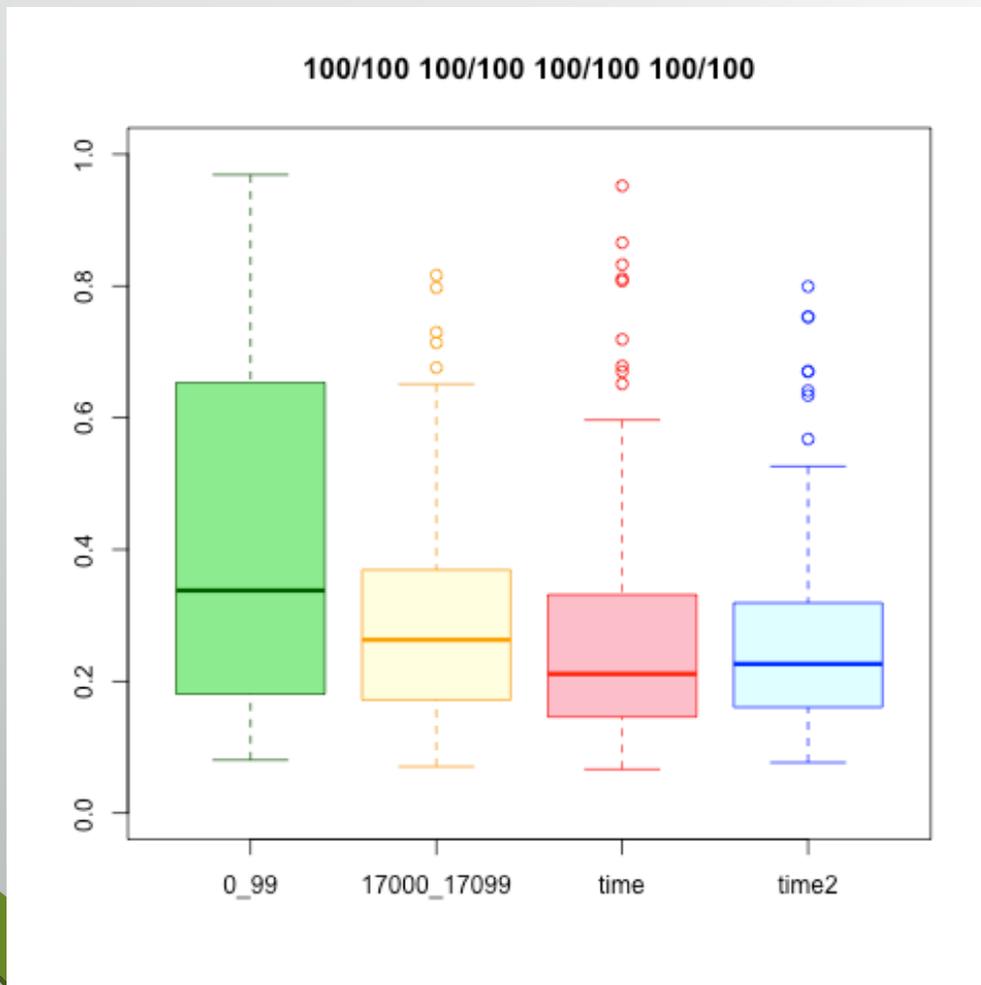


UCM(al)  $\gamma = 2.0$  媒介

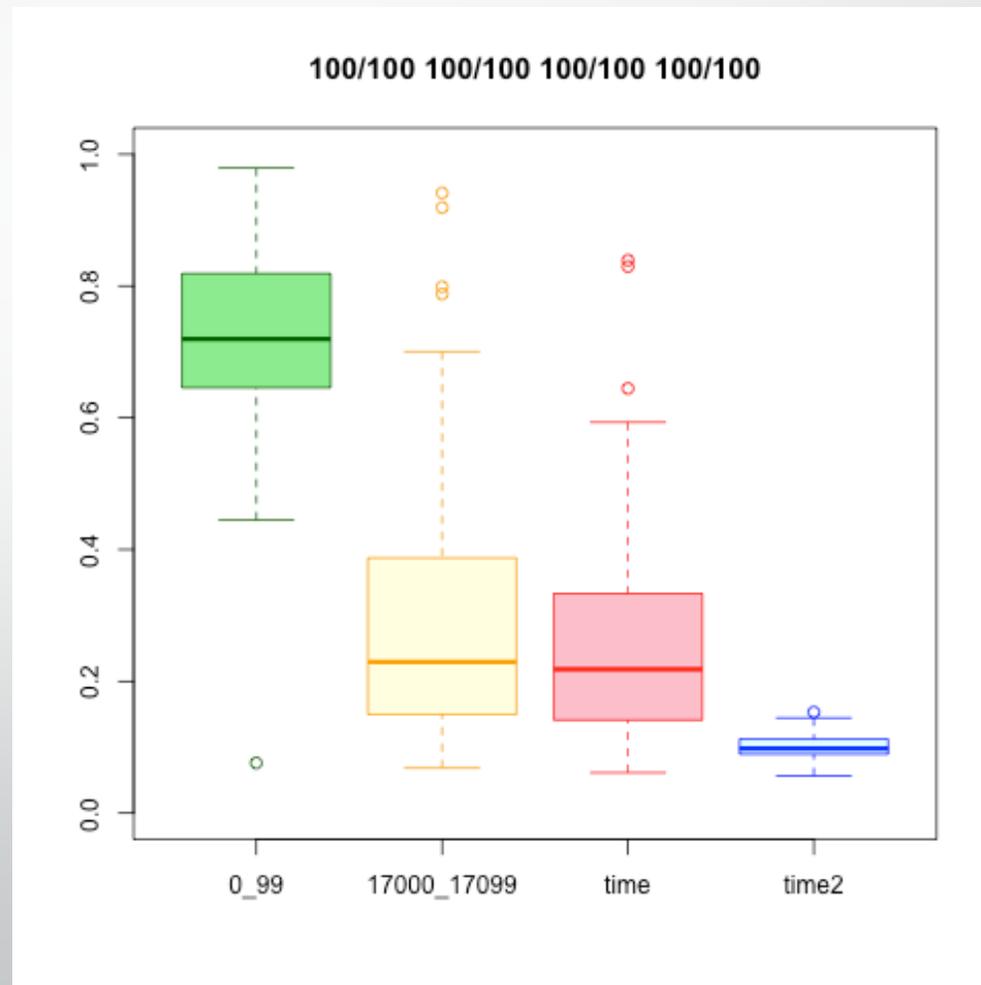


# 6. 実験結果

CM  $\gamma = 3.0$  媒介

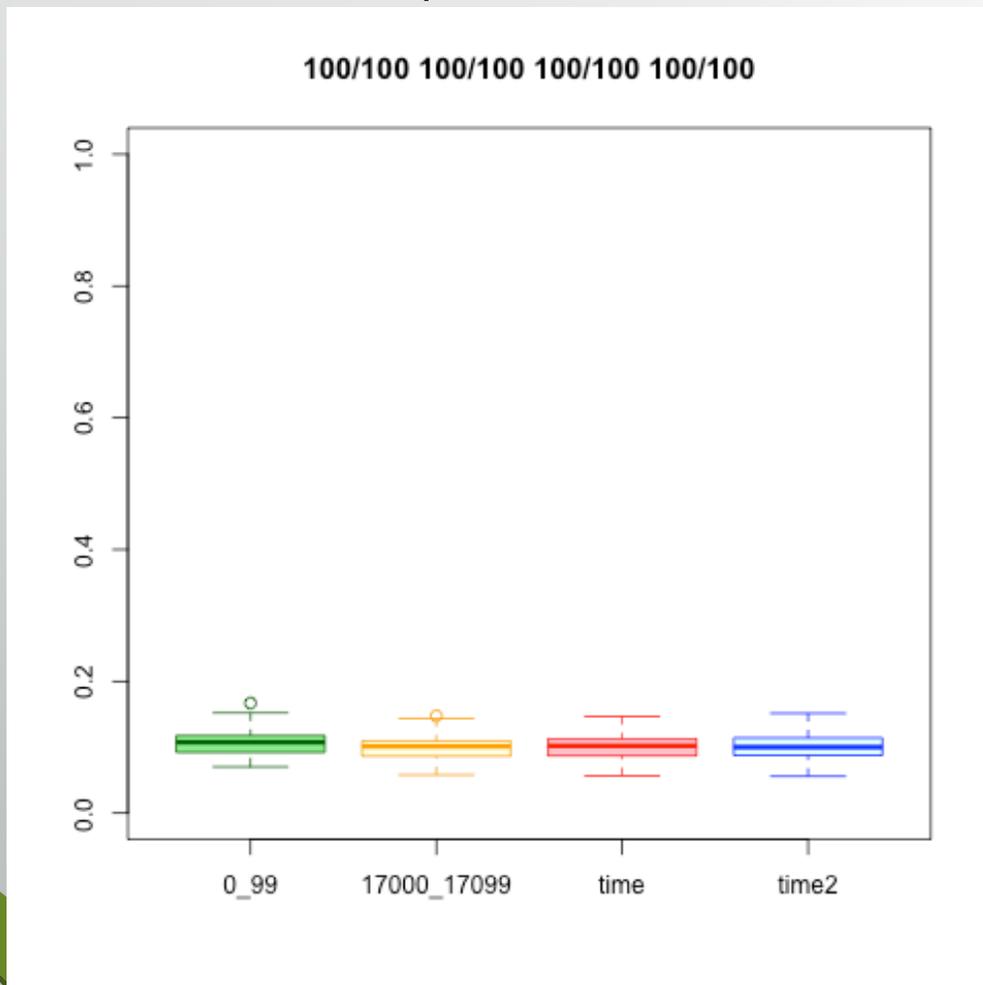


CM(al)  $\gamma = 3.0$  媒介

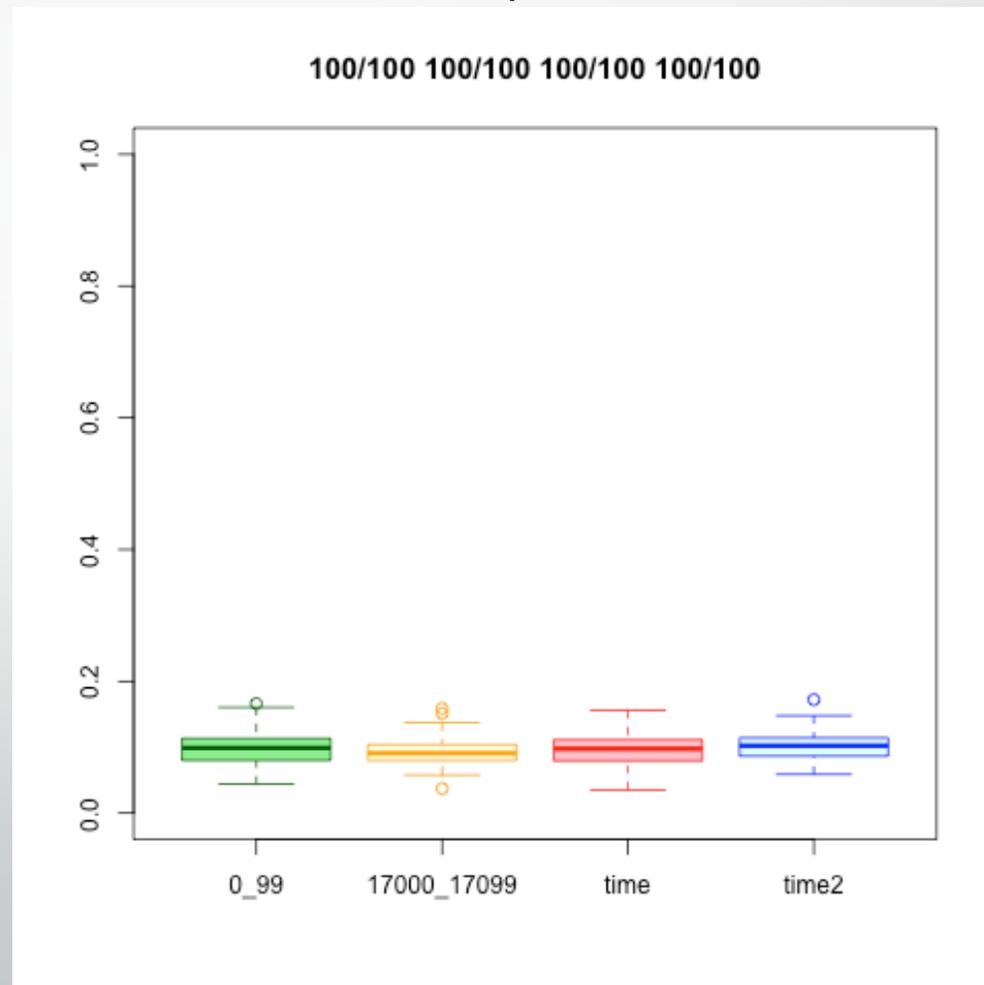


# 6. 実験結果

UCM  $\gamma = 3.0$  媒介



UCM(al)  $\gamma = 3.0$  媒介

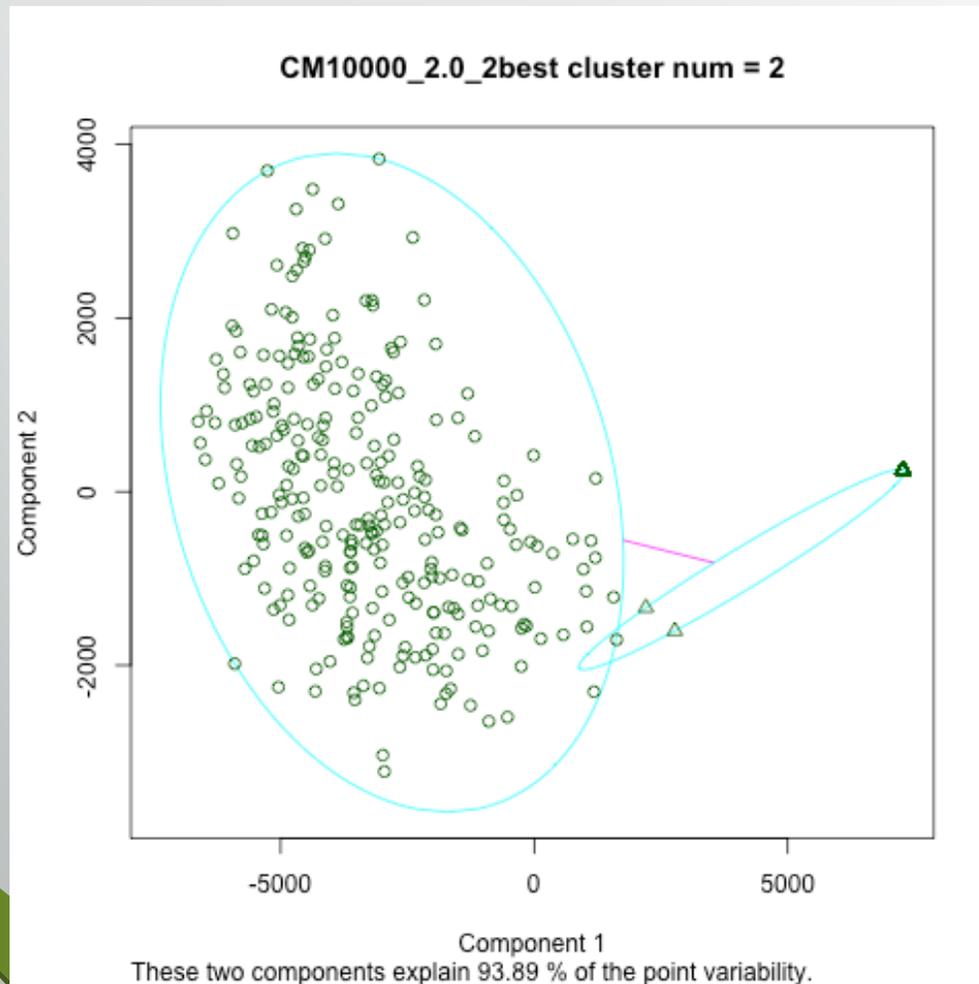


## 6. 実験結果

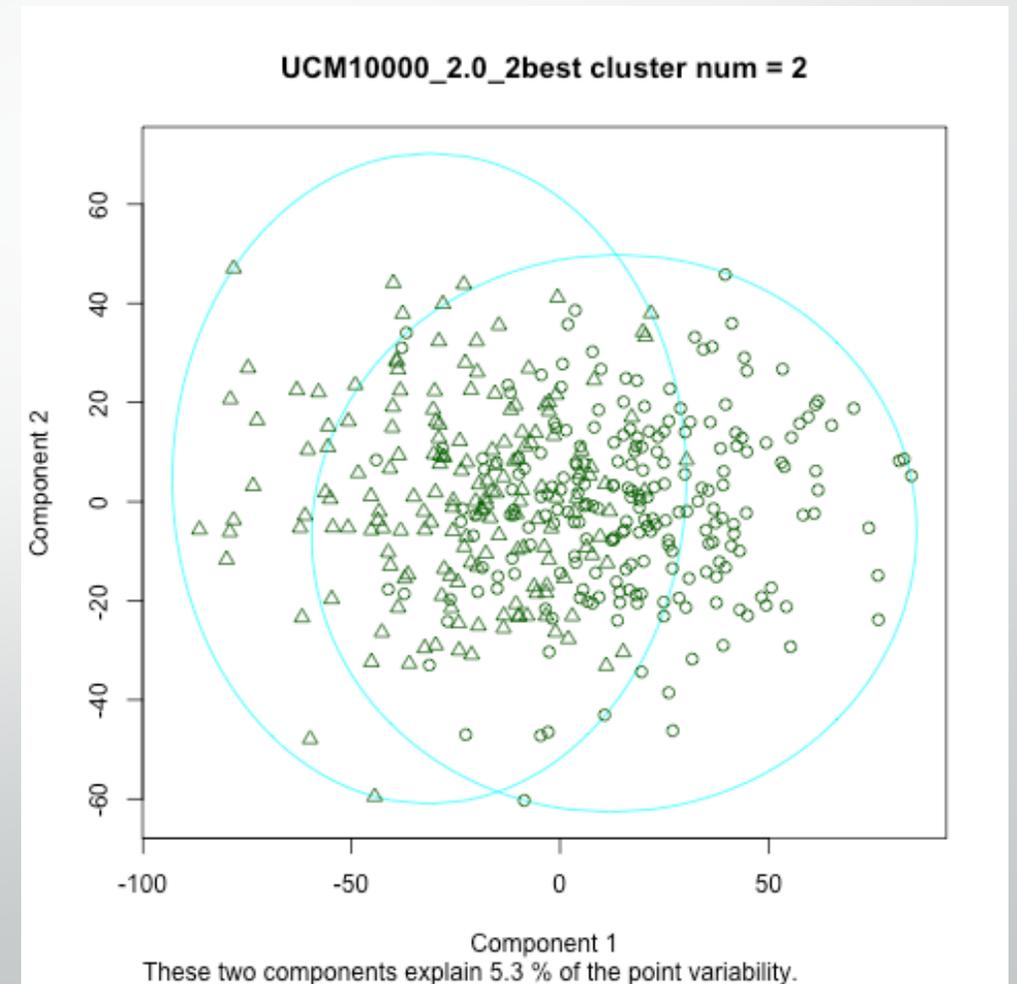
- $\gamma=2.0 \sim 2.75$  のとき  
seedによって特徴量に大きな差は見られなかった
- $\gamma = 3.0$  のとき  
 $\gamma$ が3.0未満の特徴量と取りうる値の範囲が大きく変化した  
seedによって特徴量に大きな差が見られた

# 6. 実験結果

CM  $\gamma = 2.0$  クラスタ数 = 2

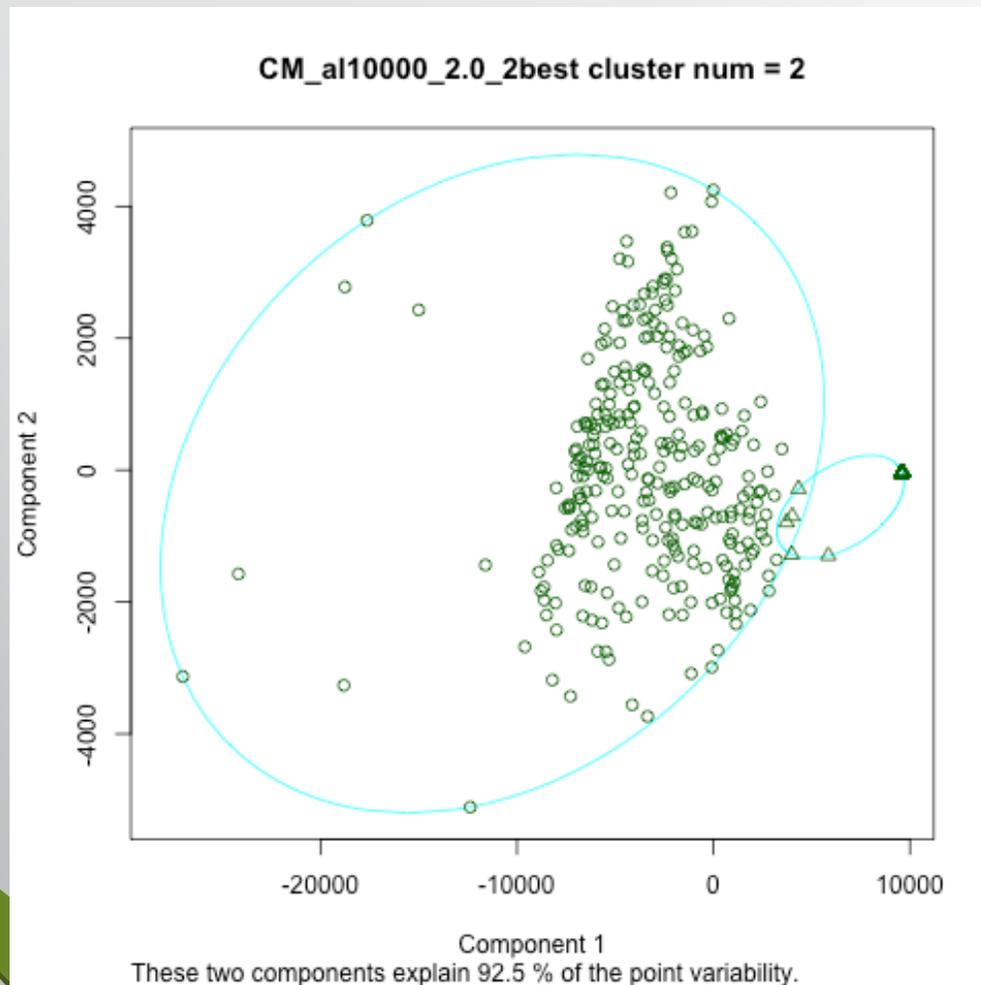


UCM  $\gamma = 2.0$  クラスタ数 = 2

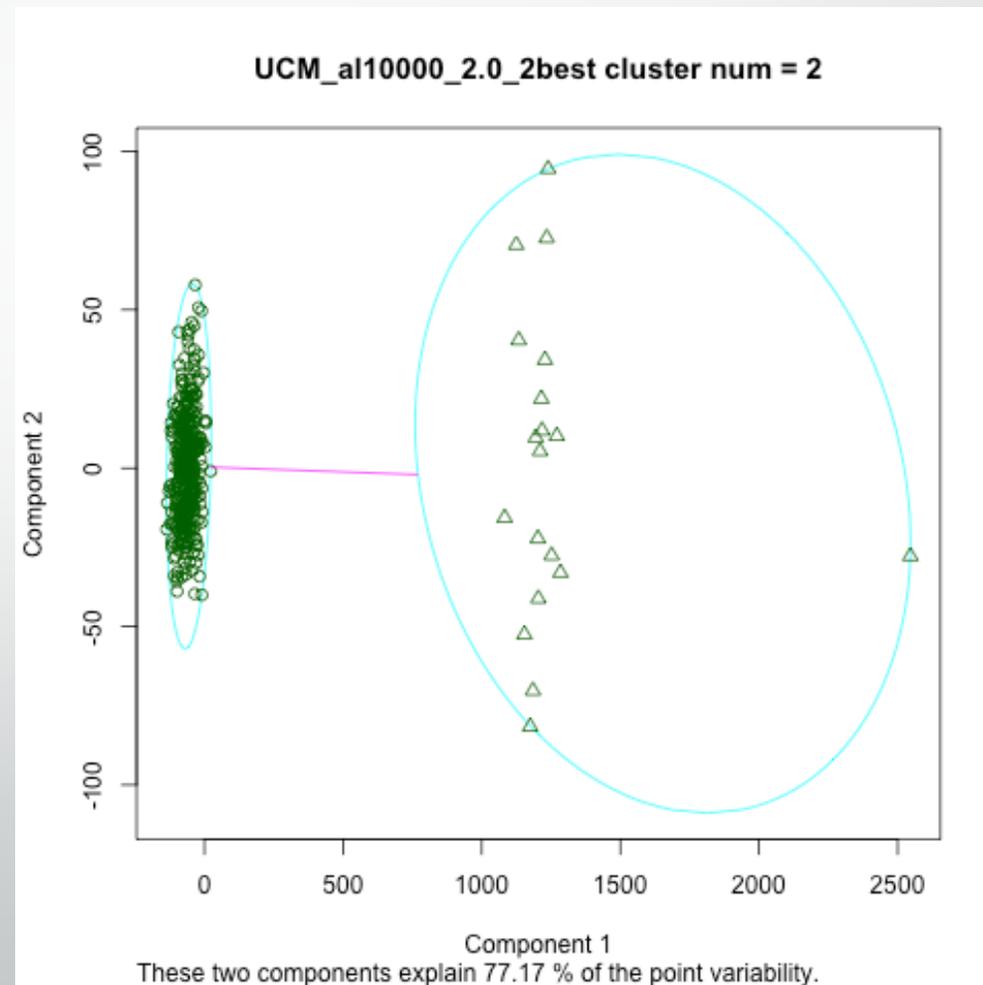


# 6. 実験結果

CM(a)  $\gamma = 2.0$  クラスタ数 = 2



UCM(a)  $\gamma = 2.0$  クラスタ数 = 2

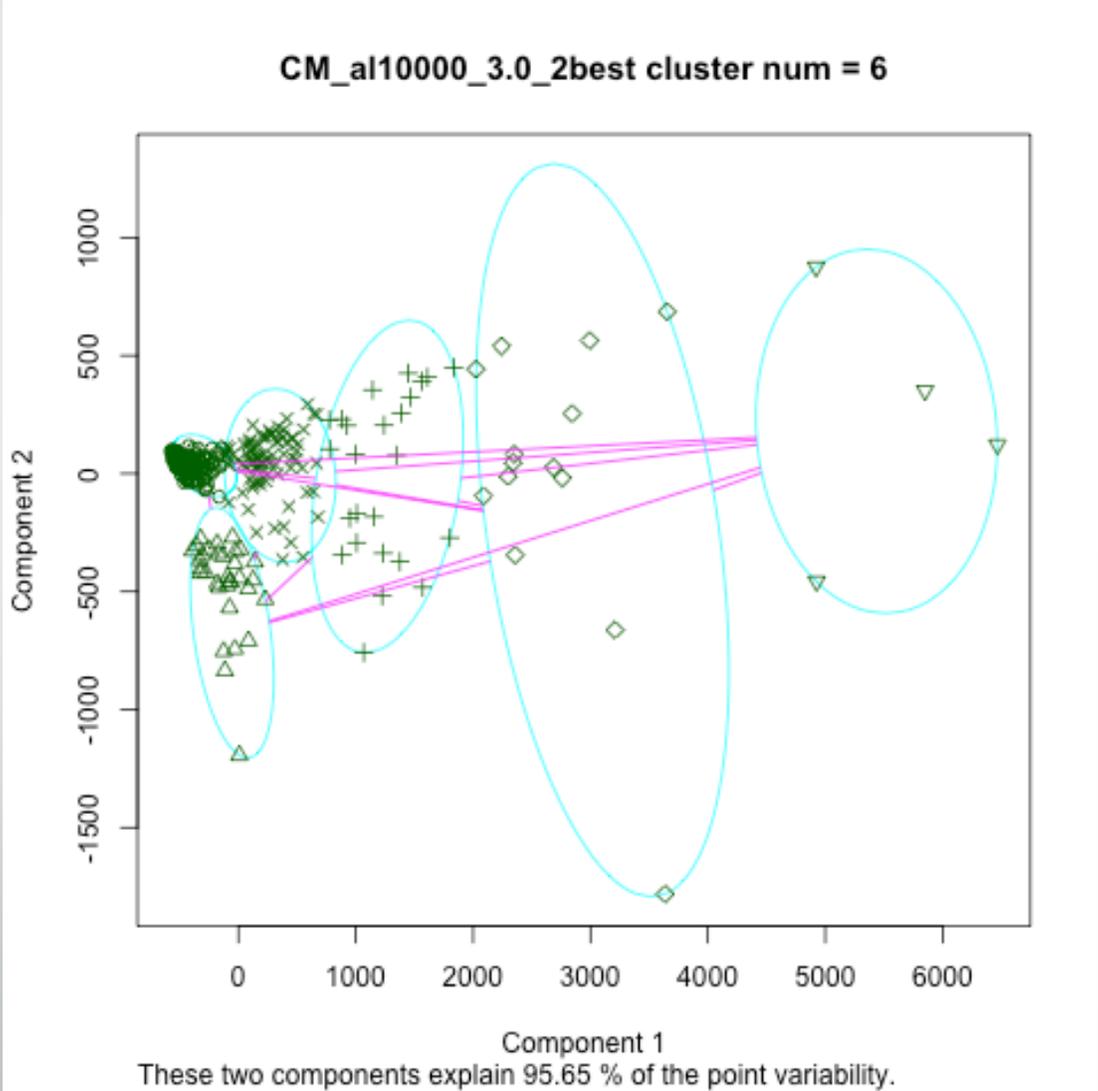


## 6. 実験結果

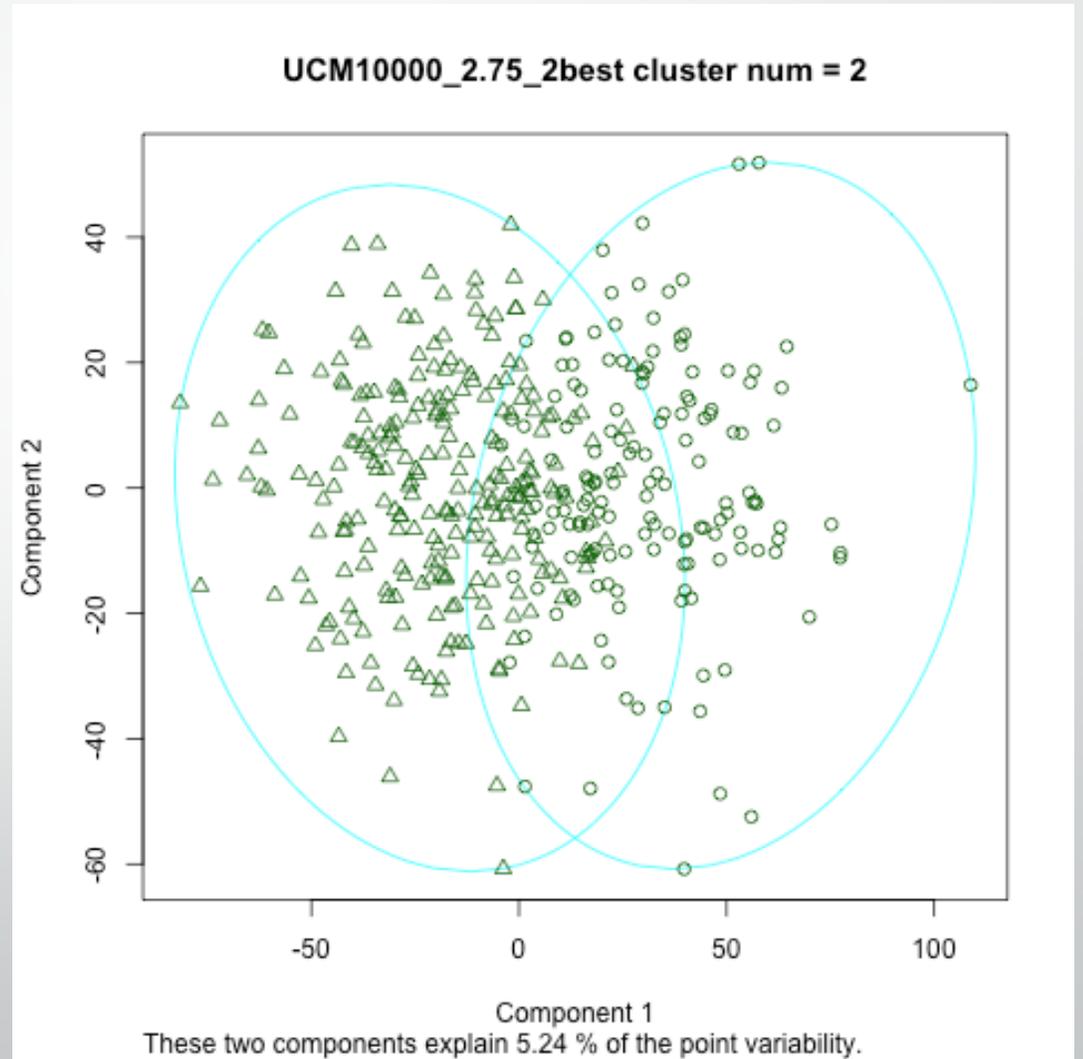
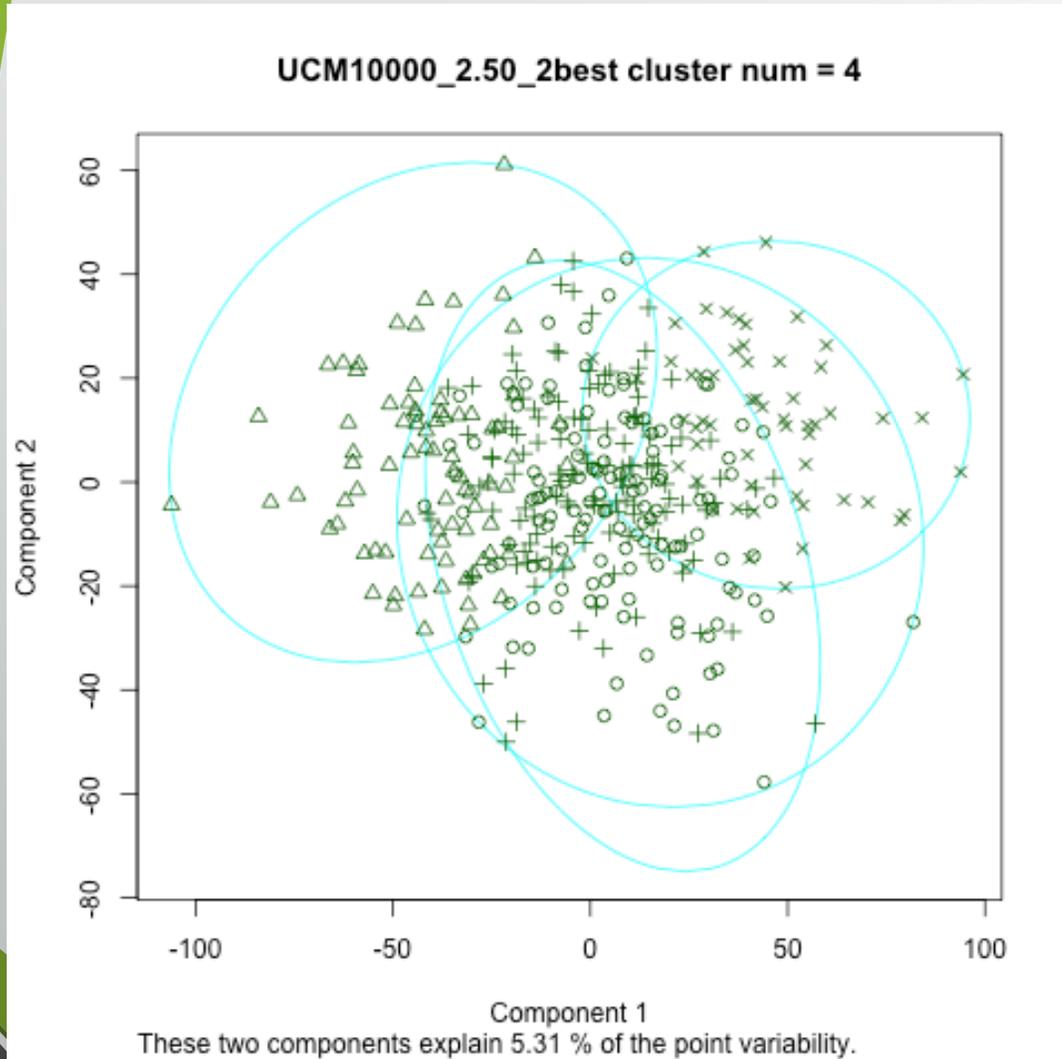
- K-medoid法によるクラスタリング結果

生成方法		べき指数	2.0	2.25	2.50	2.75	3.0
CM			2	2	2	2	2
	al		2	2	2	2	6
UCM			2	2	4	2	2
	al		2	2	2	2	2

# 6. 実験結果



# 6. 実験結果





## 7. 考察・今後の課題

# 7. 考察

- べき指数2.0~2.75の範囲では特徴量に大きな差が見られなかった  
→seedの取り方によるネットワークの特徴量への影響はほとんど無い
- べき指数3.0の特徴量に大きな差が見られた  
べき指数は3.0の前と後でネットワークの性質が大きく変わるということが知られている

# 7. 考察

- 次数平均  $\langle k \rangle$

$$\langle k \rangle = \int_{k_{\min}}^{\infty} k p(k) dk = \mathcal{N} \int_{k_{\min}}^{\infty} k^{-\gamma+1} dk = \frac{(\gamma-1)k_{\min}^{\gamma-1}}{2-\gamma} [k^{-\gamma+2}]_{k_{\min}}^{\infty}$$

$$\langle k \rangle = \begin{cases} \alpha & (\gamma > 2) \\ \infty & (\gamma \leq 2) \end{cases}$$

- 次数の二乗平均  $\langle k^2 \rangle$

$$\langle k^2 \rangle = \int_{k_{\min}}^{\infty} k^2 \times \mathcal{N} k^{-\gamma} dk = \frac{(\gamma-1)k_{\min}^{\gamma-1}}{3-\gamma} [k^{-\gamma+3}]_{k_{\min}}^{\infty}$$

$$\langle k^2 \rangle = \begin{cases} \alpha & (\gamma > 3) \\ \infty & (\gamma \leq 3) \end{cases}$$

# 7. 今後の課題

- ネットワークの性質が大きく変化する $\gamma = 3.0$ の前後のネットワーク解析  
→ 2.91, 2.92, ..., 2.99
- 今回生成したネットワークの伝搬シミュレーション実験