データ分析基礎
補足：グラフの三角形

京都大学 国際高等教育院

せきど ひろと
關戸 啓人

sekido.hiroto.7a@kyoto-u.ac.jp

グラフの三角形の数

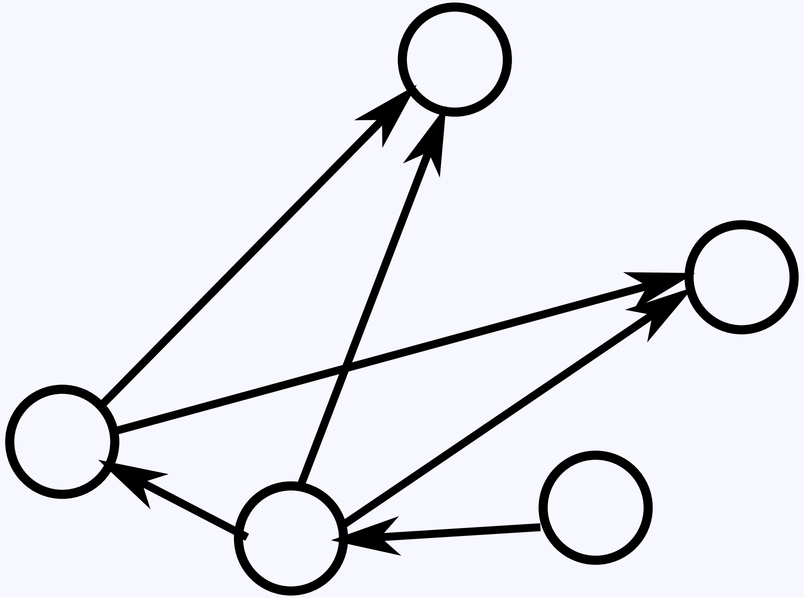
準備：グラフとは

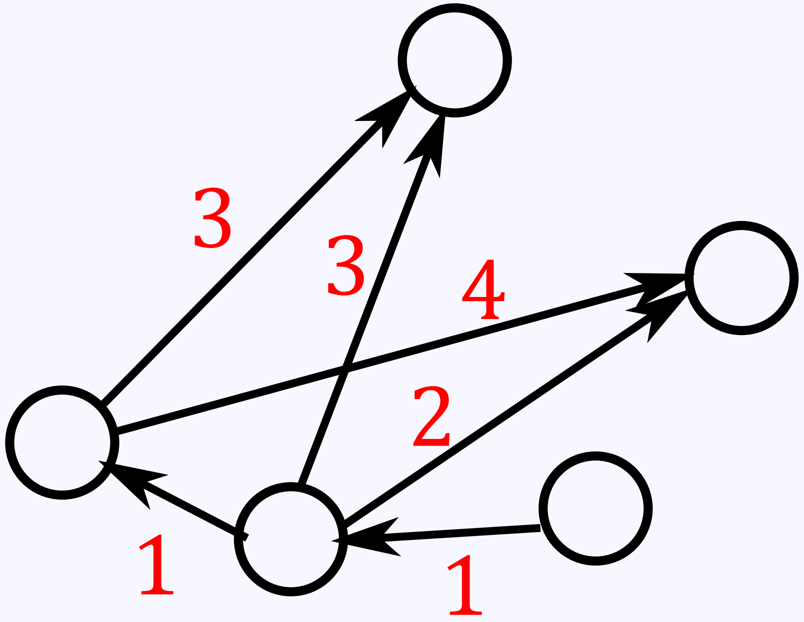
★ 節点（node）と枝（edge）からなるもの

- ★ 枝は2つの節点を結ぶ線（無向グラフ），または，矢印（有向グラフ）
- ★ 各節点，または，各枝には重みと呼ばれる実数が与えられている場合もある

★ 節点uからvへと向かう枝（有向グラフ），および，接点uとvを結ぶ枝（無向グラフ）を(u,v)と書く

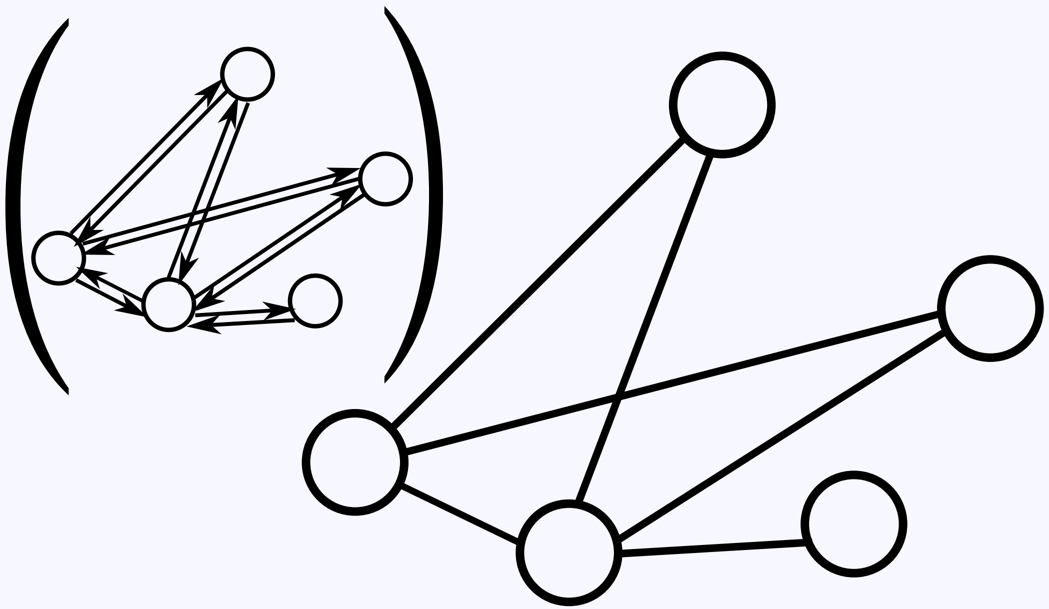
例：有向グラフ

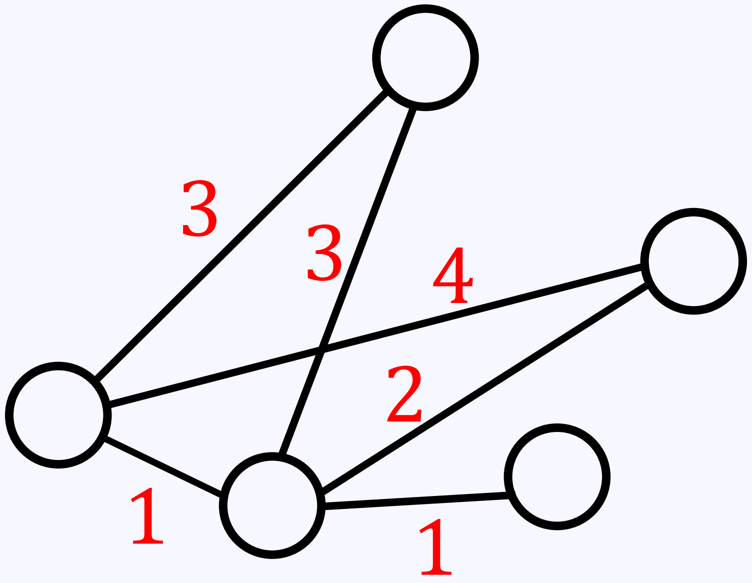




★ 例えば，Twitter のフォロー・フォロワー関係

例：無向グラフ





★ 例えば，Facebook の友達関係

準備：無向グラフに対する隣接行列

- ★ 無向グラフにおいて，節点数を N とする
- ★ このとき，N 行 N 列の行列を以下のよって定める
  - ★ 節点の番号を $1, 2, \ldots, N$ とする
  - ★ 第 (i, j) 要素は，節点 i と節点 j を結ぶ枝があれば 1，なければ 0 とする
  - ★ 節点 i と節点 j を結ぶ枝が 2 本以上ある場合，第 (i, j) 要素を枝の本数とすることもある
  - ★ それぞれの枝に実数が与えられている場合は，1 の代わりにその実数とする．

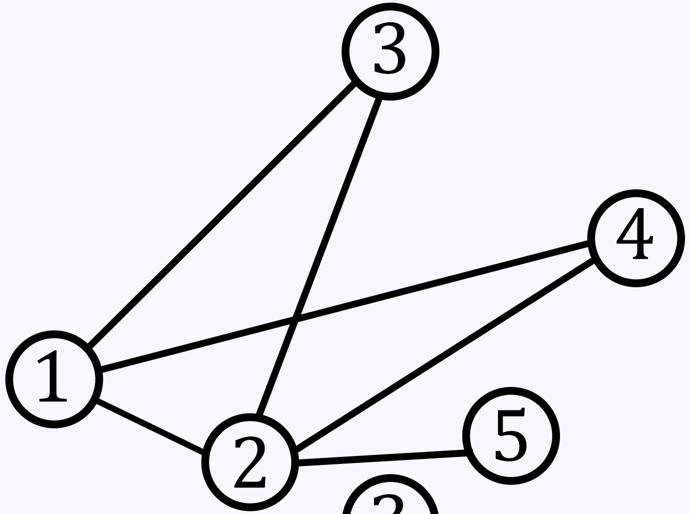
準備：有向グラフに対する隣接行列

★ 有向グラフにおいて，任意の節点の組 (u, v) に対して，u から v に向かう枝が高々1本しかないとする（2本以上ある場合の定義は色々）

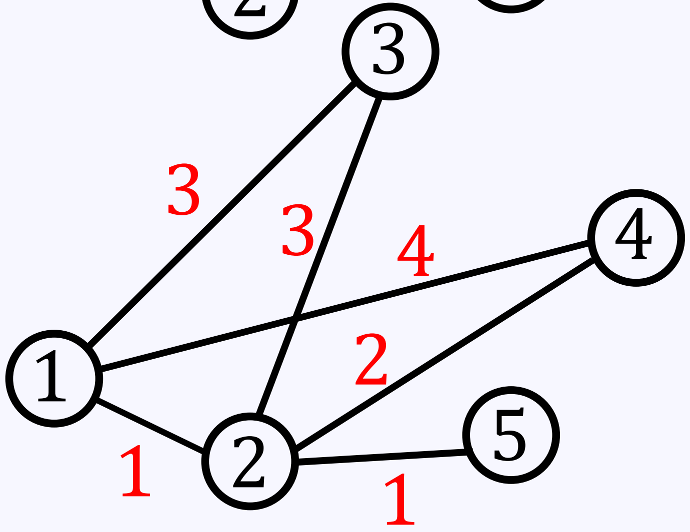
★ 節点数を N としたとき，N 行 N 列の行列を以下のよって定める

- ★ 節点の番号を $1, 2, \ldots, N$ とする
- ★ 第 (i, j) 要素は，節点 i から節点 j に向かう枝があれば1，なければ0とする
- ★ それぞれの枝に実数が与えられている場合は，1の代わりにその実数とする．

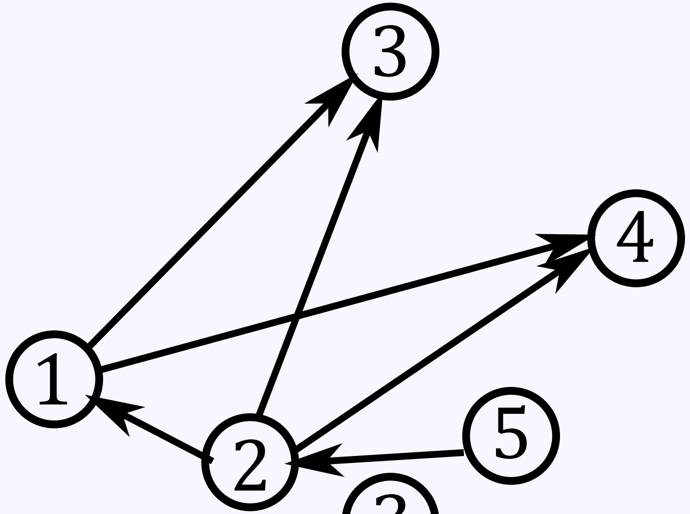
例：隣接行列



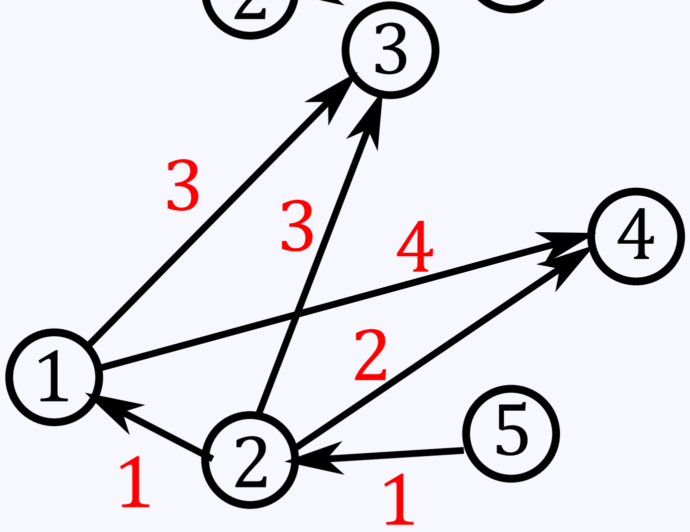
$$A = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$



$$A = \begin{pmatrix} 0 & 1 & 3 & 4 & 0 \\ 1 & 0 & 3 & 2 & 1 \\ 3 & 3 & 0 & 0 & 0 \\ 4 & 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$



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



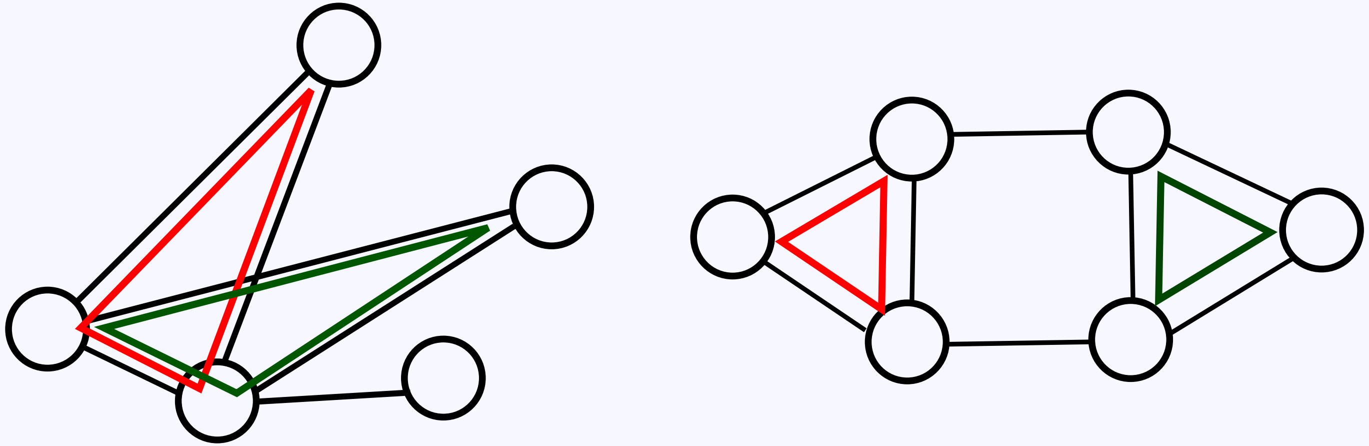
$$A = \begin{pmatrix} 0 & 0 & 3 & 4 & 0 \\ 1 & 0 & 3 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

グラフと隣接行列

- ★ グラフを考えることと，隣接行列を考えることは同値
  - ★ グラフの形で考えた方が直感的にわかりやすい
  - ★ 行列は数学的に非常によく知られた対象である
  - ★ どっちが適しているかはケースバイケース
- ★ 隣接行列を行列として考えなければ，考えつかないようなアルゴリズムとして，まず，三角形の数の近似計算を紹介する

グラフの三角形の数

- ★ 単純な無向グラフを考える
  - ★ 自分自身を結ぶ枝はない（両端点が同じ枝がない，ループ）
  - ★ 2つの節点の間には高々1本の枝しか無い
- ★ グラフの三角形の数とは枝 $(u,v),(v,w),(w,u)$ と3つの節点の間にそれぞれ枝があるような (u,v,w) の組の数である



グラフの三角形の数の応用

- 各節点iが含まれる三角形の数は重要な特徴量
- 節点iの隣接節点間に枝がある割合をクラスタ係数という
  - クラスタ係数が小さい
    - お客さんとの付き合いが多いなど
    - 無差別に友達申請しているのでは？
  - クラスタ係数が大きい
    - 共通の趣味など，共通の属性を持っている人たちが集まっているなど
    - 複数のアカウントを取得して，自分にリンクを張っているのでは？
- 異常なユーザーを検出できる
- 小さなグラフなら簡単に計算できるが，グラフがとても大きくなると大変

隣接行列 A の性質

- ★ A^n の第 (i,j) 成分は，節点 i から節点 j に n ステップで移動する方法の数
  - ★ 例えば A^2 の第 (i,j) 成分は

$$A_{i,1}A_{1,j} + A_{i,2}A_{2,j} + \ldots + A_{i,N}A_{N,j}$$

    で，第 k 項 $(i,k),(k,j)$ があるときのみ 1 で，$i \to k \to j$ と移動することを表す

- ★ グラフの三角形の数は $\mathrm{tr}(A^3)/6 = ((A^3)_{1,1} + (A^3)_{2,2} + \cdots + (A^3)_{N,N})/6$
  - ★ $\mathrm{tr}(A^3)$ は，ある頂点を出発して 3 ステップで元に戻る場合の数
  - ★ 各三角形について，どこが出発点か，時計回りかその逆か，で 6 パターンある

節点 i を含む三角形の数

- ★ 節点 i を含む三角形の数を調べたい
  - ★ $A = U\Lambda U^{\mathrm{T}}$ と固有値分解されたとする $(\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_N))$
- ★ グラフの三角形の総数は

$$\mathrm{tr}(A^3)/6 = \frac{1}{6}\sum_{k=1}^{N}\lambda_k^3$$

- ★ 節点 i を含む三角形の数は（時計回りと反時計回りがあるので /2）

$$\frac{1}{2}(A^3)_{i,i} = \frac{1}{2}(U\Lambda^3 U^{\mathrm{T}})_{i,i} = \frac{1}{2}\sum_{k=1}^{N}\lambda_k^3 U_{i,k}^2$$

- ★ **大きい固有値（とその固有ベクトル）のみを求め，小さい固有値を無視することで高速化**