사회 연결망 분석 기초

서울대학교 데이터 사이언스 부트캠프

Gyuho Lee, hci+d lab., Department of Communication, Seoul National University

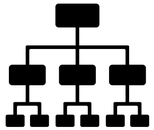
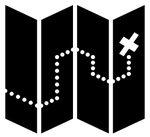
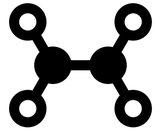
Introduction

Gyuho Lee

- 서울대학교 언론정보학과 hci+d lab. 박사과정
- Computational Social Science (Community / Journalism)
  - 자연어 처리
  - 사회 연결망
  - 데이터 시각화
- Python, R, Tableau, SQL(Database) ... 아마도 재미있어 보이는 모든 것

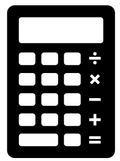
연결망 분석?

- 특별한 분야가 아니라 모든 사회 현상에 적용 가능한 접근 방식
- 대상(entities)과 연결(relation/interaction)이 있다면, 그래프 구조로 분석될 수 있다
  - 오일러의 정리부터 모레노의 소시오그램까지 다양한 학문 분야에서 꾸준히 연구


Graph부터 Network까지

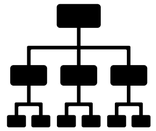
Graph



추상화

구체화

Network



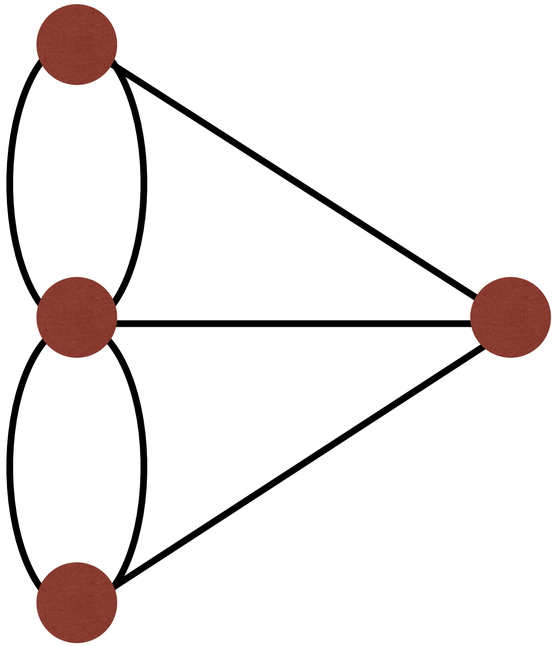
Network Science

Social Network Analysis

Network Analysis

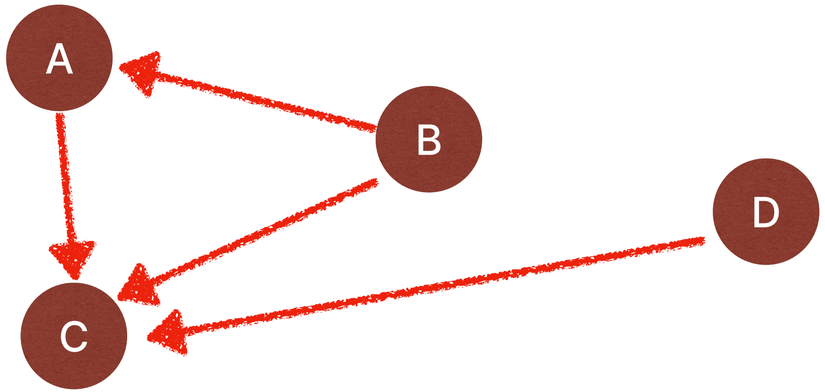
쾨니히스베르크의 7다리 건너기

- 쾨니히스베르크시의 프레겔 강을 건너는 일곱 개의 다리를 한번만 사용하여 모든 지점의 이동이 가능할까?
  - 여러 방법이 시도된 이후, 오일러의 정리로 방법이 없다는 것이 확인 (홀수, 짝수 점)



모레노의 사회연결망 연구

- 학생들의 심리/생활을 파악하는 과정에서 친교 관계를 연결망으로 해석
  - 연결 패턴이 학생들에게 미치는 영향/그룹을 연구
  - 1930년대 뉴욕 타임즈에 연결망에 기반한 분석에 대한 의견 제시
  - Moreno, J. L. (1934). Who shall survive?: A new approach to the problem of human interrelations.

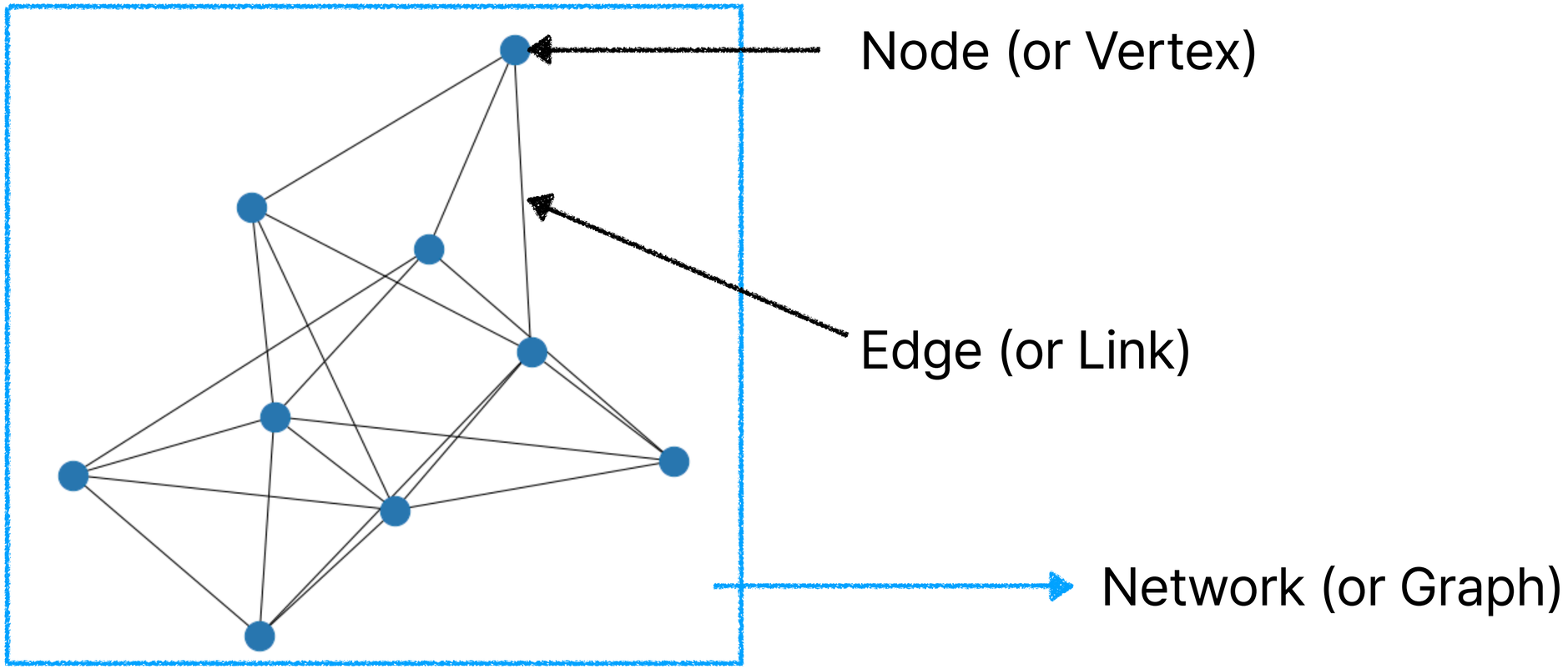


What will we learn?

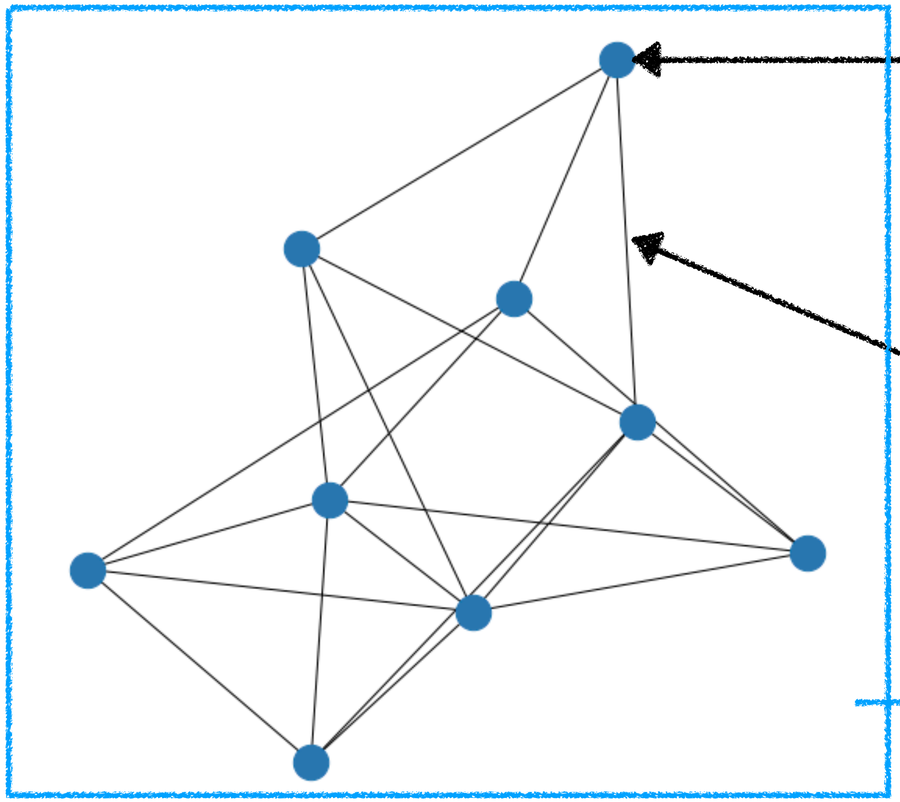
- 기초적 이해
  - 사회 연결망 분석에서 쓰이는 기본 용어
  - 사회 연결망 분석에서 쓰이는 보편적 방법론
- 실습 (오늘 수업의 핵심!)
  - Networkx을 중심으로 배운 내용 구현하기
- 응용 (시간이 된다면?)
  - 사회 연결망 분석 사례/논의

Graph? Network?

기초 구조



동일한 구조의 다양한 변형

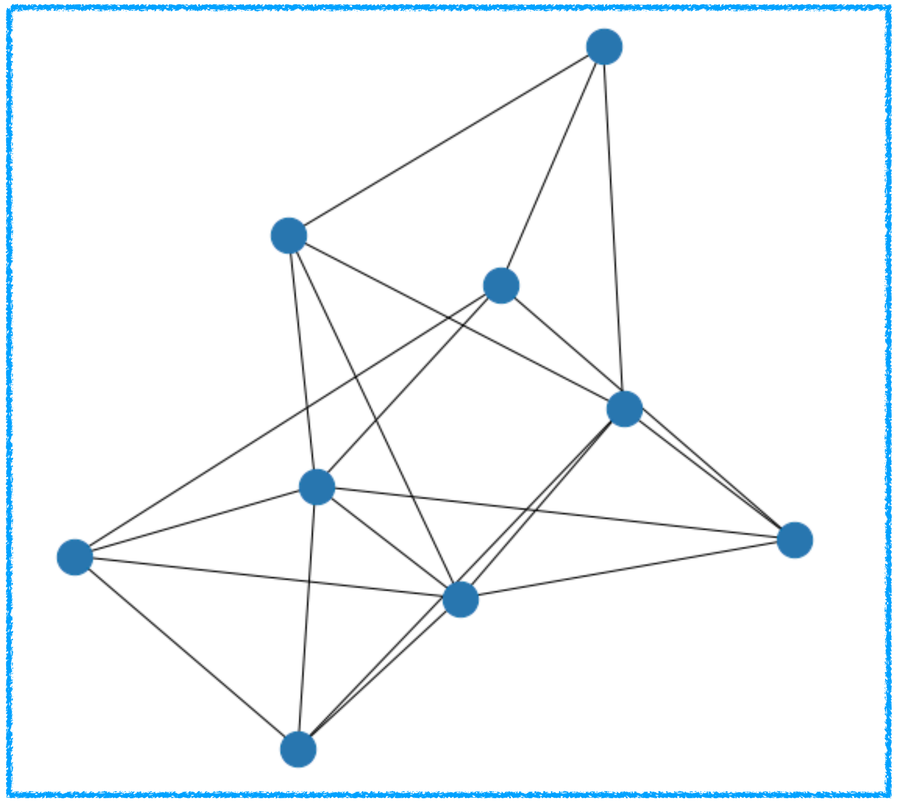


Node (or Vertex)
"친구", "배우", "지역"

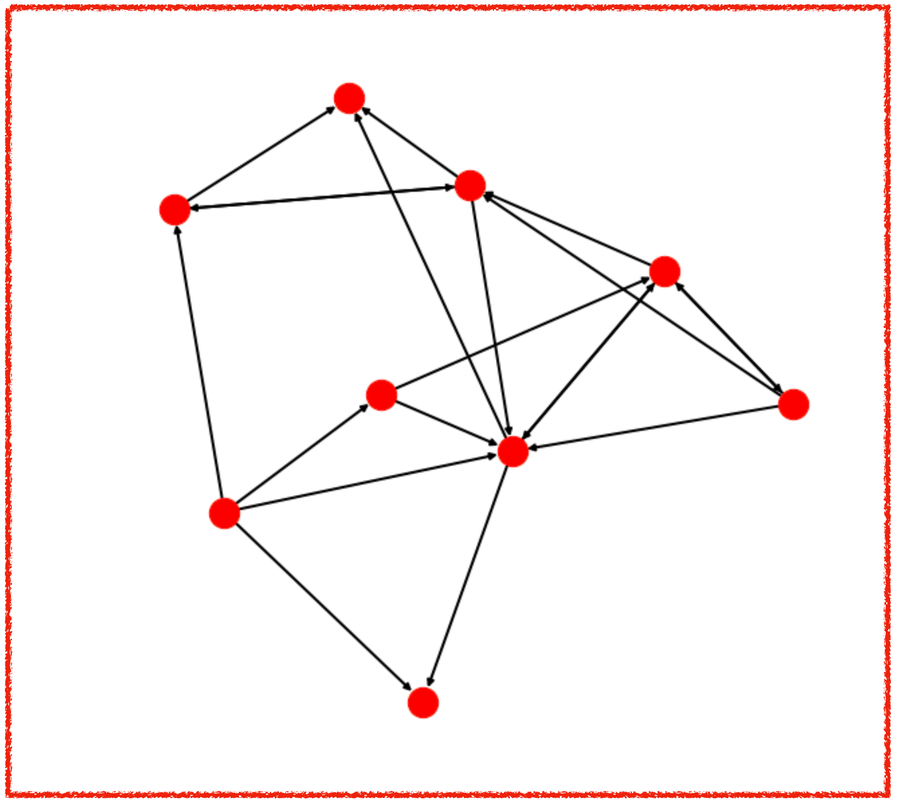
Edge (or Link)
"팔로우", "공동출연", "경로"

Network (or Graph)
"소셜네트워크", "영화(계)", "교통(망)"

Direction

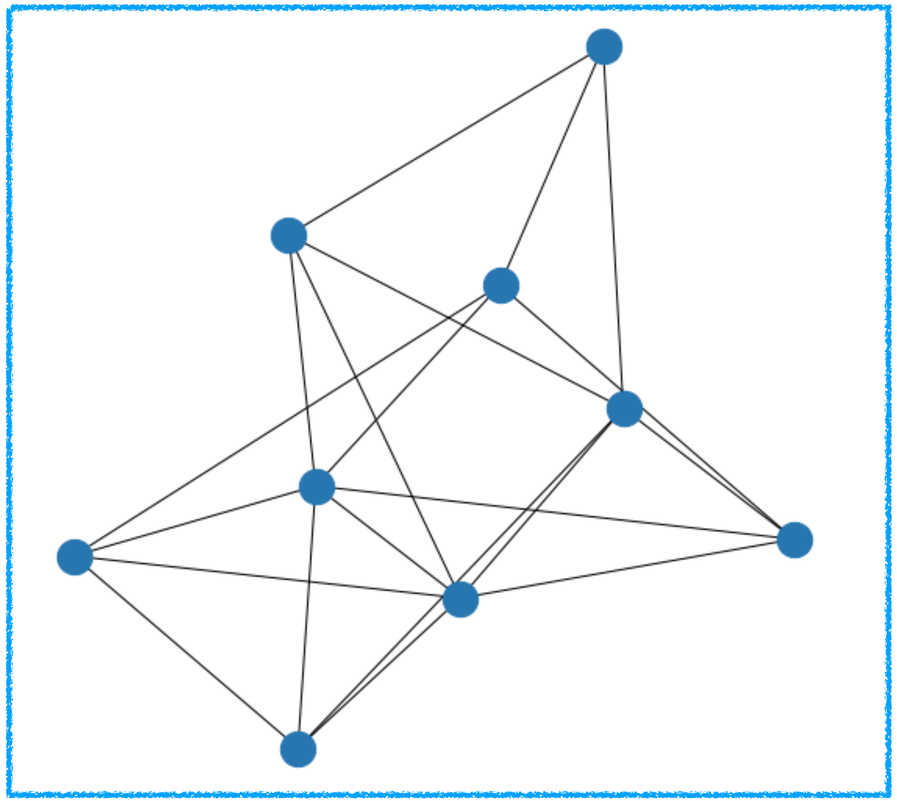


Undirected
(Symmetrical)

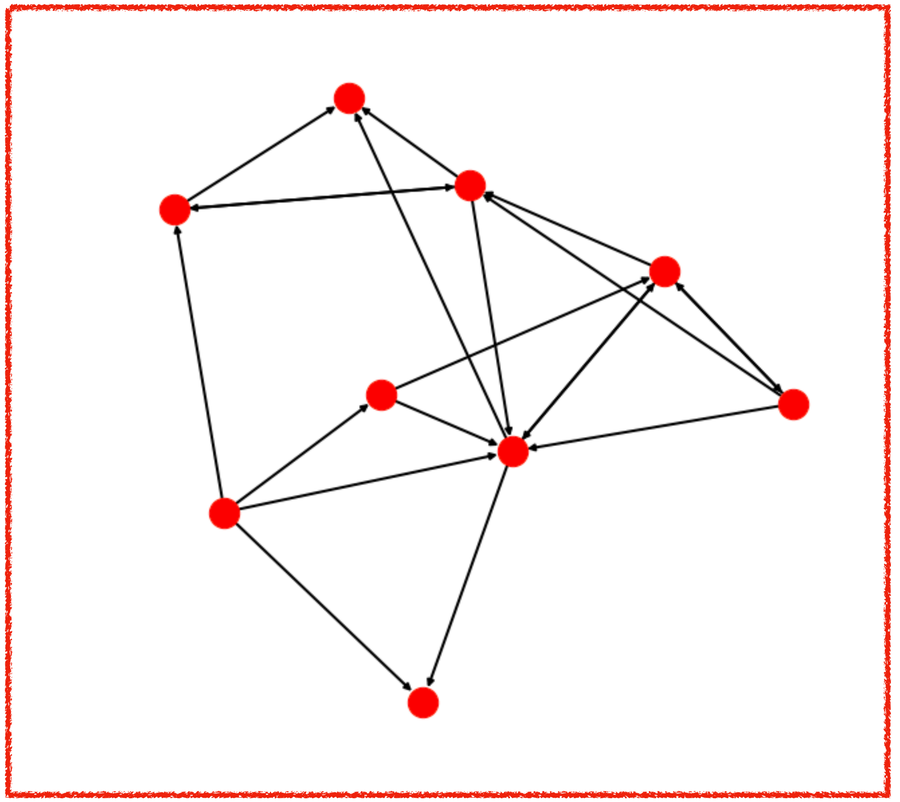


Directed
(Source → Target, Arc)

Degree?

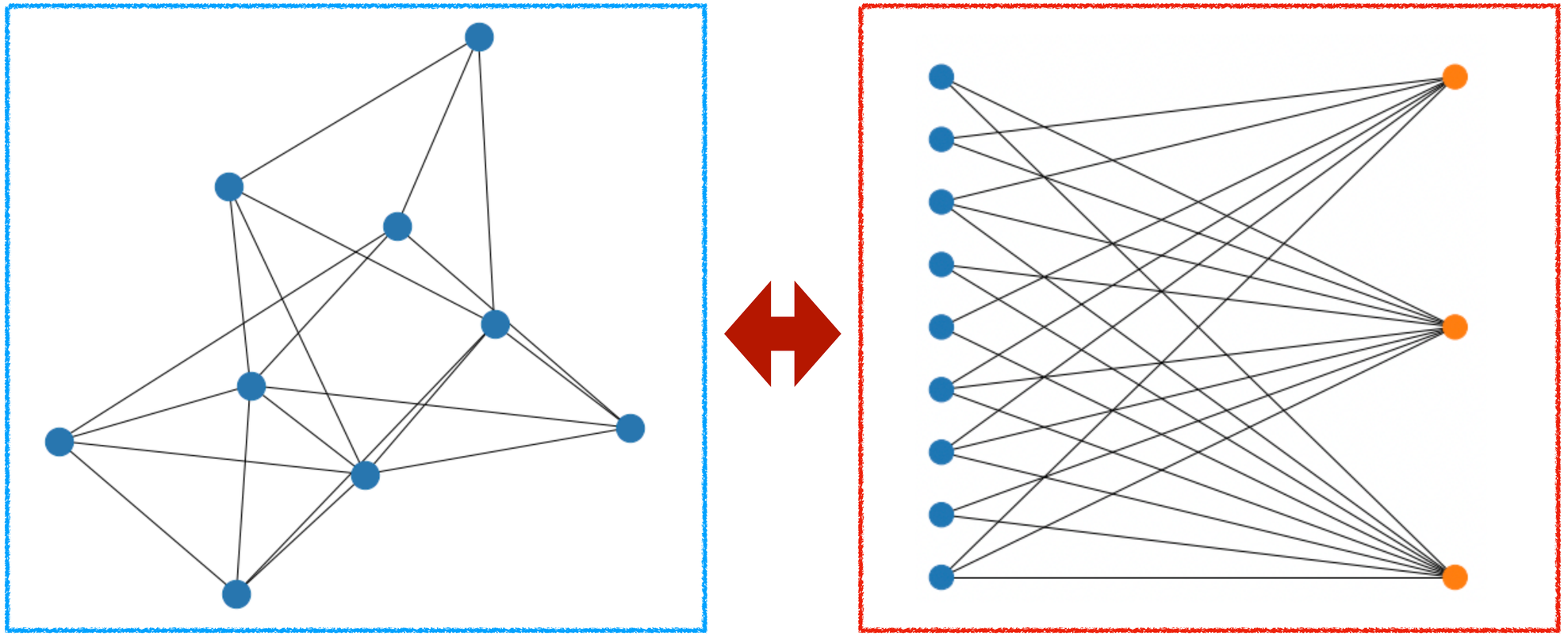


Degree : 연결된 Edge 수



(+ in/out) Degree : 연결된 Edge 수

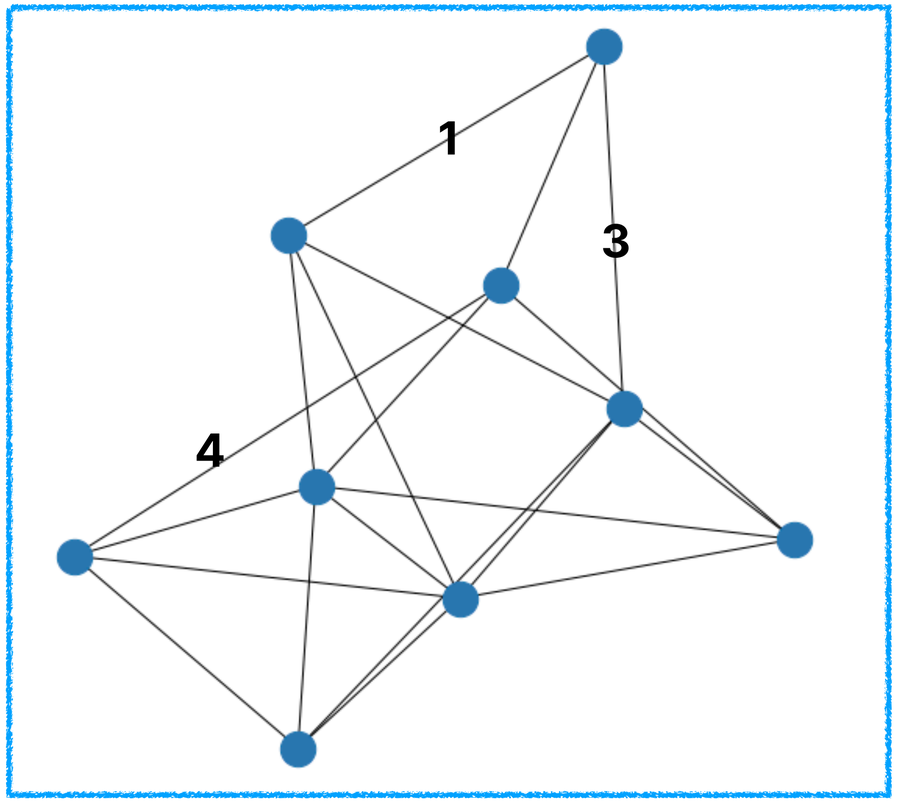
Mode



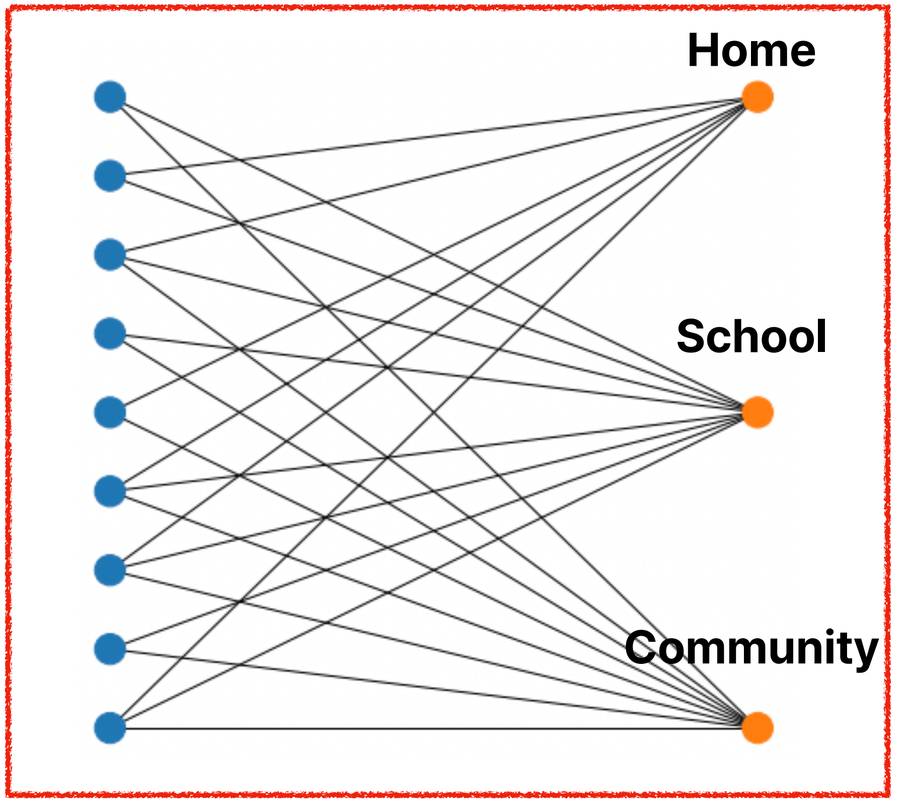
One-Mode

Two-mode/Bi-partite

Attribution - Edge/Node

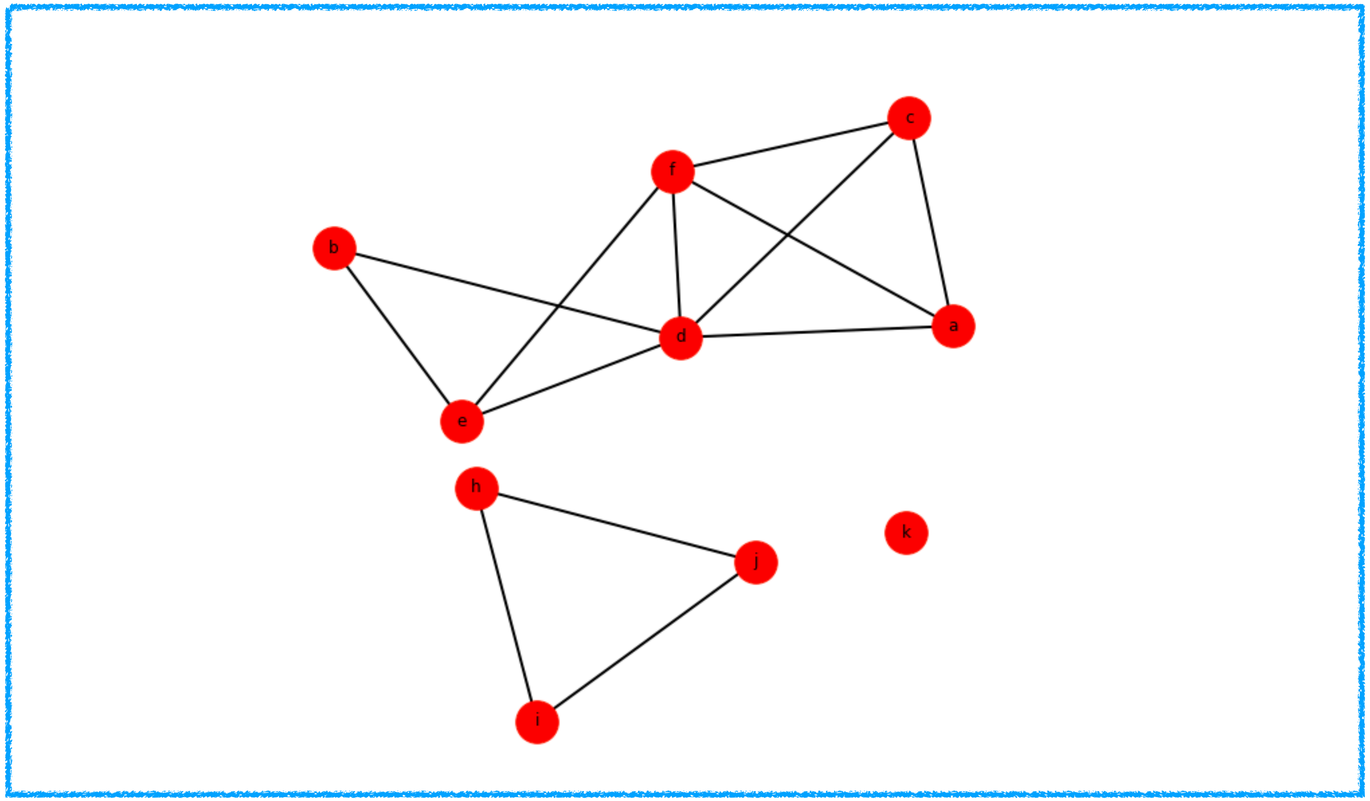


Edge (Weight)
Binary or Valued



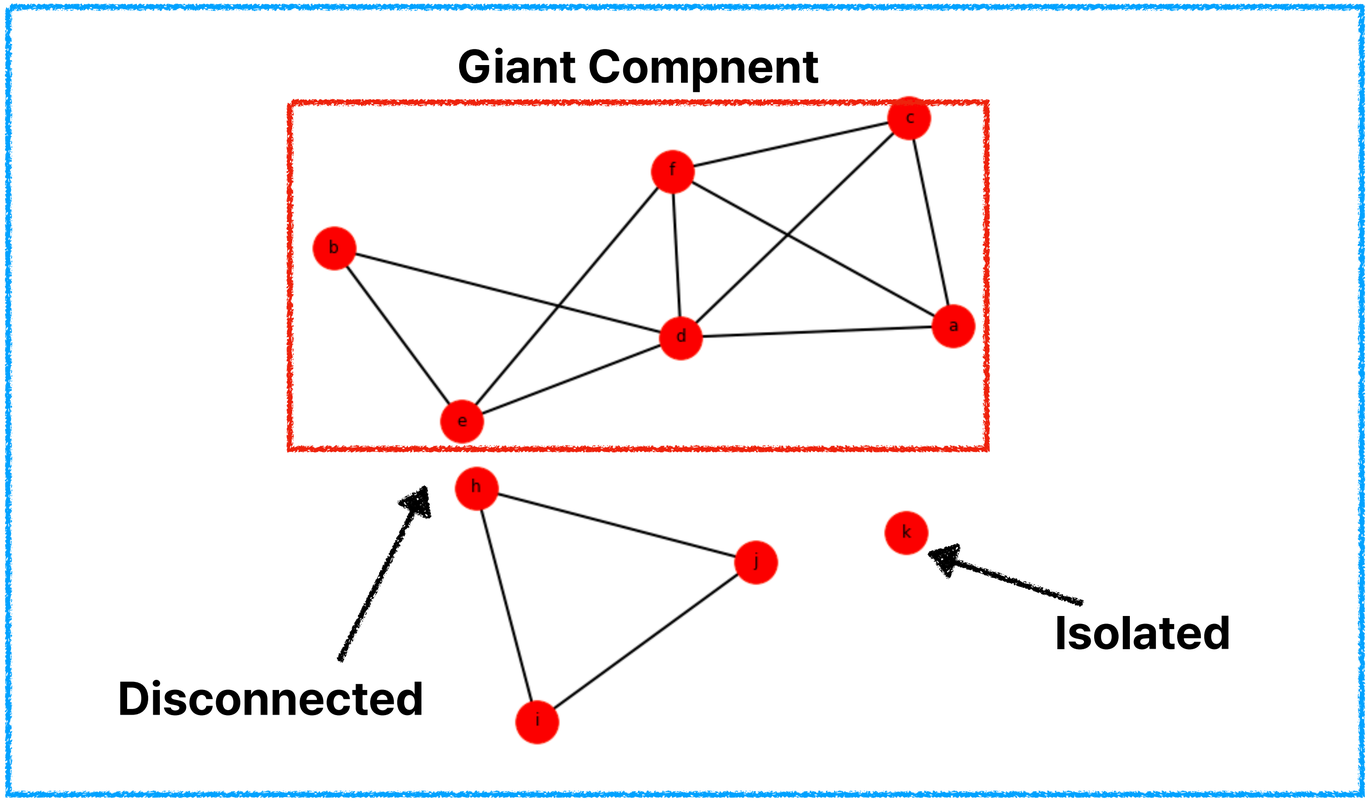
Node (Type)

Connectivity



Graph > Component (has path) > Clique

Connectivity

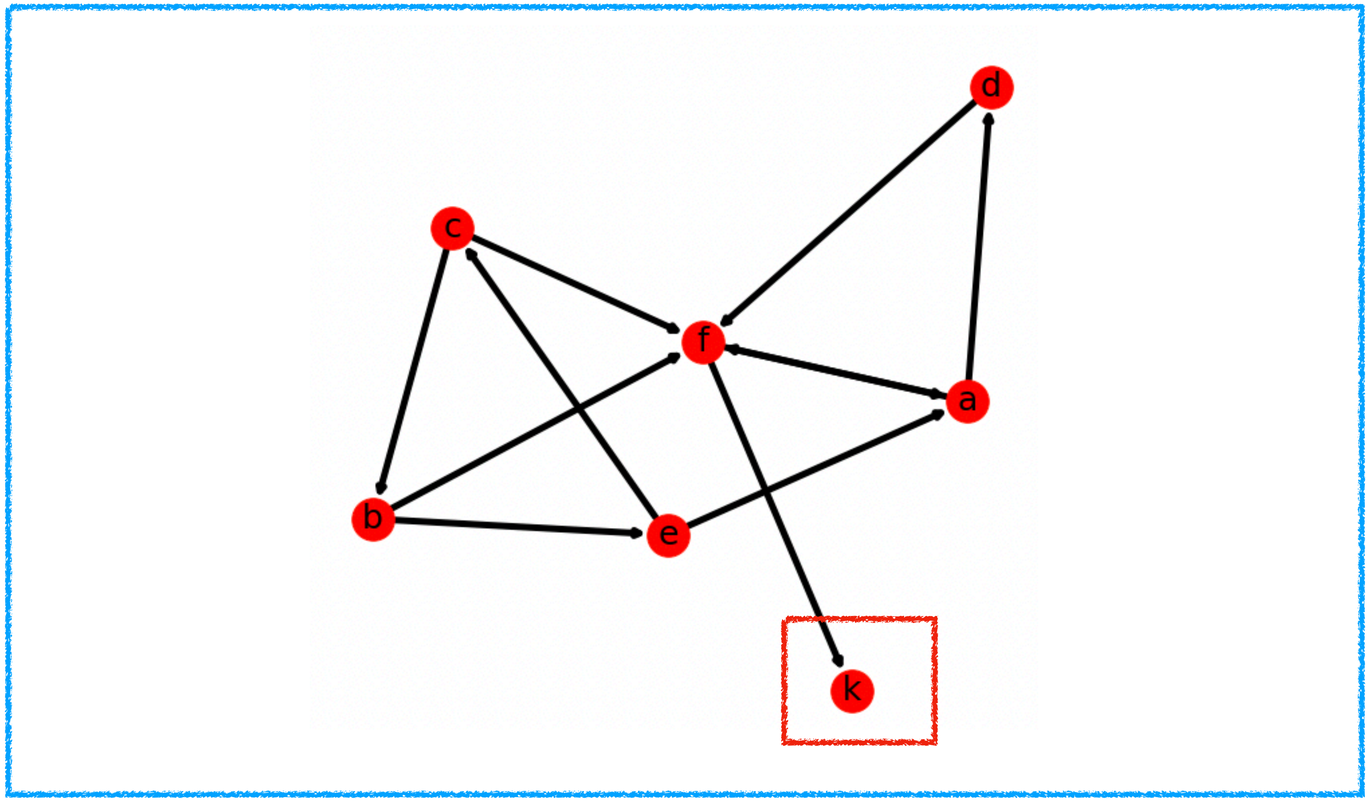


Graph > Component (has path) > Clique

Top-Down

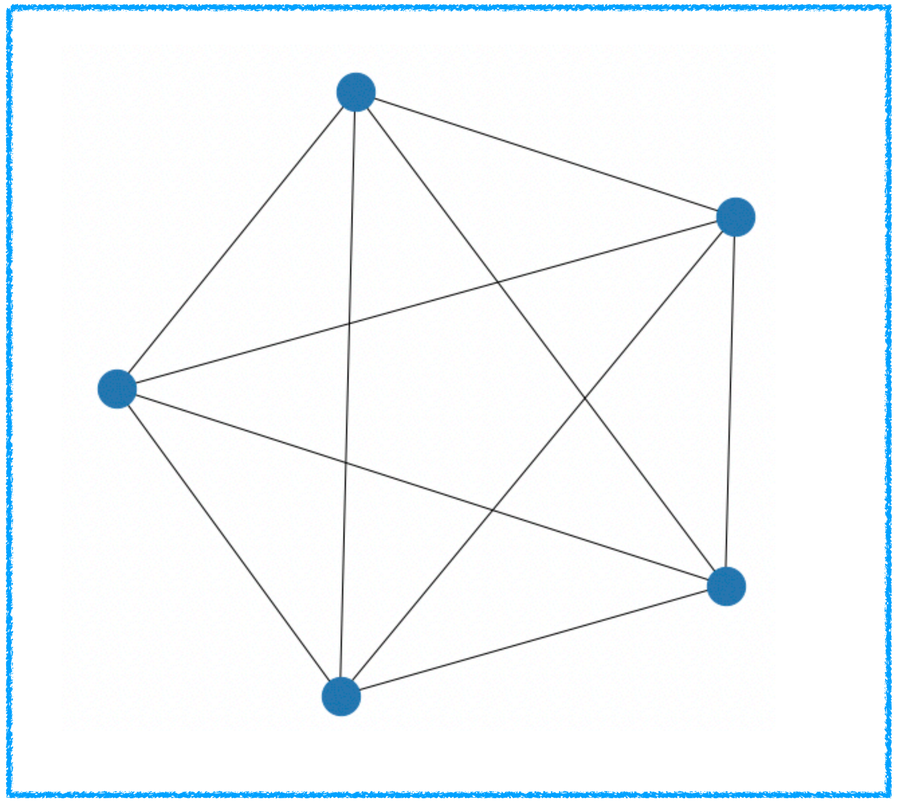
Bottom-Up

Connectivity



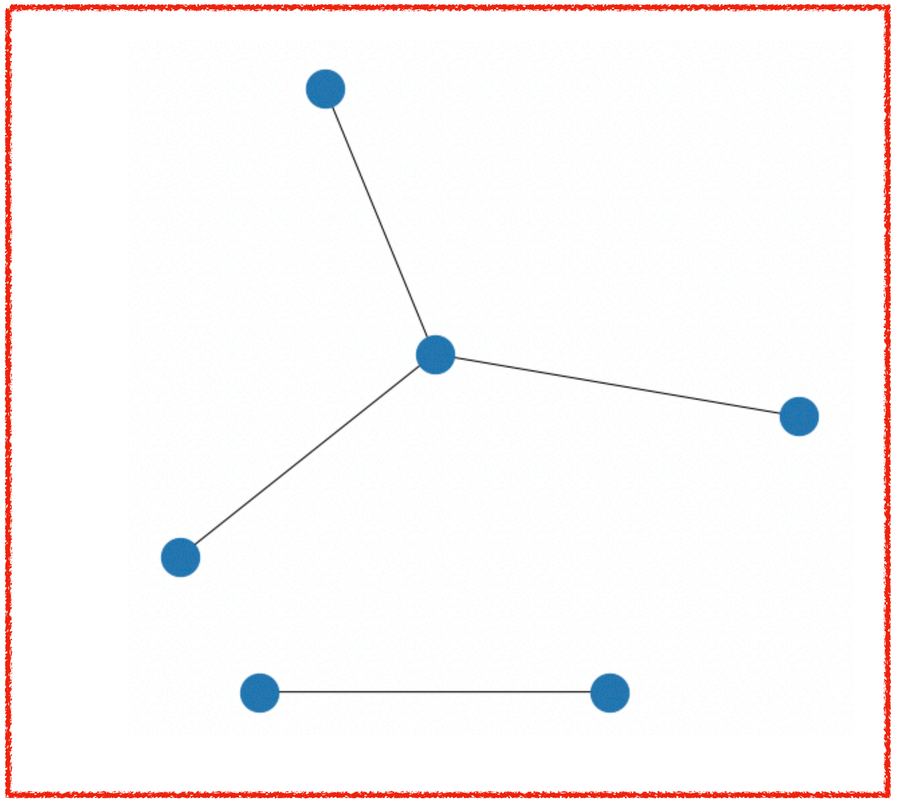
Strong vs Weak

Complete vs Sparse



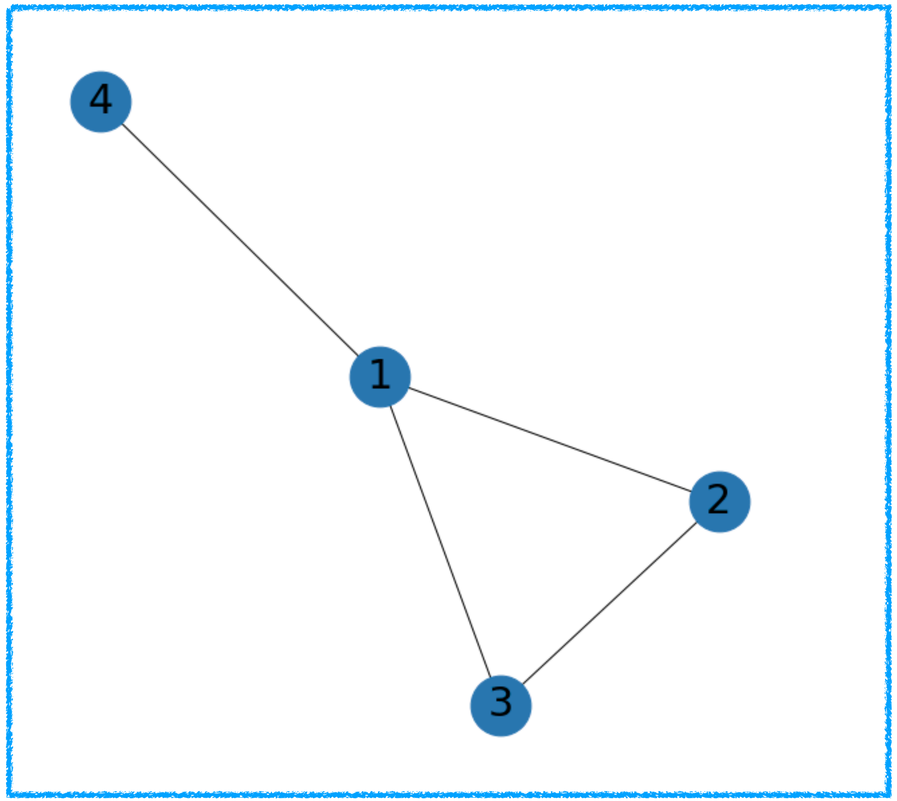
Complete

$$E_{max} = \binom{N}{2} = \frac{N(N-1)}{2}$$



Sparse
(Real World?)

Adjacency Matrix

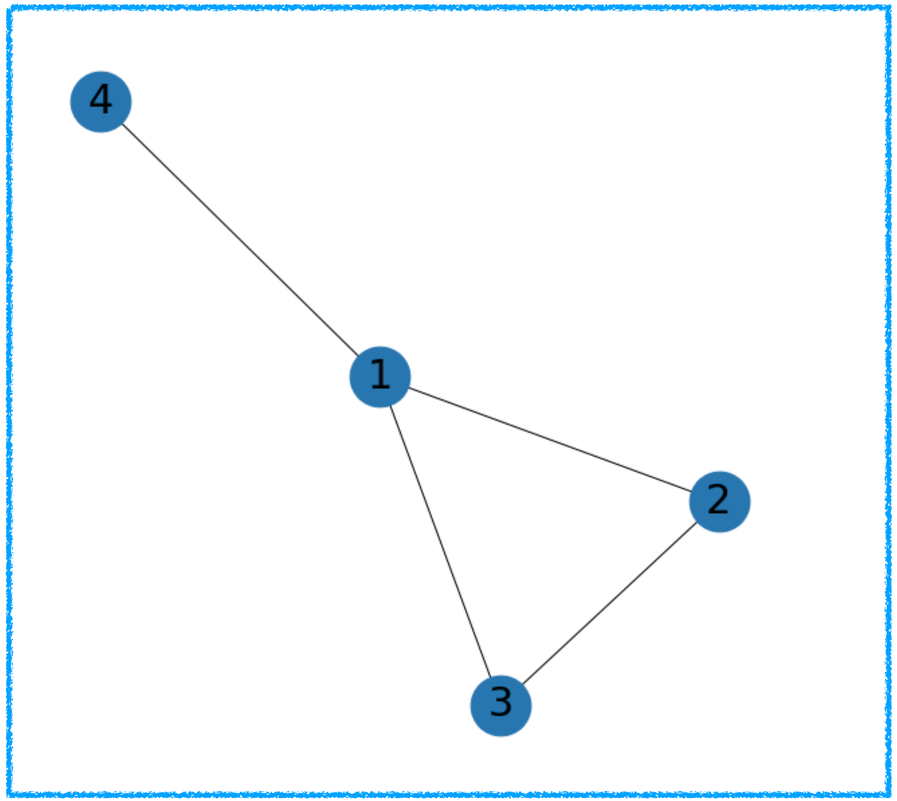


Graph (Undirected)

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

Matrix

Adjacency Matrix



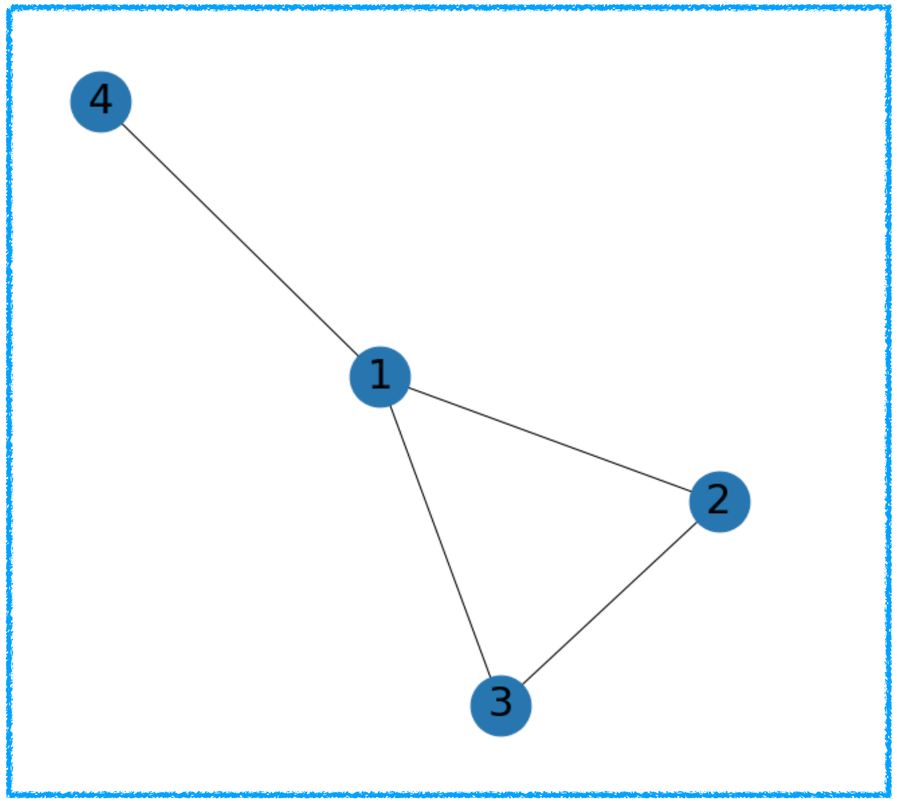
Graph (Undirected)

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

Symmetrical

Matrix

Adjacency Matrix



Graph (Undirected)

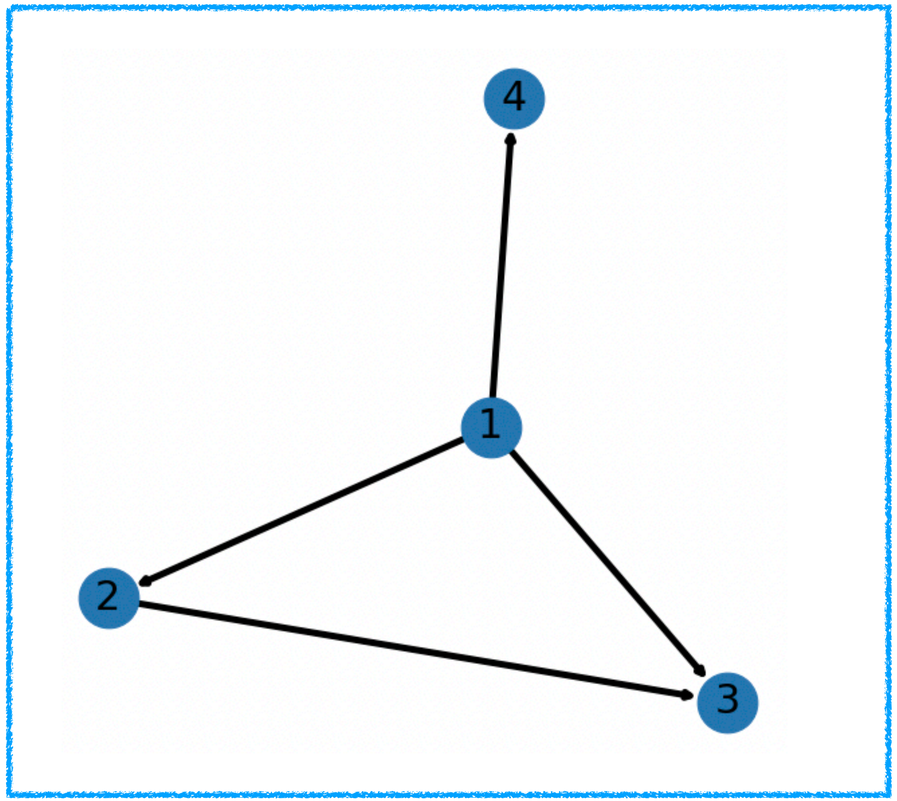
$$\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

$$X_{i,j}$$

Symmetrical

Matrix

Adjacency Matrix



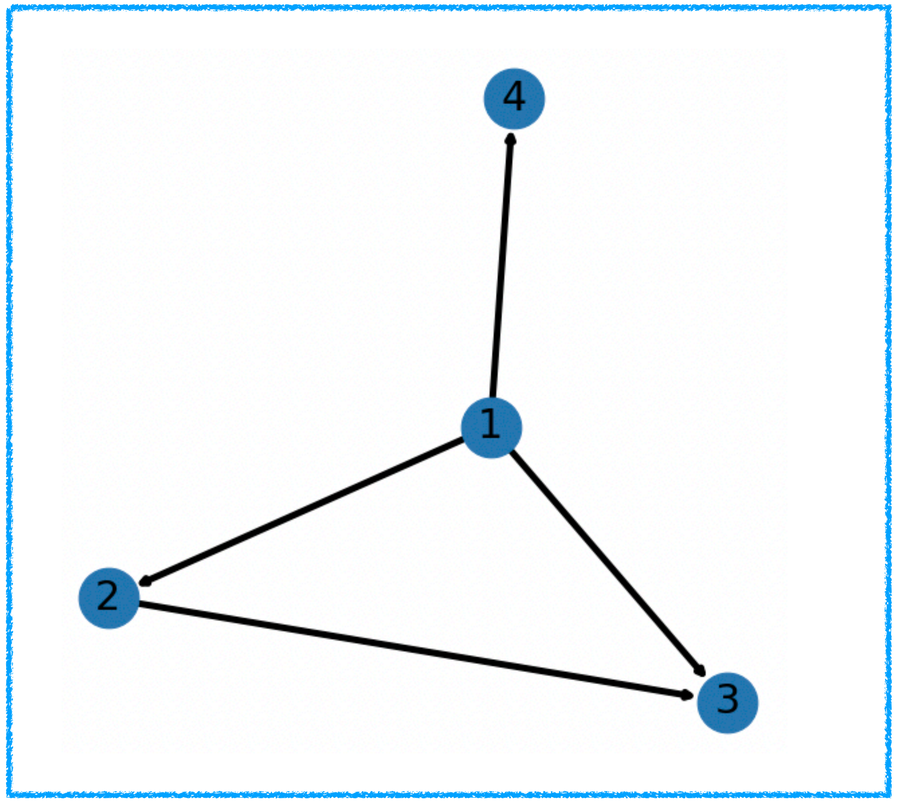
Graph (Directed)

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

Source

Matrix

Adjacency Matrix



Graph (Directed)

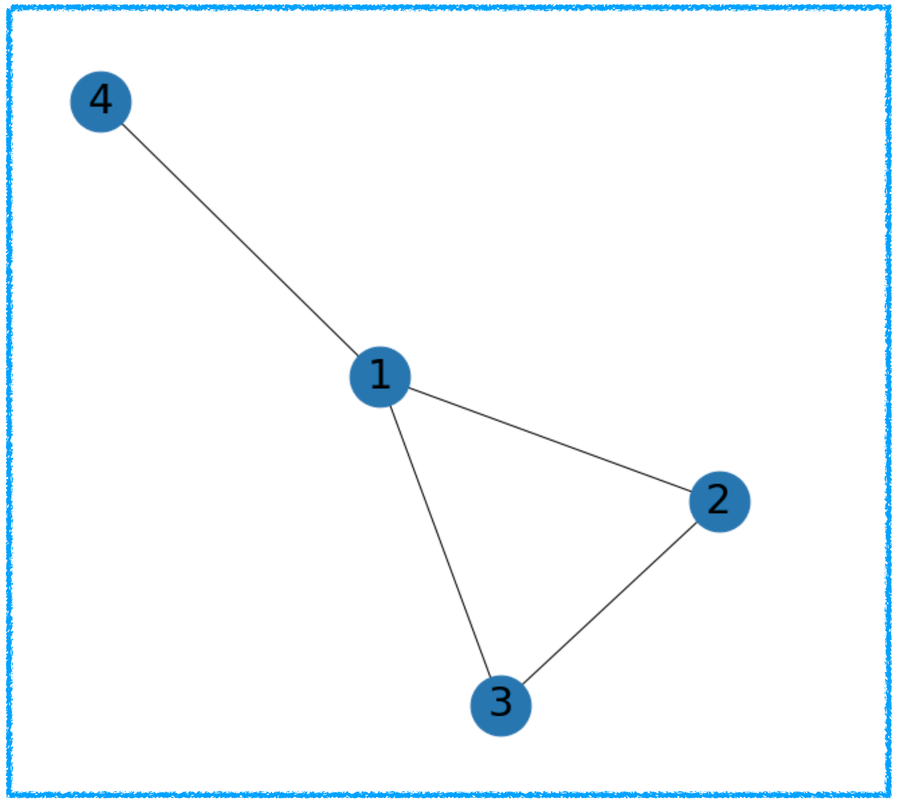
$$\begin{pmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$X_{i,j}$

~~Non Symmetrical~~

Matrix

Edge List

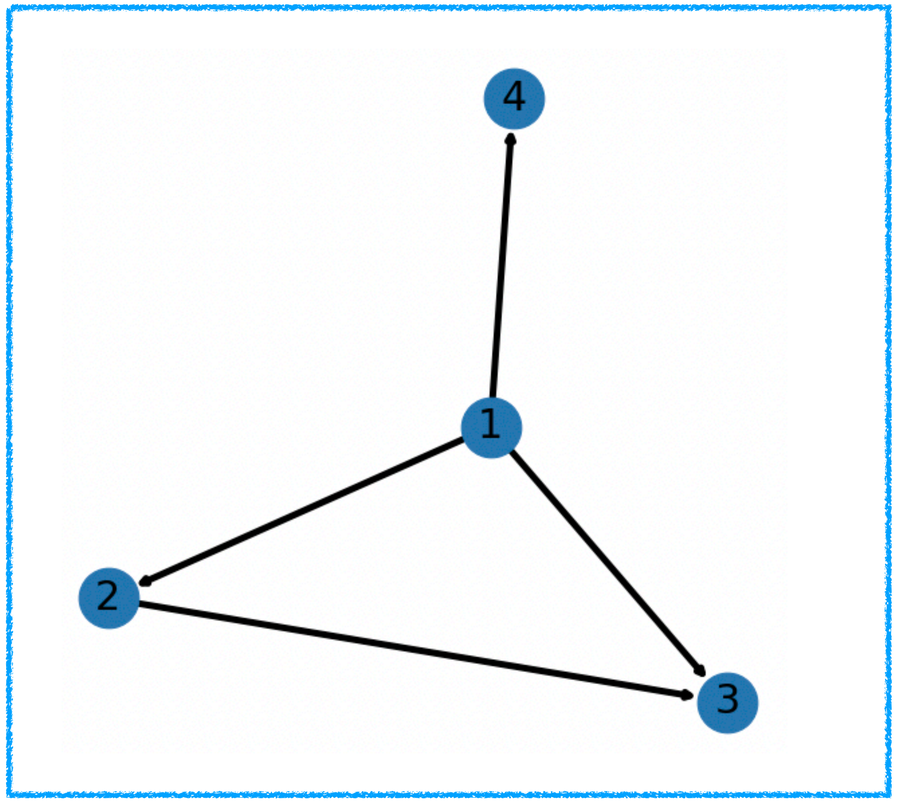


Graph (UnDirected)

Source	Target	Weight	Type
1	2	1	UnDirected
1	3	1	UnDirected
1	4	1	UnDirected
2	3	1	UnDirected

Edge List (Table)

Edge List



Graph (Directed)

Source	Target	Weight	Type
1	2	1	Directed
1	3	1	Directed
1	4	1	Directed
2	3	1	Directed

Edge List (Table)

사회연결망 기초 정리 / 연구 접근법

- 그래프 분석을 위해서는 Node, Edge 설정이 필요하다
- (설정 이후) 그래프의 특징 + 문제를 고려해서 분석 방향을 결정한다
  - 방향?
  - 모드?
  - 속성?
  - 연결(정도)?
- 그래프를 분석하기 위해 여러 형식(Format)이 활용 : Adjacency Matrix, Edge List
  - Adjacency Matrix는 대표적인 그래프 표현(입/출력) 형식이다 (연산, 통계에 유리)
  - Edge List는 Adjacency Matrix를 대체하는 또 다른 그래프 표현 형식이다 (직관적, 공간절약?)

연습문제

- 자신의 연구, 관심분야를 바탕으로 연결망을 구성한다면 어떤 형태로 Node, Edge가 구성 될 수 있는지 생각해보자

Tools for graph analysis

Software for Graph analysis (독립된 프로그램)

- Gephi
- Cytoscape
- Pajek
- SocNetV
- Neo4J
- NodeXL
- NetMiner
- UCINet

Packages for Graph Analysis (설치 패키지)

- Python 지원
  - SNAP (Stanford Network Analysis Project) (C++ 기반)
  - **Networkx**
  - iGraph (R 기반)
- R 지원
  - **iGraph**
  - statnet
  - sna
- 그리고 현재도 꾸준히 만들어 지고 있는 중 (github에서 검색하면 수많은 패키지가...)

Python for Social Network Analysis

- 많은 소프트웨어가 유료 or 무료지만 중요 기능이 제한된 경우가 많음
- (빅)데이터 연구는 수많은 전처리 과정이 요구 / 데이터 전처리 툴과 연계가 중요(!)
  - 보기 좋은(?) 데이터를 가지고 시작하는 경우가 적음
    - [전처리, 분석, 다시 전처리, 다시분석, 다시다시 전처리....] ...
- 최신 연구 결과는 주로 Python를 기반으로 패키지가 만들어짐
  - 특히, 딥러닝에 기반한 분석 방법론에서는 Python (Numpy, Scipy)과 연계가 필수적
  - 많은 분석 방법론+패키지가 Numpy, Scipy 기반

Python Notebook?

- Python을 이용하는 방식은 크게 두 가지
  - 파이썬 파일(.py)을 터미널에서 실행하는 방식
    - 간편함, 코딩 중 중간 과정 실행/확인하기 어려움, 협업 과정에 적합
  - 주피터 노트북을 통해 실행하는 방식(.ipynb)
    - 복잡, 코딩 중 실행/확인 손쉬움, 개인 연구에 적합
- Jupyter를 사용하는 일반적 과정 → Python 설치, 노트북 설치, 패키지 설치 필요
  - 최근에는 Anaconda와 같은 종합 설치 툴이 발전, 상대적으로 간편해짐

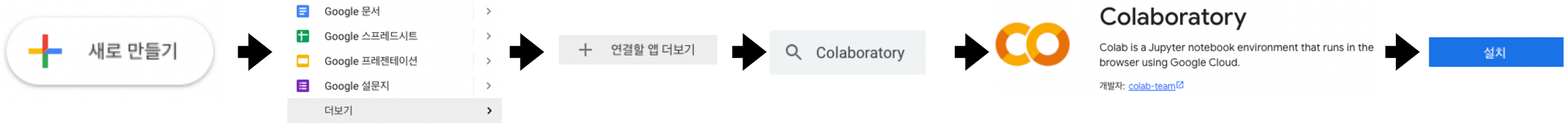
Colaboratory 테스트/설정

Google Colaboratory?

- 구글 드라이브에서 실행 가능한 코딩 작업 공간(Jupyter Notebook 형식)
  - 구글 드라이브에 접속되는 PC가 있다면 누구나 사용 가능
- 성능 및 시간에 제한이 있지만, 개인이 사용하기에는 거의 제한이 없는 수준
  - 크롤링 같은 장기작업(10시간 초과), 고도의 딥러닝 작업(TPU 요구)을 위해서는 과금 필요
- 구글 드라이브에서 콜랩 파일을 시작하거나, 노트북 파일(ipynb)을 업로드 해서 사용
- 다른 코딩 플랫폼이나, 구글의 다른 서비스와 연계 가능
  - Github과 연계, 구글 드라이브 파일 불러오기/저장하기

Google Colaboratory?

- 처음 콜랩을 실행하는 경우, 구글 드라이브에서 부가기능 설치 필요



- 설치된 이후에는 새로 만들기에서 간단히 실행 가능

파이썬 실습 1
그래프 입력/시각화

Networkx Basic

- Networkx를 통해 다음과 같은 기본 조작 가능
  - 그래프 객체 생성
    - 생성된 그래프 객체에 노드/엣지 추가
    - 노드/엣지 한번에 추가
  - 그래프 객체 저장/불러오기
    - Adjacency Matrix
    - Edge List
  - 시각화 확인

Networkx Basic

- Networkx는 기본적으로 "딕셔너리" 형태
  - G = {"Node" : "Information (neighbors, type ...)"}
    - G.nodes ⇒ [모든 노드정보]
    - G.edges ⇒ [모든 엣지정보]
- 위 구조에 따라 구조를 빠르게 확인할수 있음
    - G['하나의 노드'] ⇒ 이웃(연결된) 노드의 정보 (딕셔너리)
    - G['연결된 노드1']['연결된 노드2'] ⇒ 두 노드 사이의 정보, Attribution (e.g. Weight)
- for node in G : 모든 노드에 대해 반복작업 생성(!)

Graph Format?

- 일반적으로 그래프 객체는 Edgelist, Adjacency Matrix 사용
  - 대용량일수록 Format이 간단한 경우가 많음(용량/호환성 측면)
  - 노드/엣지를 같이 표기하거나, ID를 기준으로 표기하는 경우가 다수
- 여러 그래프 속성이 추가될경우 (e.g. 노드에 이름 외 다른 값 부여) 그래프 전용 포맷을 사용하기도 함 (본질적으로는 비슷한 형식)
  - GEXF (Graph Exchange XML Format)
  - GML (Graph Modeling Language)
  - 범용 포맷인 Json등을 사용하는 경우도 존재 (본 수업에서 따로 다루지는 않음)

Graph Format?

- Networkx 지원 그래프 포맷 (Adjacency Matrix, Edge list, Json 제외)
  - DOT - Graphviz 포맷
    - GML - DOT과 유사
  - GEXF - XML 기반
  - GraphML - XML 기반
  - Pajek - Edge List와 노드, 엣지 정보 결합
  - 기타 - LEDA, SparseGraph6, GIS Shapefile, Matrix Market, Pickle ...

Graph/Network Features

그래프의 속성을 어떻게 측정할까?

- Centrality - 해당 Node가 얼마나 중요한지 수치화
  - Degree
  - Betweenness
  - Closeness
  - Eigenvector
- 수많은 Centrality 측정법이 존재 (e.g. Eigenvector의 확장/개선 모델들)
  - Katz Centrality, PageRank (Web Based), CCoef

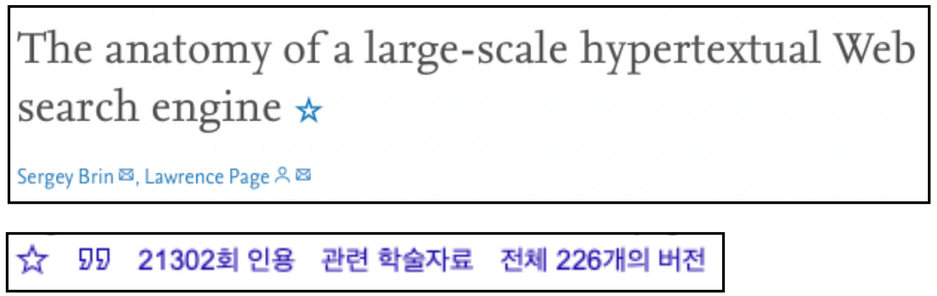
그래프의 속성을 어떻게 측정할까?

- Éminence grise?

images : https://unsplash.com

그래프의 속성을 어떻게 측정할까?

- Pagerank?
  - 구글의 창업 멤버인 래리 페이지, 세르게이 브린이 개발한 알고리즘
  - 웹의 특성을 고려하여 중요도를 판단



left-pad 사건

- 2016년 3월 22일, 전세계 Node.js 커뮤니티의 수천개 프로그램에서 오류 발생
  - 소규모 프로젝트부터 페이팔, 야후, 넷플릭스 같은 대규모 프로젝트까지 오류 확인
- Azer Koçulu 라는 개발자가 운영진에 대한 항의로 자신의 모든 프로젝트를 삭제하는 과정에서 left-pad라는 11줄짜리 코드가 삭제된 것이 원인
  - left-pad는 line-numbers라는 프로젝트에, line-numbers는 babel을 포함한 대규모 프로젝트에 사용되고 있었음
    - 작은 코드 블록에서 연결된 모든 프로젝트에서 오류 발생
  - (의존성의 높은) 오픈 소스 연결망 구조에 내재된 구조적 특징(or 문제)을 보여줌

파이썬 실습 2
그래프 분석/시각화

그래프의 속성을 어떻게 측정할까?

- Community
  - 그래프에서 군집된 하위 집단(Cohesive Subgroup) 을 파악하는 접근법
    - 그라노베터의 연구(1973) - The Strength of Weak Ties (세상은 Cluster...?)
  - 전체 연결망 내의 하위 집단을 측정하는 방법중 보편적으로 사용
    - Component, Community, Clique ....
  - 분석을 위해 수많은 알고리즘이 고안 /연구 중
    - Girvan-Newman algorithm, Louvain algorithm ....

Modularity

- 커뮤니티의 기준? → 간단히 생각하면 "(집단)내부는 견고하고, 외부는 배타적" (연결기준)
- 좀 더 수치적으로 명료한 접근법은 없을까? → Modularity (Q)
  - 각 커뮤니티에 대해 [#실제 그룹내의 연결 - #(Null Model)]
  - Modularity가 최대화되는 지점?
    - 여러 Modularity가 고안 / 연구 (e.g. Girvan-Newman, 2004)

$$Q = \frac{1}{2m} \sum_{i,j} [A_{ij} - \frac{k_i k_j}{2m}] \delta(c_i, c_j)$$

Null Model

Indicator Function

Modularity Matrix

파이썬 실습 3
커뮤니티 탐색/시각화

사회 연결망 분석 (논의/사례)

연결망 분석의 다양한 사례들

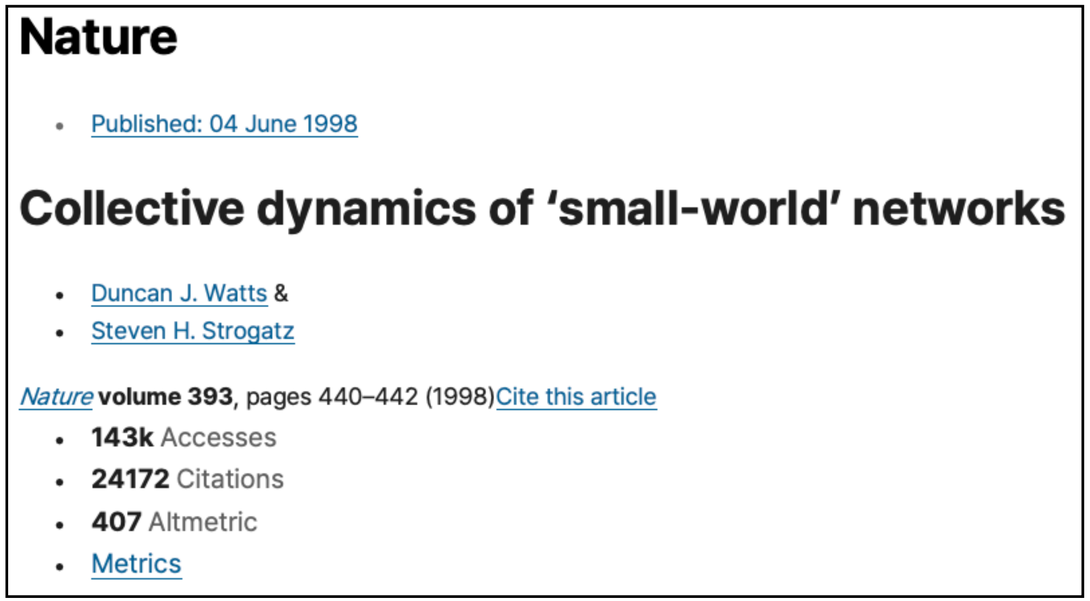
- 커뮤니케이션/소셜 네트워크 (가짜뉴스, 인플루언서)
- 금융 (이상 거래 탐색)
- 컴퓨터 네트워크 (바이러스 전파, 최적경로)
- 생물의학 (단백질 상호 작용, 유전정보)
- 자연어/NLP (키워드 추출, Knowledge Graph)
- 생태계 구조 (Food Web)
- 교통량 (최단경로 탐색, 문제지역 예측)

밀그램부터 SNS까지

- 밀그램의 실험 (1967)
  - (오마하, 위치타)의 300명의 인원을 대상으로 보스턴의 주식 브로커에게 편지를 전달하는 실험을 수행
  - 64개의 편지가 전달
    - 1~10 단계 내에서 편지가 전달 **(평균 6.2 단계, Six degrees of separation)**
    - 주식을 가지고 있거나 / 보스턴 출신 인물의 경우 평균적으로 더 짧은 단계를 거침
      - 주식을 가지고 있는 인물(5.4), 보스턴 출신의 경우 (4.4)

밀그램부터 SNS까지

- 던컨 와츠, 스티븐 스트로가츠 → 1998년 Nature에 논문 발표



Nature

- Published: 04 June 1998

Collective dynamics of ‘small-world’ networks

- Duncan J. Watts &
- Steven H. Strogatz

Nature **volume 393**, pages 440–442 (1998)Cite this article

- **143k** Accesses
- **24172** Citations
- **407** Altmetric
- Metrics

☆ 99 47337회 인용 관련 학술자료 전체 152개의 버전

그래프의 속성을 어떻게 측정할까?

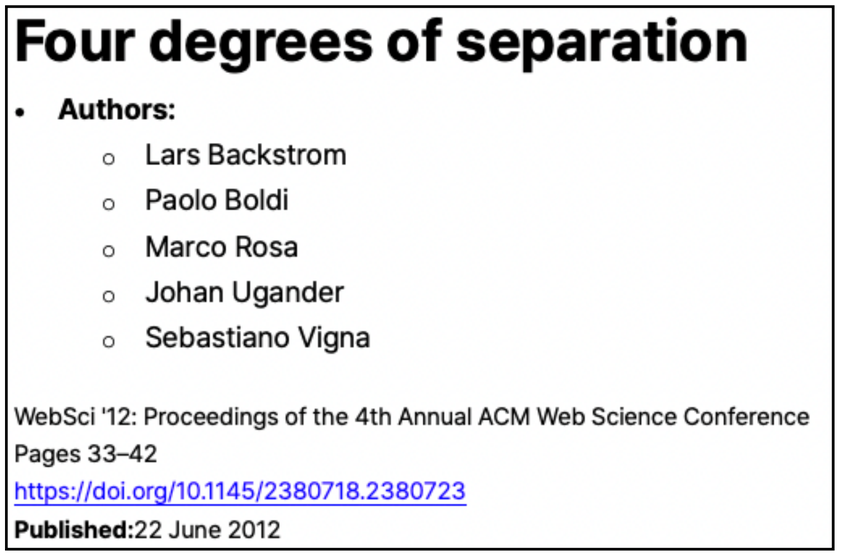
- Clustering coefficient
  - 기준 노드(i)와 연결된 주변 노드(ki)가 얼마나 밀집한 관계를 갖는가?

$$C_i = \frac{2e_i}{k_i(k_i - 1)}$$

- Ci가 높을수록 (구조가/이웃이) 밀집되어 있다

밀그램부터 SNS까지

- 소셜 네트워크에서는 어떨까?
  - 2011년 5월 기준, 한달 동안의 Facebook 빅데이터
    (약 7억명의 유저, 690억개의 연결) 수집/분석 → 그래프(연결)의 특성 탐색



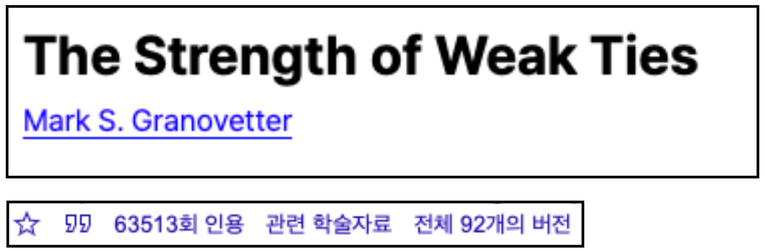
Four degrees of separation

- **Authors:**
  - Lars Backstrom
  - Paolo Boldi
  - Marco Rosa
  - Johan Ugander
  - Sebastiano Vigna

WebSci '12: Proceedings of the 4th Annual ACM Web Science Conference
Pages 33–42
https://doi.org/10.1145/2380718.2380723
Published:22 June 2012

밀그램부터 SNS까지

- 세상은 Cluster + 의외의 연결
  - 구직 활동을 통해 확인한 인간 사회 연결망의 구조(특이점)



밀그램부터 SNS까지

- 온라인 상에서 우리는 정말로 토론하고 의견을 교환하는가?

ARTICLE

The political blogosphere and the 2004 U.S. election: divided they blog

- **Authors:**
  - Lada A. Adamic
  - Natalie Glance

LinkKDD '05: Proceedings of the 3rd international workshop on Link discovery
Pages 36–43
https://doi.org/10.1145/1134271.1134277
Published:21 August 2005

☆ 99 3346회 인용 관련 학술자료 전체 20개의 버전