

# 공동발의 데이터를 활용한 국회의원 간 친밀도 시각화

# Table of Contents

Points to discuss:

개요	01
데이터 소개	02
Network Model	03
결론	04

# 개요

Our Objective 01

Method 02

활용 대상 03

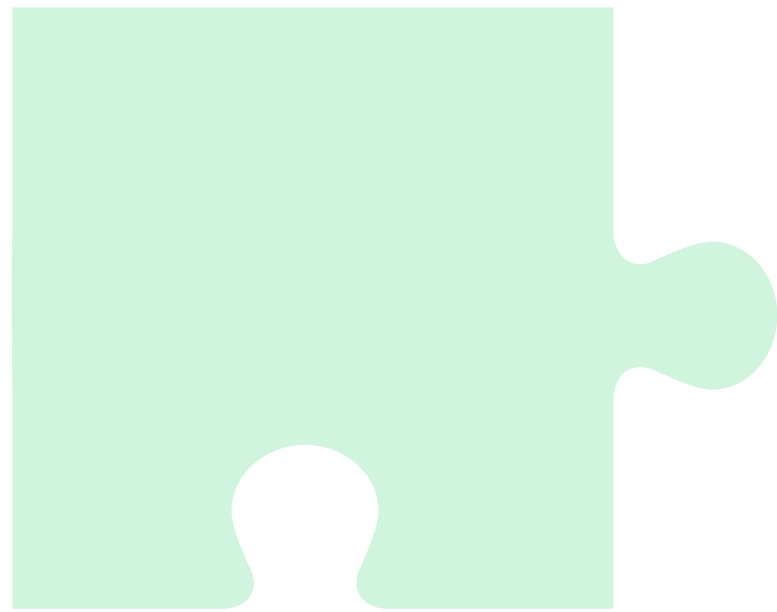
# Our Objective



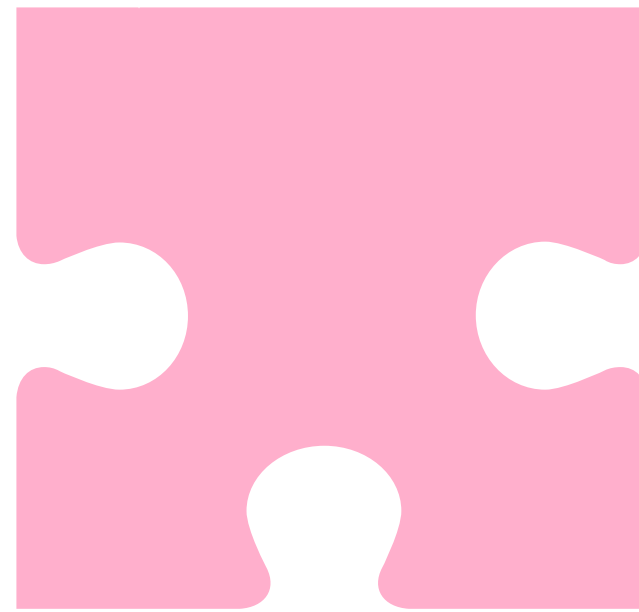
**공동발의로 국회의원 사이 '친밀도' 측정**



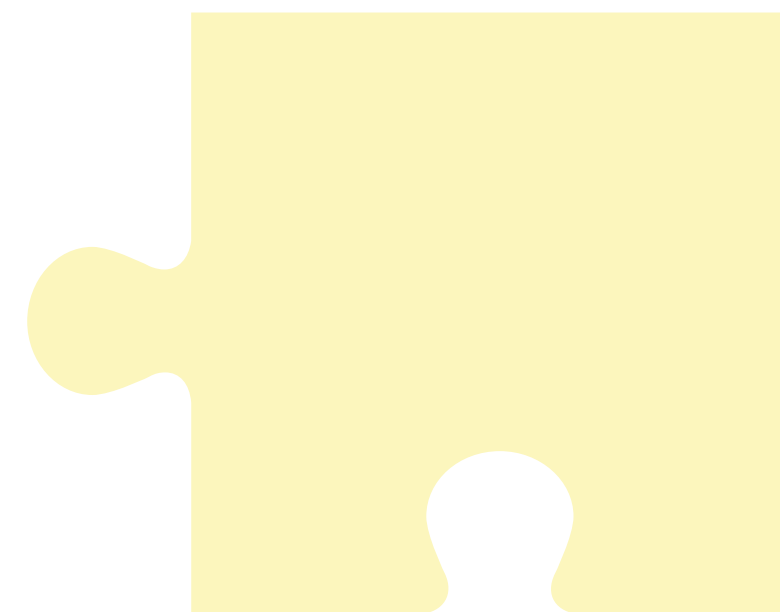
# Method



국회 API  
국회의원 데이터

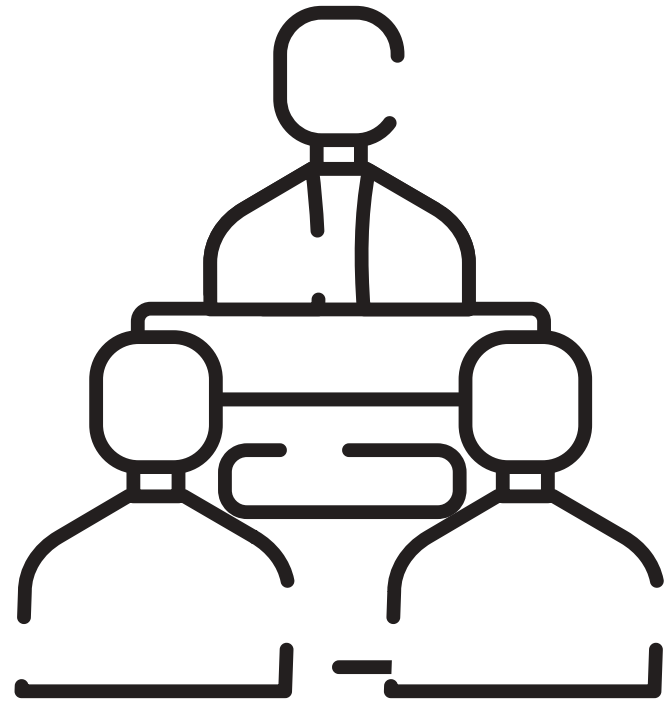


Pyvis Network  
Model



모델 분석

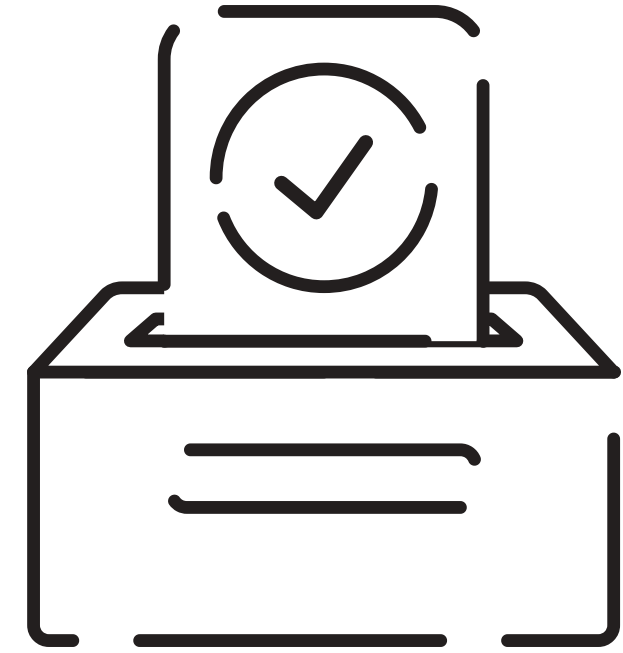
# 활용 대상



정계 입문자



입법(개정) 시도



유권자

# 데이터 소개

데이터 수집 01

국회의원 발의법률안 API 02

국회의원 인적사항 API 03

# 데이터 수집



<https://open.assembly.go.kr/portal/data/service/selectAPIServicePage.do/OK7XM1000938DS17215>



# 국회의원 발의법률안 API



국회의원 발의법률안 데이터

국회의원별 대표발의법률안 정보를 제공합니다.

총평점 ★★★★★ ★★★★★ OK

변수명	변수 설명	변수 예시
<u>BILL_NO</u>	의안번호	BILL_NO = '2113663'
AGE	대수	AGE = '21'
PROPOSE_DT	제안일	PROPOSE_DT = '2024-04-25'
PUBL_PROPOSER	공동발의자	PUBL_PROPOSER = '이달곤, 김미애, 이종성'
RST_PROPOSER	대표발의자	RST_PROPOSER = '강기윤'

# 국회의원 인적사항 API



국회의원 인적사항 데이터

국회의원별 소속정당, 위원회, 재선여부, 당선대수 등 인적정보를 제공합니다.

총평점 ★★★★★ ★★★★★ OK

변수명	변수 설명	변수 예시
POLY_NM	정당명	POLY_NM = '정당명 검색어' (예시) POLY_NM = 정의당
HG_NM	이름	HG_NM='이름 검색어' (예시) HG_NM=황희

# Network Model

데이터 활용 방법 01

구현 코드 02

프로토타입 03

구현 기능 04

# 데이터 활용 방법

목적: **공동발의**로 국회의원 사이 '**친밀도**' 측정

최근 정당 양극화 추세를 반영하여 정당을 최우선 기준으로 세움

공동발의 기준으로 국회의원 사이 친밀도 측정

공동발의 횟수 = 연결선의 두께(weight)

2번 이상 공동발의

날짜 기준: 2024년 3월 1일 이후

# 구현 코드

```
# Function to convert date string to datetime object
def parse_date(date_string):
    ✨ return datetime.strptime(date_string, '%Y-%m-%d')

# Initialize an empty graph
G = nx.Graph()

# Iterate over the list of bills
for bill in bill_data:
    if bill['PROPOSE_DT'] and parse_date(bill['PROPOSE_DT']) > parse_date('2024-03-01'):
        # Combine PUBL_PROPOSER and RST_PROPOSER
        proposers = []
        if bill['PUBL_PROPOSER'] and isinstance(bill['PUBL_PROPOSER'], str):
            proposers.extend(bill['PUBL_PROPOSER'].split(','))
        if bill['RST_PROPOSER'] and isinstance(bill['RST_PROPOSER'], str):
            proposers.append(bill['RST_PROPOSER'])

        proposers = list(set([p.strip() for p in proposers if p.strip()]))

        for i in range(len(proposers)):
            for j in range(i + 1, len(proposers)):
                if proposers[i] and proposers[j]:
                    if G.has_edge(proposers[i], proposers[j]):
                        G[proposers[i]][proposers[j]]['weight'] += 1
                    else:
                        G.add_edge(proposers[i], proposers[j], weight=1)

# Filter edges by weight, keeping only those with weight > 5
edges_to_remove = [(u, v) for u, v, d in G.edges(data=True) if d['weight'] <= 1]
G.remove_edges_from(edges_to_remove)

# Visualization using PyVis
nt = Network(notebook=False)
for node, attrs in G.nodes(data=True):
    nt.add_node(node, title=node, color=attrs.get('color', 'gray'))

for u, v, attrs in G.edges(data=True):
    nt.add_edge(u, v, value=attrs['weight'])

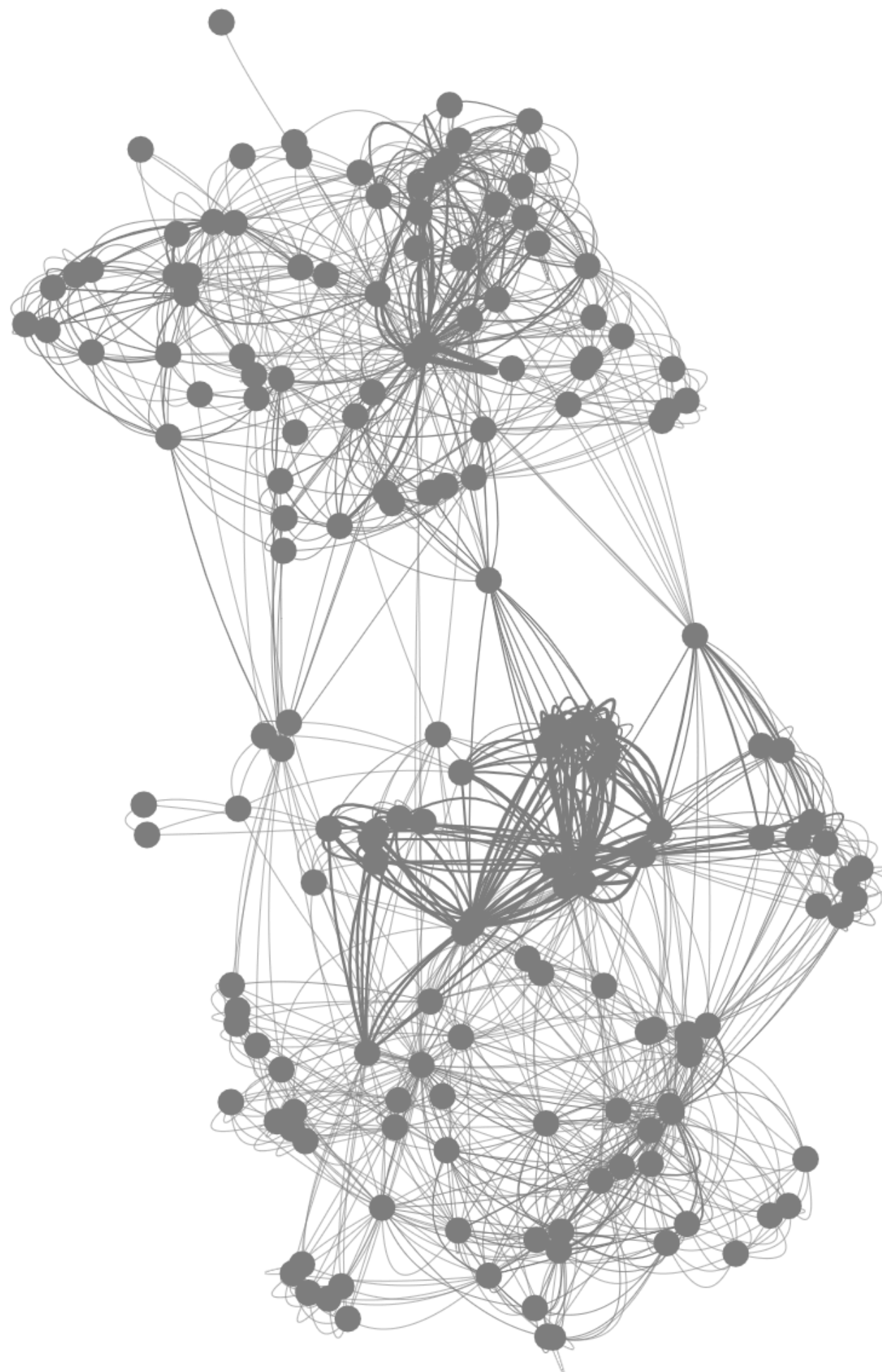
nt.show_buttons(filter_=['physics', 'nodes', 'edges'])

nt.show('bill_network(no_color).html')
```

✓ 0.2s



# 프로토타입



# 정당별 색 지정

```
# Define a color scheme for parties
party_colors = {
    '국민의힘': '#E61E2B', # Example: Party name and
    '국민의미래': '#E4002B',
    '더불어민주당': '#152484',
    '더불어민주연합': '#1A2C7B',
    '녹색정의당': '#FFF100',
    '새로운미래': '#45BABD',
    '개혁신당': '#FF7210',
    '조국혁신당': '#0073CF',
    '자유통일당': '#E24A49',
    '새진보연합': '#00D2C3',
    '진보당': '#D6001C',
    # Add more parties and their colors as needed
}
```

# 조절 기능 추가

## physics

enabled: ☒

### barnesHut

theta:	<input type="range"/>	0.5
gravitationalConstant:	<input type="range"/>	-43450
centralGravity:	<input type="range"/>	0.2
springLength:	<input type="range"/>	265
springConstant:	<input type="range"/>	0.04
damping:	<input type="range"/>	0.07
avoidOverlap:	<input type="range"/>	0
maxVelocity:	<input type="range"/>	50
minVelocity:	<input type="range"/>	0.75

solver:

barnesHut ▼

timestep:

0.5

### wind

x:

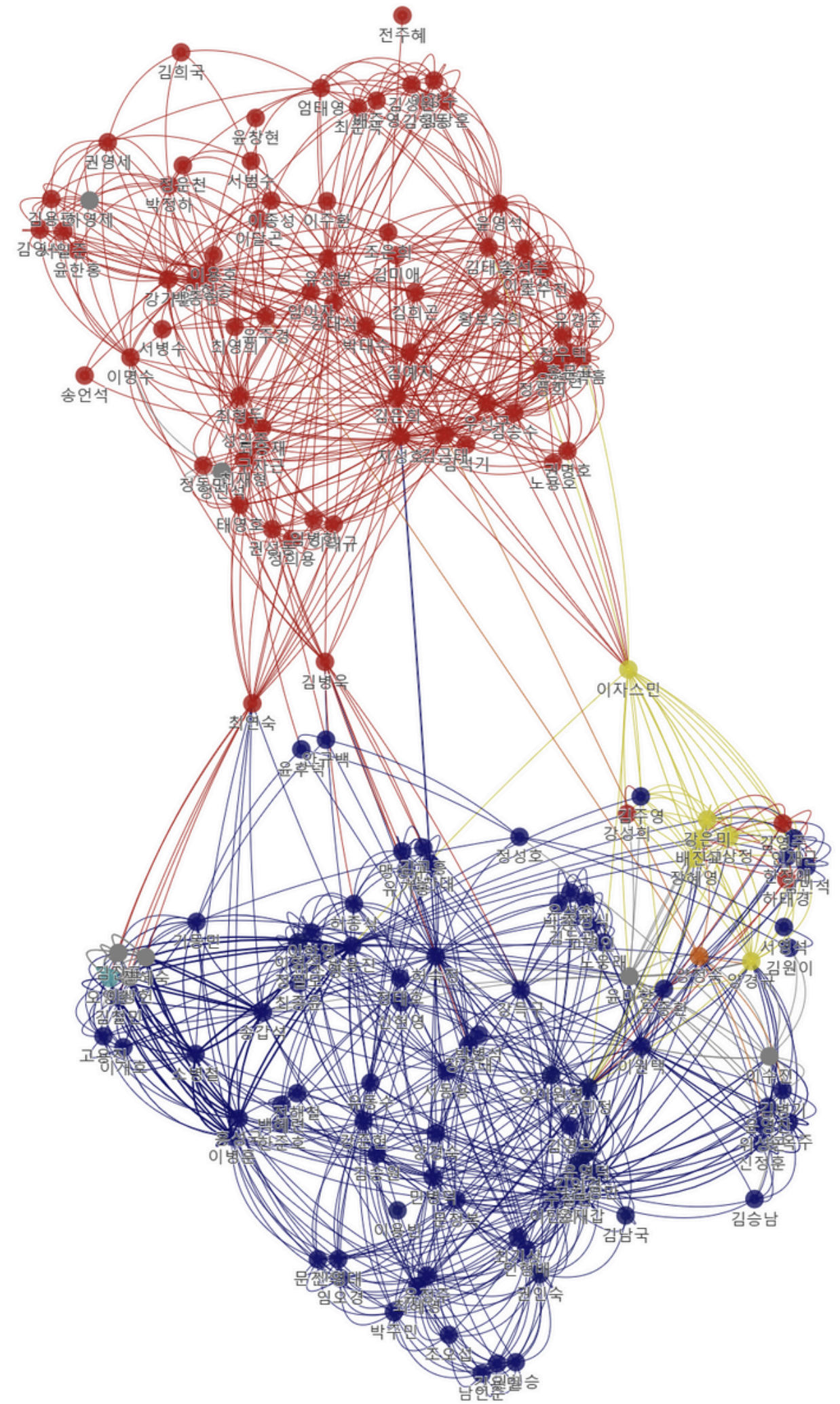
0

y:

0

# 최종 완성본

- 정당별 색상 추가
- 노드 크기 ↓, 노드 폰트 ↑
- 상호 인력(gravity) ↓, 상호 거리 ↑



# 결론

결론 및 의의 01

한계점과 발전 방향 02

# 결론 및 의의

정당의 막강한 영향력

정당 내 계파 존재 가능성



# 한계점과 발전 방향

동명이인 구분 어려움 → 데이터 자체의 한계  
(ex. 김병욱 의원, 이수진 의원)

**Large dataset** → 거시적으로 파악하기 어려움

시간의 흐름에 따른 변화 파악 → 국회 회기별로 나눠서 보기

다른 변수 활용 → 학연, 지연, 선거구, 상임위, 선수,  
회관 위치, 보좌진 변동 사항 등에 따른 변화 파악

**감사합니다**