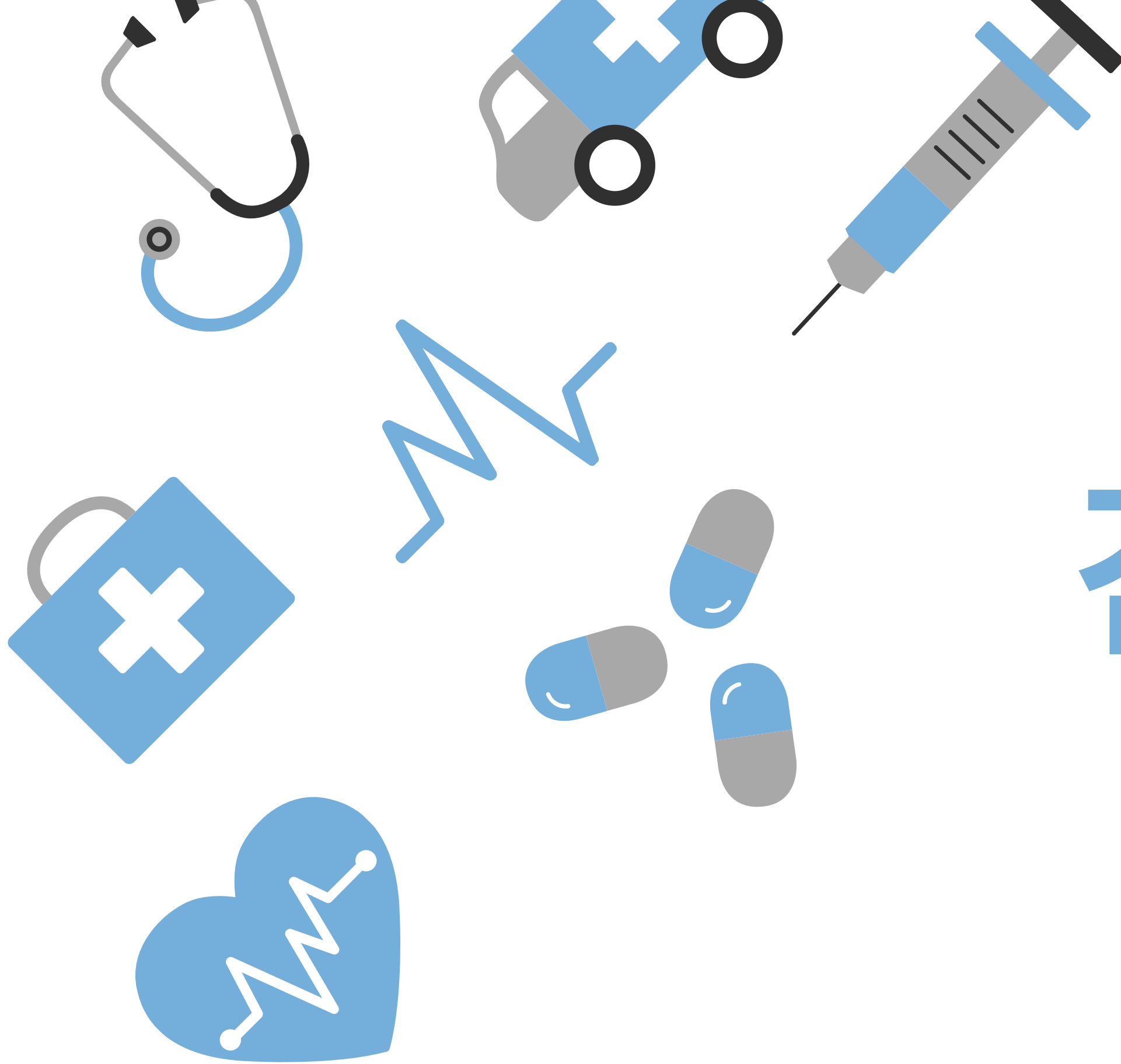


우리들의 건강한 삶을 위해



발표 목차

- 분석 배경
- 기획 목적
- 분석 내용
- 건강보험 분석
- 대구 구군별 의료시설 분석
- 대구와 다른 지역간 분석
- 직업군별 병원 방문 분석
- 결론 및 제언

분석 배경

뉴스를 보고...



2023.06.01 KBS 뉴스

‘응급실 뱅뱅이’ 이유는?...10명 중 3명은 ‘전문의 부재’



2024.01.15 매일신문

[2차종합병원 부족] 거꾸로 뒤집힌 대구 의료전달체계...



2023.06.09 경향신문

의사 부족에...전국 아동병원 71% “야간·휴일 단축 진료...”



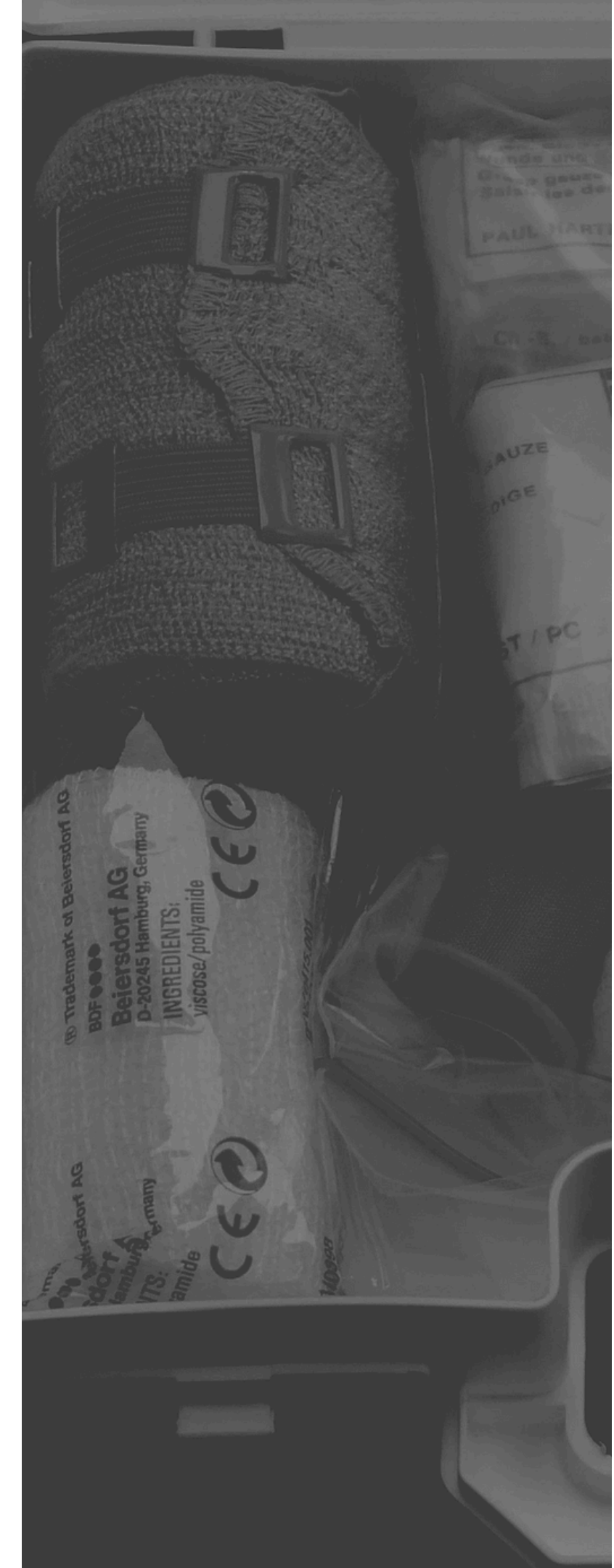
2024.09.02 머니투데이

소상공인 5명 중 4명 "일과 생활의 균형 중요"...건강·안전 '최우선'



2022.09.05 새누리신문

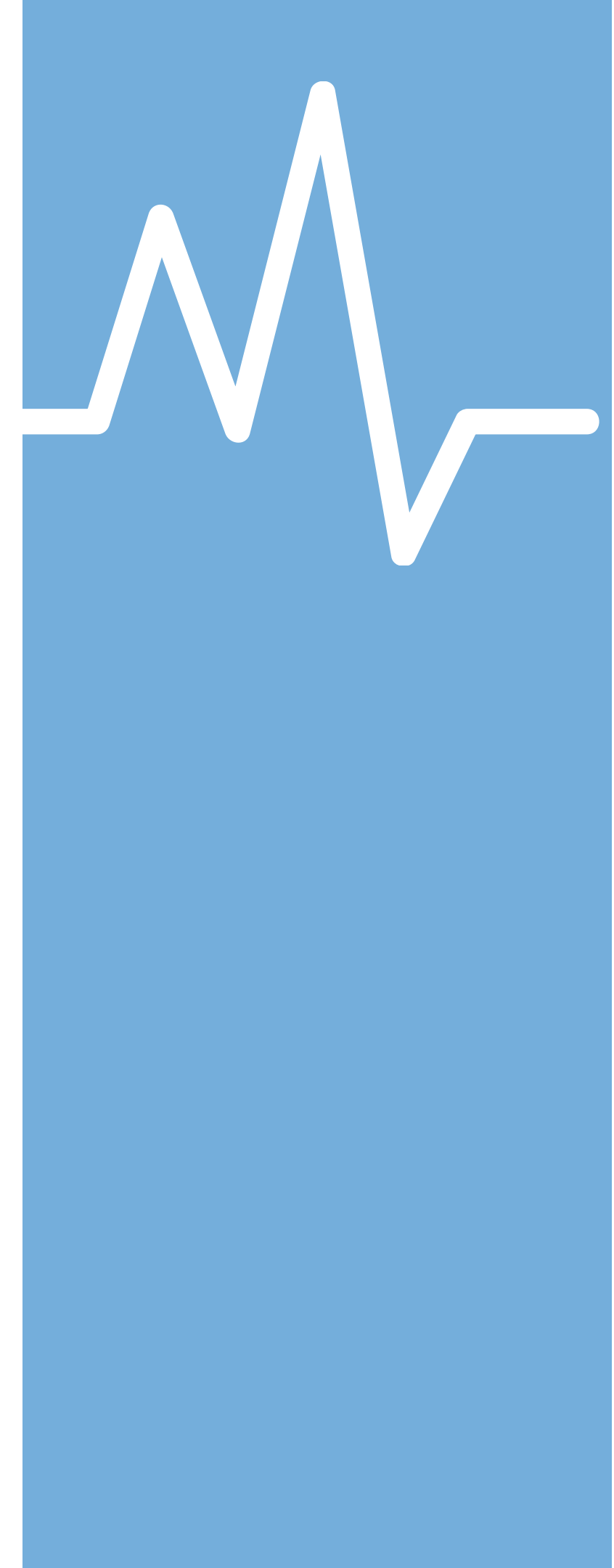
지역 간·지역 내 의료시설 불균형, 서울-대구 간 치료가능사망률...



기획 목적

본 기획은 다음과 같은 목적을 가지고 있습니다.

- 의료 서비스 접근성 분석 : 시도별 1000명당 의사 수를 비교하여 각 지역의 의료 서비스의 접근성을 평가 이를 통해 부족한 의료 인프라가 있는 지역을 식별하고 개선 방안을 모색
- 대구시 의료 서비스 현황 파악 : 대구시 구군별 다양한 의료 기관의 분포 및 병상 수를 분석하여 지역 내 의료 서비스의 불균형을 확인, 특히 종합병원과 소아과, 산부인과 같은 전문의료기관의 분포를 비교하여 주민들이 필요로 하는 의료 서비스의 종류와 양을 파악
- 병원 이용 패턴 분석 : 대구시 직업군별 병원 이용 요일 분석을 통해 주민들의 의료 서비스 이용 패턴을 이해하고, 이를 바탕으로 의료 서비스 제공 시간 및 운영 방식의 개선 방안을 제시
- 타지역 병원 이용 현황 조사 : 대구 주민들이 타지역 병원을 이용하는 현황을 분석하여 지역 내 의료 서비스의 부족함을 확인하고, 이에 따른 정책적 대응 방안을 마련



분석 내용

활용 데이터 및 출처

- Kosis
 - 물가상승률, 기간말 소비자물가지수
 - 인구, 가구 및 주택 - 읍면동(연도 끝자리 0, 5), 시군구(그 외 연도)
 - 시도별, 의료기관 유형별 근무의사 인력현황
- 대구통계
 - 주민등록 인구통계
 - 건강보험대상자_진료실적
- 공공데이터포털
 - 대구광역시_의료기관 현황
- 성인지통계시스템
 - 가구주의 혼인상태별 가구(성/시군구/연령별)
- 대구 빅데이터 활용센터(삼성카드)
 - 삼성 카드 데이터



분석 내용

데이터 불러오기 및 전처리

```
1 docc = pd.read_excel('/content/drive/MyDrive/3학년/데이터 시각화 분야_박진우_분석결과보고서 및 발표자료 제출/활용 데이터/시도별_의료기관_유형별_근무의사_인력현황_20240902215345.xlsx')
2 docc.drop('근무의사별(1)',axis = 1, inplace=True)
3 docc.columns = docc.iloc[0]
4 docc = docc.drop(0)
5 docc.rename(columns={'시도별(1)':'시도'},inplace=True)
6 docc
```

```
1 kp = pd.read_excel('/content/drive/MyDrive/3학년/데이터 시각화 분야_박진우_분석결과보고서 및 발표자료 제출/활용 데이터/인구_가구_및_주택_-_읍면동_연도_결자리_0_5_시군구_그_외_연도_20240902205752.xlsx')
2 kp.rename(columns={'행정구역별(읍면동)':'시도'},inplace=True)
3 kp = kp.drop(0)
4 kp
```

```
1 insurance = pd.read_excel('/content/drive/MyDrive/3학년/데이터 시각화 분야_박진우_분석결과보고서 및 발표자료 제출/활용 데이터/건강보험대상자_진료실적_20240903193723.xlsx')
2 insurance = insurance.replace({' ':' '}, regex=True)
3 insurance = insurance.apply(pd.to_numeric, errors='ignore')
```

```
1 price = pd.read_excel('/content/drive/MyDrive/3학년/데이터 시각화 분야_박진우_분석결과보고서 및 발표자료 제출/활용 데이터/물가상승률_소비자물가지수_20240904133820.xlsx')
2 price = price.drop(price.index[0], axis=0)
3 price = price.drop('국가', axis=1)
4 price = price.astype('float')
5 price = price.T
6 price.columns = ['물가상승률']
7 price.index.name = '연도'
```

```
1 # 대구시 인구데이터 불러오기
2 p = pd.read_excel('/content/drive/MyDrive/3학년/데이터 시각화 분야_박진우_분석결과보고서 및 발표자료 제출/활용 데이터/대구시군구별인구.xlsx')
3 p.rename(columns={'행정기관':'구군'},inplace=True)
```

```
1 with open('/content/drive/MyDrive/3학년/데이터 시각화 분야_박진우_분석결과보고서 및 발표자료 제출/활용 데이터/대구광역시_의료기관 현황_20240430.csv', 'rb') as f:
2     result = chardet.detect(f.read())
3     print(result['encoding'])
4
5 # 감지된 인코딩으로 파일 읽기
6 hloc = pd.read_csv('/content/drive/MyDrive/3학년/데이터 시각화 분야_박진우_분석결과보고서 및 발표자료 제출/활용 데이터/대구광역시_의료기관 현황_20240430.csv', encoding=result['encoding'])
7 hloc = hloc.drop('연번', axis=1)
8 hloc['구군'] = hloc['구군'].str.replace(r'^\d+##$', '', regex=True)
```

```
1 # 각 그룹에 따른 데이터프레임 나누기 및 인덱스 초기화
2 한방병원_한의원 = hloc[hloc['종별'].isin(['한방병원', '한의원']),reset_index(drop=True)]
3 정신병원 = hloc[hloc['종별'] == '정신병원',reset_index(drop=True)]
4 의원 = hloc[hloc['종별'] == '의원'],reset_index(drop=True)
5 종합병원_병원_부속의원_상급종합병원 = hloc[hloc['종별'].isin(['종합병원', '병원', '부속의원', '상급종합병원']),reset_index(drop=True)]
6 요양병원 = hloc[hloc['종별'] == '요양병원'],reset_index(drop=True)
7 치과 = hloc[hloc['종별'].isin(['치과의원', '치과병원']),reset_index(drop=True)]
```



분석 내용

데이터 불러오기 및 전처리

```
1 # 특정 단어 리스트
2 keywords = [
3     '내과', '외과', '소아과', '소아', '아동', '아이',
4     '이비인후과', '안과', '치과', '산부인과',
5     '정형외과', '마취', '피부', '재활', '가정의학', '여성', '정신', '통증', '비뇨', '신경', '영상', '연합', '한의원'
6 ]
7
8 # 포함된 병원 이름 기준으로 데이터프레임 나누기
9 포함된_병원 = 의원[의원['의료기관명'].str.contains('|'.join(keywords), na=False)].reset_index(drop=True)
10 일반의원 = 의원[~의원['의료기관명'].str.contains('|'.join(keywords), na=False)].reset_index(drop=True)
```

```
1 내과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('내과', na=False)].reset_index(drop=True)
2 외과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('외과', na=False)].reset_index(drop=True)
3 소아과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('|'.join(['소아과', '아동', '아이', '소아']), na=False)].reset_index(drop=True)
4 이비인후과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('이비인후과', na=False)].reset_index(drop=True)
5 안과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('안과', na=False)].reset_index(drop=True)
6 산부인과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('|'.join(['산부', '여성']), na=False)].reset_index(drop=True)
7 정형외과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('정형외과', na=False)].reset_index(drop=True)
8 마취통증의학과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('|'.join(['마취', '통증']), na=False)].reset_index(drop=True)
9 피부과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('피부', na=False)].reset_index(drop=True)
10 재활의학과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('재활', na=False)].reset_index(drop=True)
11 가정의학과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('가정의학', na=False)].reset_index(drop=True)
12 신경외과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('신경', na=False)].reset_index(drop=True)
13 정신과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('정신', na=False)].reset_index(drop=True)
14 비뇨기과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('비뇨', na=False)].reset_index(drop=True)
15 영상의학과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('영상', na=False)].reset_index(drop=True)
16 연합의원 = 포함된_병원[포함된_병원['의료기관명'].str.contains('연합', na=False)].reset_index(drop=True)
17 새로운_치과 = 포함된_병원[포함된_병원['의료기관명'].str.contains('|'.join(['치과의원', '치과병원']), na=False)].reset_index(drop=True)
18 새로운_한의원 = 포함된_병원[포함된_병원['의료기관명'].str.contains('한의원', na=False)].reset_index(drop=True)
19 # 기존 치과 데이터프레임에 새로운 데이터 추가
20
21 한방병원_한의원 = pd.concat([한방병원_한의원, 새로운_한의원], ignore_index=True)
22 정신병원_정신과 = pd.concat([정신병원, 정신과], ignore_index=True)
23 치과 = pd.concat([치과, 새로운_치과], ignore_index=True)
```



분석 내용

데이터 불러오기 및 전처리

```
1 wp = pd.read_excel('/content/drive/MyDrive/3학년/데이터 시각화 분야_박진우_분석결과보고서 및 발표자료 제출/활용 데이터/대구시군구별여자인구.xlsx')
2 wp.rename(columns={'행정기관':'구군'}, inplace=True)
3 wp = wp.drop(['여 인구수', '연령구간인구수'], axis=1)
4 wp.drop(wp.index[0], axis=0, inplace=True)
5
```

```
1 # 혼인 여성데이터 불러오기
2 marry = pd.read_excel('/content/drive/MyDrive/3학년/데이터 시각화 분야_박진우_분석결과보고서 및 발표자료 제출/활용 데이터/가구의_혼인상태별_가구_성_시군구_연령별__20240827170100.xlsx')
3 # 멀티 인덱스를 제거하고 첫 번째 행을 열 이름으로 설정
4 marry.columns = marry.iloc[1] # 첫 번째 행을 열 이름으로 사용
5 marry = marry.drop(0) # 첫 번째 행 제거
6 marry = marry.drop(1) # 첫 번째 행 제거
7 marry = marry.drop(2) # 첫 번째 행 제거
8 marry = marry.drop(3) # 첫 번째 행 제거
9 marry = marry.drop('전체', axis=1)
10
11 # 열 이름을 하나의 레벨로 단순화
12 marry.columns.name = None # 컬럼 이름의 인덱스 이름 제거
13 marry.reset_index(drop = True, inplace=True)
14
15 # 데이터 합치기
16 marry['20~29세'] = marry[['20~24세', '25~29세']].sum(axis=1)
17 marry['30~39세'] = marry[['30~34세', '35~39세']].sum(axis=1)
18 marry['40~49세'] = marry[['40~44세', '45~49세']].sum(axis=1)
19 marry['50~59세'] = marry[['50~54세', '55~59세']].sum(axis=1)
20 marry['60~69세'] = marry[['60~64세', '65~69세']].sum(axis=1)
21 marry['70~79세'] = marry[['70~74세', '75~79세']].sum(axis=1)
22 marry['80대 이상'] = marry[['80~84세', '85세 이상']].sum(axis=1)
23
24 # 기존 열 제거
25 marry.drop(columns=['20~24세', '25~29세', '30~34세', '35~39세', '40~44세', '45~49세',
26 | '50~54세', '55~59세', '60~64세', '65~69세', '70~74세', '75~79세',
27 | '80~84세', '85세 이상'], inplace=True)
28
29 for col in marry.columns[1:]: # '행정기관'을 제외한 나머지 열에 대해
30 | marry[col] = marry[col].replace(',', '', regex=True).astype(int)
31
32 marry.rename(columns={'행정구역별':'구군'}, inplace=True)
```

```
1 obgyn_count = 산부인과.groupby('구군').size().reset_index(name='산부인과 수')
2 obgyn_count
```



분석 내용

데이터 불러오기 및 전처리

```
1 # 공백 제거 및 소문자로 변환
2 marry['구군'] = marry['구군'].str.strip().str.lower()
3 산부인과['구군'] = 산부인과['구군'].str.strip().str.lower()
4 # 산부인과 수 계산
5 obgyn_count = 산부인과.groupby('구군').size().reset_index(name='산부인과 수')
6 # 병합 후 NaN을 0으로 채우기
7 merged_df = pd.merge(marry, obgyn_count, on='구군', how='left')
8 merged_df['산부인과 수'] = merged_df['산부인과 수'].fillna(0)
9 print(merged_df[['구군', '산부인과 수']])
```

```
1 df = df.replace({'WEEK_GROUP': 'A'}, '월요일')
2 df = df.replace({'WEEK_GROUP': 'B'}, '화요일')
3 df = df.replace({'WEEK_GROUP': 'C'}, '수요일')
4 df = df.replace({'WEEK_GROUP': 'D'}, '목요일')
5 df = df.replace({'WEEK_GROUP': 'E'}, '금요일')
6 df = df.replace({'WEEK_GROUP': 'F'}, '토요일')
7 df = df.replace({'WEEK_GROUP': 'G'}, '일요일')
8 df = df.replace({'WEEKEND_GROUP': 'A'}, '평일')
9 df = df.replace({'WEEKEND_GROUP': 'B'}, '주말/공휴일')
```

```
1 inner_p = df[df['내외지인'] == 1]
2 nearby_p = df[df['내외지인'] == 2]
3 outsiders = df[df['내외지인'] == 3]
```

```
1 hp = df[df['업종명_중분류'].isin(['일반병원', '특화병원'])]
2 hp
3 pd.set_option('display.max_rows', None)
```

```
1 job_weekdata_analysis = hospital_data.groupby(['직업군', 'WEEK_GROUP', '업종명_중분류']).agg({
2     'CNT': 'sum',
3     'AMT': 'sum'}).reset_index()
4 pd.set_option('display.max_rows', None)
5 job_weekdata_analysis
```



분석 내용

데이터 불러오기 및 전처리

```
1 job_weekdata_analysis['평균_이용건수'] = job_weekdata_analysis['CNT'] / job_weekdata_analysis.groupby(['직업군', 'WEEK_GROUP'])['CNT'].transform('count')
2 job_weekdata_analysis['평균_이용금액'] = job_weekdata_analysis['AMT'] / job_weekdata_analysis['CNT']
3
4 def format_number(x):
5     if pd.isna(x):
6         return x
7     elif isinstance(x, (int, np.integer)):
8         return f"{x:,}"
9     elif isinstance(x, float):
10        return f"{x:,.2f}"
11    else:
12        return x
13 pd.set_option('display.max_rows', None)
14 numeric_columns = ['CNT', 'AMT', '평균_이용건수', '평균_이용금액']
15 for col in numeric_columns:
16    job_weekdata_analysis[col] = job_weekdata_analysis[col].apply(format_number)
```

```
1 population_df = p
2 hospital_df = 소아과
3
4 # 병원 수 세기
5 hospital_count = hospital_df.groupby('구군')['의료기관명'].count().reset_index(name='병원수')
6 population_df['구군'] = population_df['구군'].str.strip()
7 # 데이터 병합
8 merged_df = pd.merge(population_df, hospital_count, on='구군', how='left')
```

```
1 for col in p.columns[2:]: # '행정기관'을 제외한 나머지 열에 대해
2     p[col] = p[col].replace(',', '', regex=True).astype(int)
3 p.info()
```

숨겨진 출력 표시

```
1 p['구군'] = p['구군'].str.strip().str.lower()
2 소아과['구군'] = 소아과['구군'].str.strip().str.lower()
3
4 obgyn_count = 소아과.groupby('구군').size().reset_index(name='소아과 수')
5
6 merged_df = pd.merge(p, obgyn_count, on='구군', how='left')
7 merged_df['소아과 수'] = merged_df['소아과 수'].fillna(0)
8 print(merged_df[['구군', '소아과 수']])
```

```
1 p['80세 이상'] = p['80~89세'] + p['90~99세'] + p['100세 이상']
2
3 p = p.drop(columns=['80~89세', '90~99세', '100세 이상'])
4
5 population_df = p
6 hospital_df = 요양병원
7
8 hospital_count = hospital_df.groupby('구군')['의료기관명'].count().reset_index(name='병원수')
9 population_df['구군'] = population_df['구군'].str.strip()
10
11 merged_df = pd.merge(population_df, hospital_count, on='구군', how='left')
12 merged_df['총 인구수'] = merged_df['총 인구수'].str.replace(',', '').astype(int)
13 merged_df['병원 비율'] = merged_df['병원수'] / merged_df['총 인구수'] * 10000
14
15 filtered_df = merged_df[merged_df['병원 비율'] > 0]
16
17 obgyn_count = 요양병원.groupby('구군').size().reset_index(name='요양병원 수')
18
```

```
1 # 공백 제거 및 소문자로 변환
2 p['구군'] = p['구군'].str.strip().str.lower()
3 요양병원['구군'] = 요양병원['구군'].str.strip().str.lower()
4 # 산부인과 수 계산
5 obgyn_count = 요양병원.groupby('구군').size().reset_index(name='요양병원 수')
6 # 병합 후 NaN을 0으로 채우기
7 merged_df = pd.merge(p, obgyn_count, on='구군', how='left')
8 merged_df['요양병원 수'] = merged_df['요양병원 수'].fillna(0)
```



분석 내용

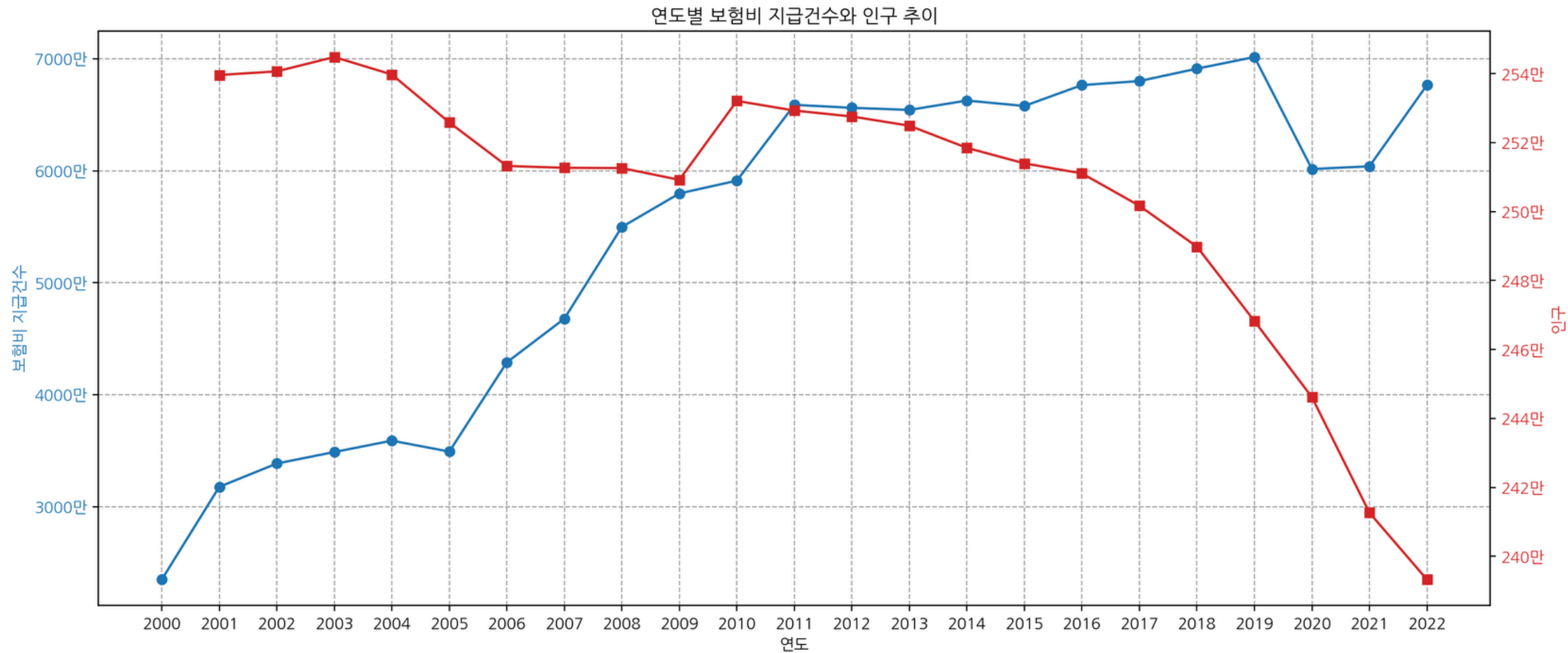
분석도구, 분석기법

- 분석도구 : Python
- 사용 모듈
 - pandas
 - numpy
 - seaborn
 - matplotlib.pyplot
 - koreanize_matplotlib
 - matplotlib.ticker.FuncFormatter
 - chardet



건강보험 분석

연도별 보험비 지급건수



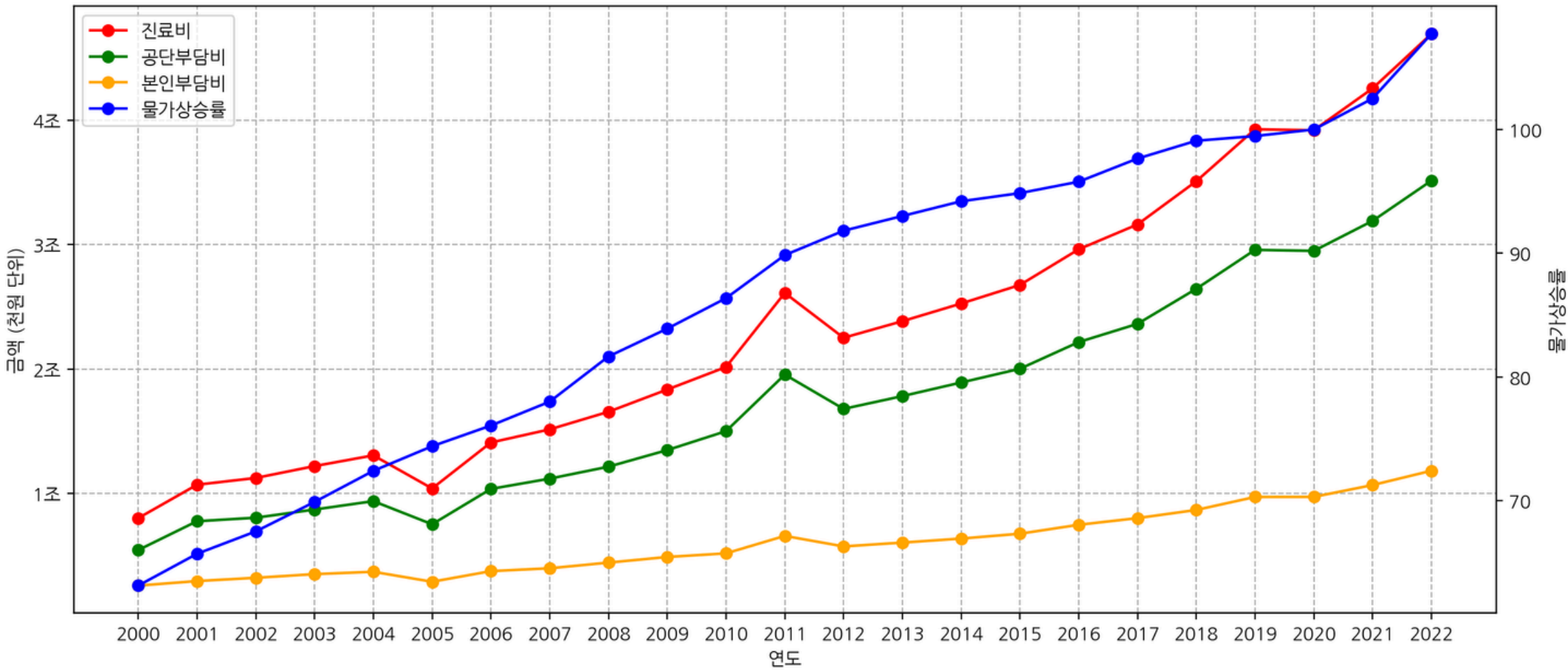
```
1 columns_to_plot = ['지급건수 (건)', '진료비 (천원)', '공단부담비 (천원)', '본인부담비 (천원)']
2 years = insurance.columns.str.extract(r'(\d{4})')[0].dropna().unique()
3 yearly_data = insurance.loc[1, insurance.columns.str.contains('|'.join(years))]
4
5 data_dict = {}
6 for year in years:
7     data_dict[year] = yearly_data.filter(like=year).values
8
9 df_insurance = pd.DataFrame(data_dict, index=columns_to_plot),T
10
11 df_pop = pop.set_index('인구현황별 (1)')
12 df_pop = df_pop.iloc[:, 1:]
13 df_pop = df_pop.apply(pd.to_numeric, errors='coerce')
14
15 df_combined = pd.DataFrame({
16     '지급건수 (건)': df_insurance['지급건수 (건)'],
17     '인구': df_pop.loc['등록인구 (명)']
18 })
19
20 def ten_thousands_formatter(x, pos):
21     return f'{int(x / 10000)}만' if x != 0 else '0'
22
23 fig, ax1 = plt.subplots(figsize=(14, 6))
24
25 color = 'tab:blue'
26 ax1.set_xlabel('연도')
27 ax1.set_ylabel('보험비 지급건수', color=color)
28 ax1.plot(df_combined.index, df_combined['지급건수 (건)'], color=color, marker='o')
29 ax1.tick_params(axis='y', labelcolor=color)
30 ax1.yaxis.set_major_formatter(FuncFormatter(ten_thousands_formatter))
31
32 ax2 = ax1.twinx()
33 color = 'tab:red'
34 ax2.set_ylabel('인구', color=color)
35 ax2.plot(df_combined.index, df_combined['인구'], color=color, marker='s')
36 ax2.tick_params(axis='y', labelcolor=color)
37 ax2.yaxis.set_major_formatter(FuncFormatter(ten_thousands_formatter))
38
39 plt.title('대구시 연도별 보험비 지급건수와 인구 추이')
40
41 ax1.grid(True, which='both', axis='both', linestyle='--', color='gray', alpha=0.7)
42
43 ax1.set_axisbelow(True)
44
45 fig.tight_layout()
46 plt.show()
```

- 인구는 감소하지만 보험비 지급건수는 증가하는 추세
- 현재의 추세가 지속된다면, 인구당 보험비 지급건수는 계속 증가할 것으로 예상 이에 대한 대비책이 필요

건강보험 분석

연도별 진료비, 공단부담비, 본인부담비, 물가상승률

연도별 진료비, 공단부담비, 본인부담비 및 물가상승률

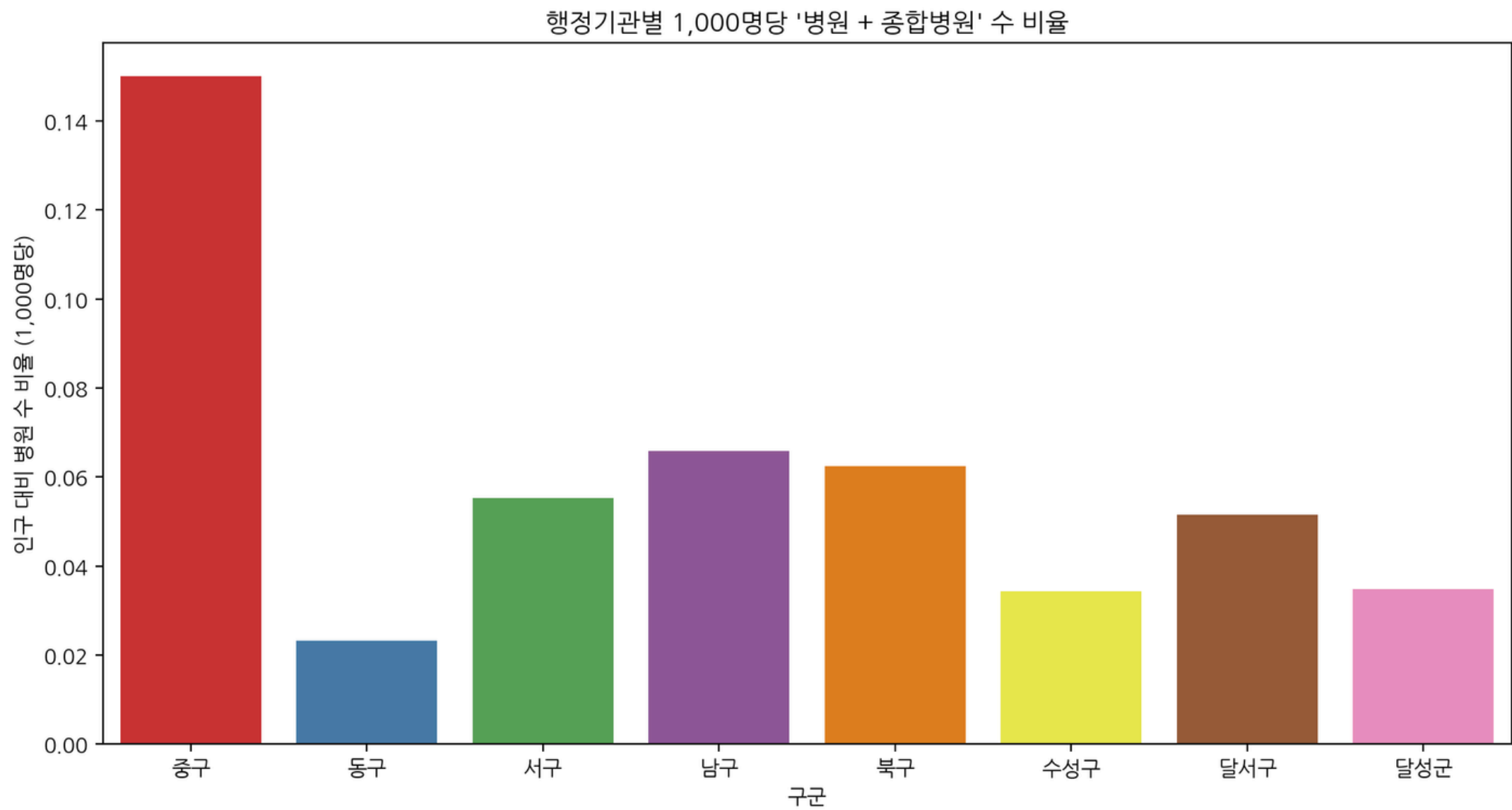


```
1 columns_to_plot = ['지급건수 (건)', '진료비 (천원)', '공단부담비 (천원)', '본인부담비 (천원)']
2 years = insurance.columns.str.extract(r'(\d{4})')[0].dropna().unique()
3 yearly_data = insurance.loc[1, insurance.columns.str.contains('|'.join(years))]
4
5 data_dict = {}
6 for year in years:
7     data_dict[year] = yearly_data.filter(like=year).values
8
9 df = pd.DataFrame(data_dict, index=columns_to_plot, T
10
11 df[['진료비 (천원)', '공단부담비 (천원)', '본인부담비 (천원)']] /= 10
12
13
14 def billion_formatter(x, pos):
15     if x == 0:
16         return '0'
17     elif x % 1e8 == 0:
18         return f'{int(x / 1e8):,}조'
19     else:
20         return f'{x / 1e8:.1f}조'
21
22 fig, ax1 = plt.subplots(figsize=(14, 6))
23
24 line1, = ax1.plot(df.index, df['진료비 (천원)'], marker='o', color='red', label='진료비')
25 line2, = ax1.plot(df.index, df['공단부담비 (천원)'], marker='o', color='green', label='공단부담비')
26 line3, = ax1.plot(df.index, df['본인부담비 (천원)'], marker='o', color='orange', label='본인부담비')
27 ax1.set_xlabel('연도')
28 ax1.set_ylabel('금액')
29 ax1.yaxis.set_major_formatter(FuncFormatter(billion_formatter))
30 ax1.grid(True, linestyle='--')
31
32 ax2 = ax1.twinx()
33 line4, = ax2.plot(price.index, price['물가상승률'], marker='o', linestyle='-', color='blue', label='물가상승률')
34 ax2.set_ylabel('물가상승률')
35
36 lines = [line1, line2, line3, line4]
37 labels = [line.get_label() for line in lines]
38 ax1.legend(lines, labels, loc='upper left')
39
40 fig.suptitle('연도별 진료비, 공단부담비, 본인부담비 및 물가상승률')
41 plt.show()
```

- 본인부담비는 완만하게 증가하지만 진료비와 공단부담비는 가파르게 증가

대구 구군별 의료시설 분석

종합병원(병원)

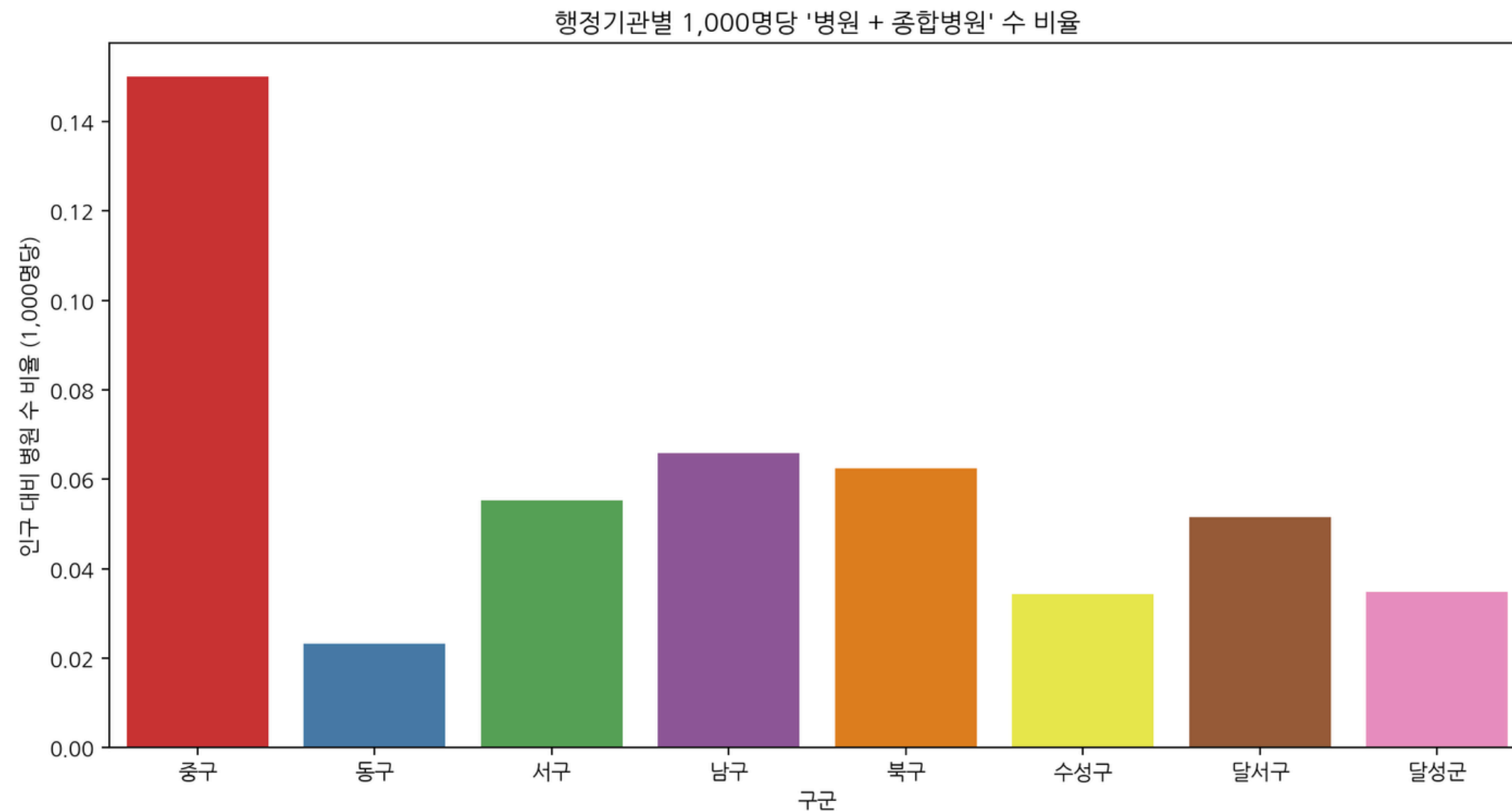


구군	총인구수	병원수	"병원 비율 (1,000명)"
중구	93,338	14	0.15
동구	343,392	8	0.02
서구	162,903	9	0.06
남구	136,582	9	0.07
북구	416,109	26	0.06
수성구	408,215	14	0.03
달서구	524,448	27	0.05
달성군	259,024	9	0.03
군위군	22,649	0	0

- 중구를 제외한 7개의 구군 모두 1,000명당 종합병원 비율이 중구의 절반에 못미침
- 특히 군위군에는 종합병원이나 병원의 수가 하나도 없어
- 지역간 의료 불균형이 심각함

대구 구군별 의료시설 분석

종합병원(병원)



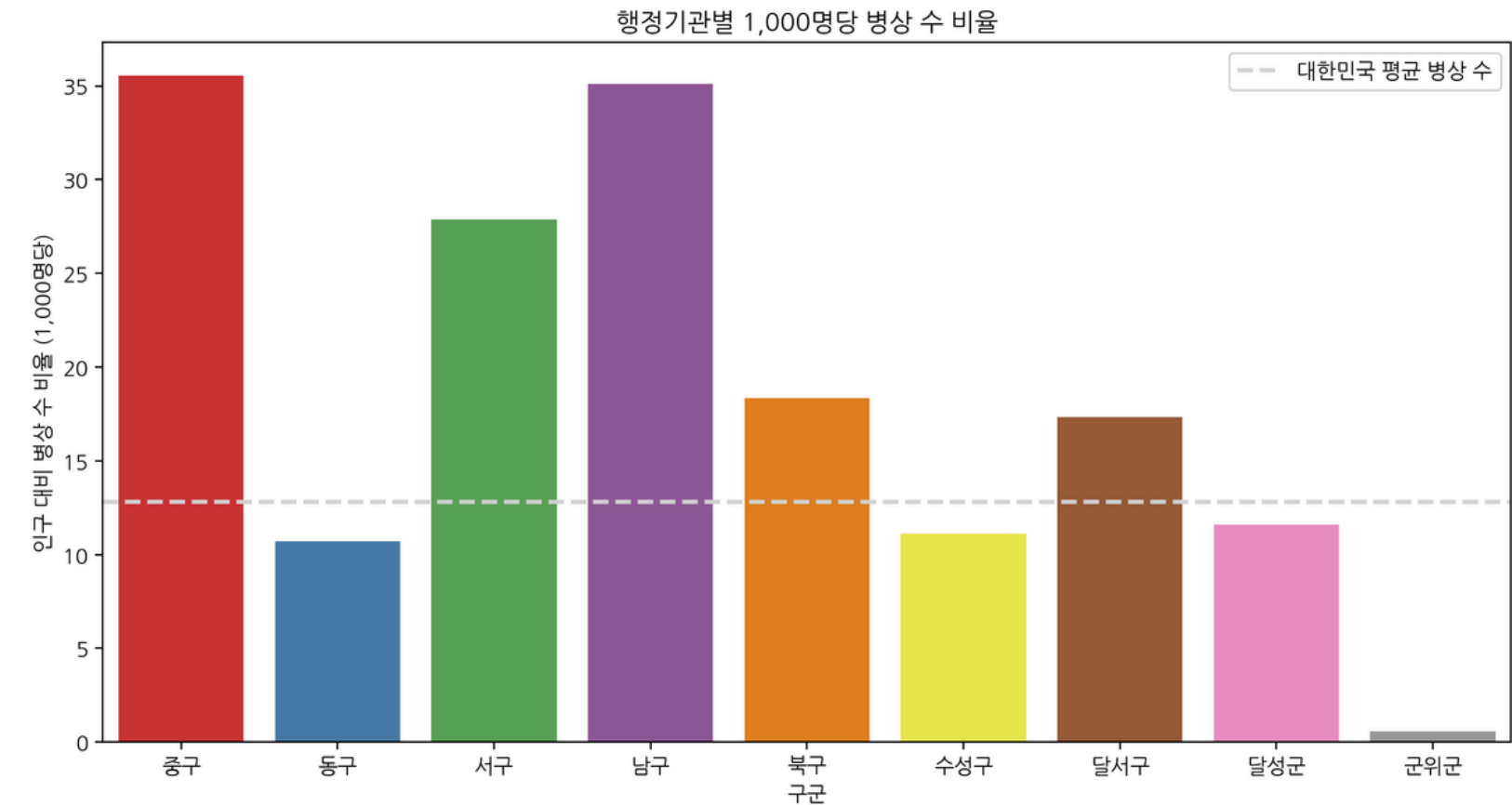
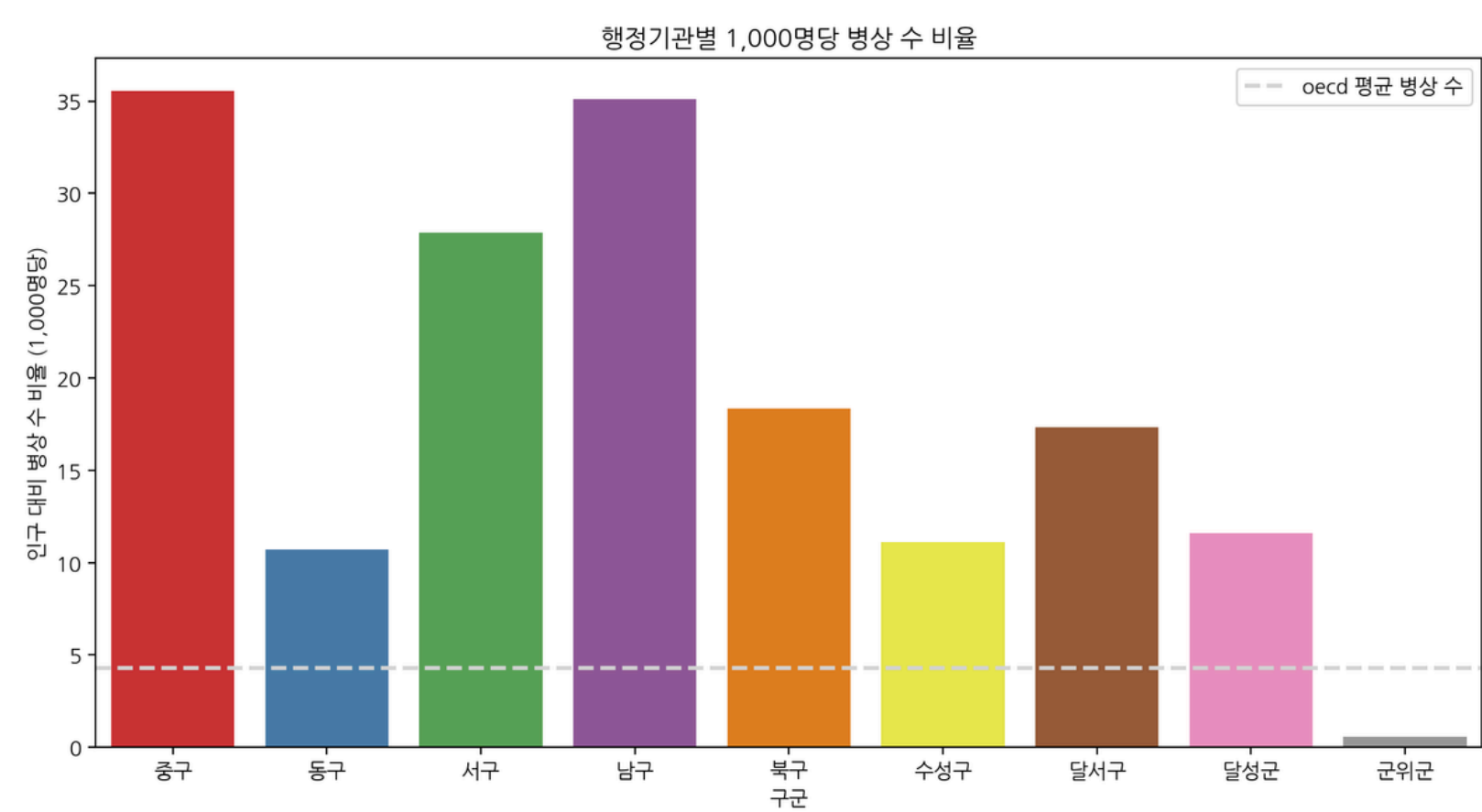
- 중구를 제외한 7개의 구군 모두 1,000명당 종합병원 비율이 중구의 절반에 못미침
- 특히 군위군에는 종합병원이나 병원의 수가 하나도 없어
- 지역간 의료 불균형이 심각함

```
1 population_df = p
2 hospital_df = 종합병원_병원_부속의원_상급종합병원
3
4 # 병원 수 세기
5 hospital_count = hospital_df.groupby('구군')['의료기관명'].count().reset_index(name='병원수')
6 population_df['구군'] = population_df['구군'].str.strip()
7 # 데이터 병합
8 merged_df = pd.merge(population_df, hospital_count, on='구군', how='left')

1 merged_df['총 인구수'] = merged_df['총 인구수'].str.replace(',','',).astype(int)
2 merged_df['병원_비율'] = merged_df['병원수'] / merged_df['총 인구수'] * 10000
3
4 filtered_df = merged_df[merged_df['병원_비율'] > 0]
5
6 # 시각화
7 plt.figure(figsize=(12, 6))
8 sns.barplot(data=filtered_df, x='구군', y='병원_비율', hue='구군', palette='Set1')
9 plt.xlabel('구군')
10 plt.ylabel('인구 대비 병원 수 비율 (1,000명당)')
11 plt.title('행정기관별 1,000명당 병원 + 종합병원 수 비율')
12 plt.xticks(rotation=0)
13
14 plt.show()
```

대구 구군별 의료시설 분석

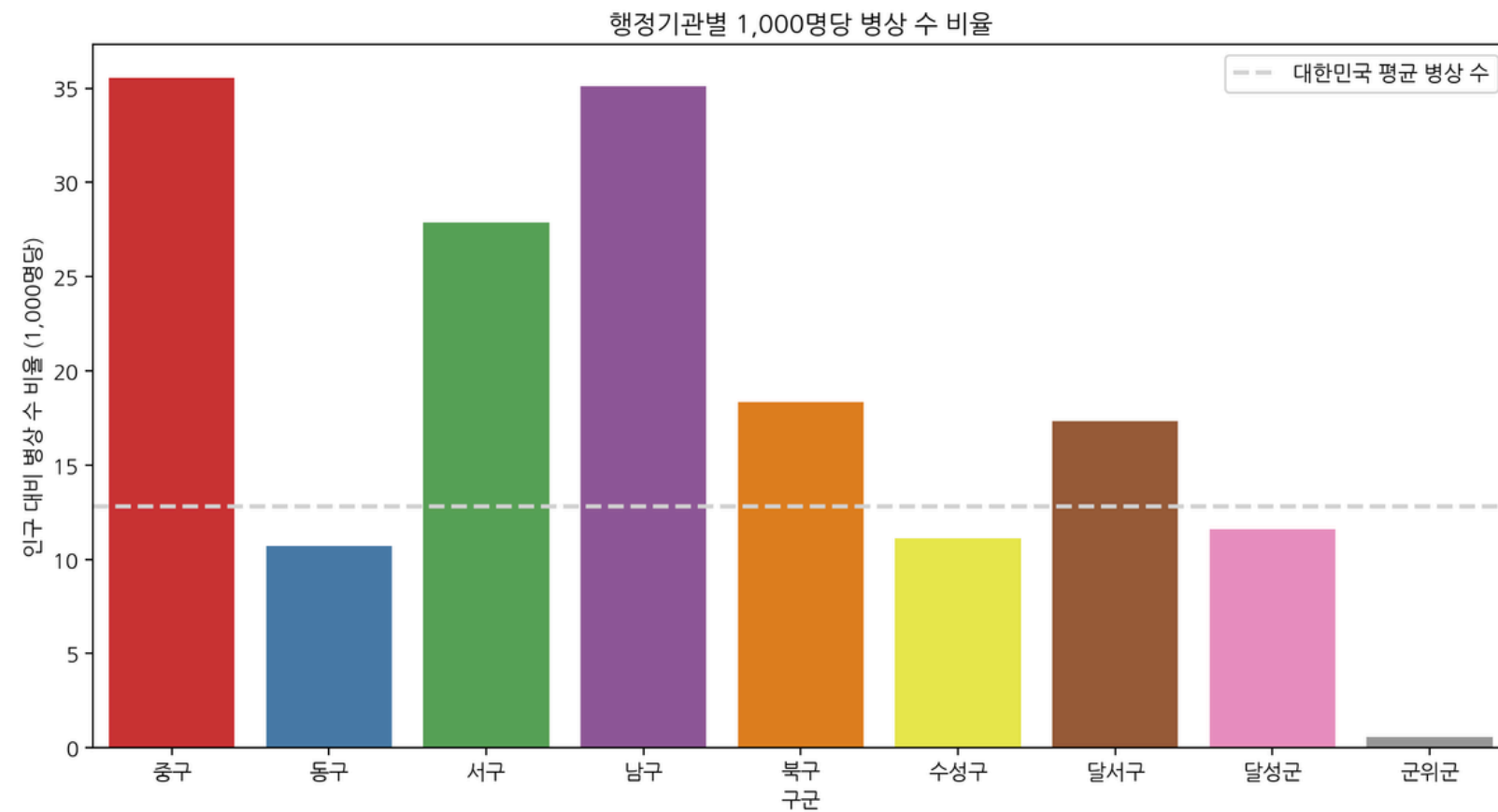
구군별 병상 수 비율



- 군위군을 제외한 7개의 구군 모두 oecd 평균 병상 수를 넘김
- 하지만 대한민국 평균 병상 수와 비교하면 동구, 수성구, 달성군, 군위군은 대한민국의 평균 병상 수에 못미침
- 대한민국 평균 병상 수에 못 미치는 구군에 병상 수 증가가 필요

대구 구군별 의료시설 분석

구군별 병상 수 비율



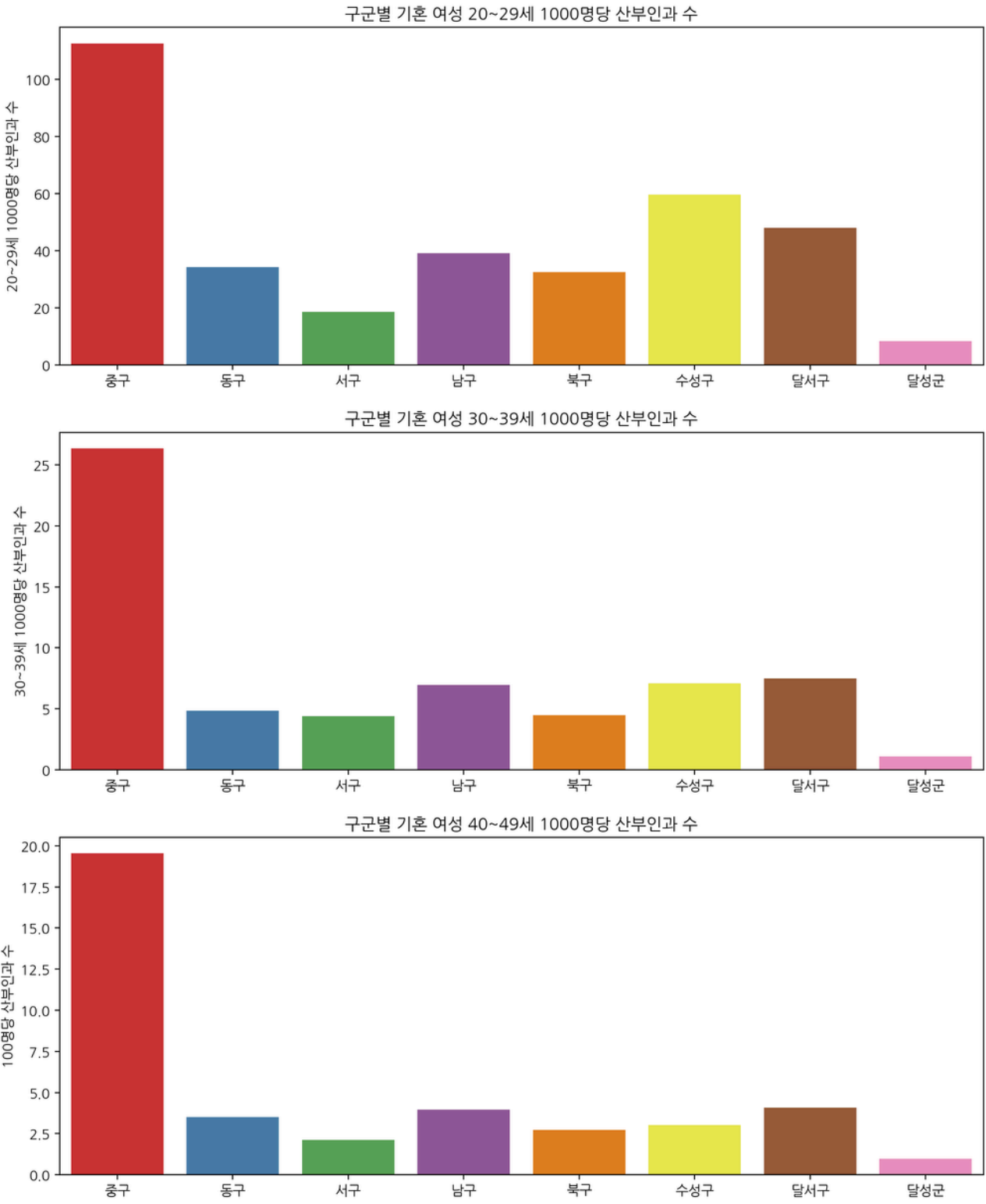
```
1 population_df = p
2
3 # 병상 수 합계 구하기
4 bed_count = hloc.groupby('구군')['병상수 합계'].sum().reset_index(name='병상수 합계')
5
6 # 인구 데이터와 병상 수 데이터 병합
7 merged_bed_df = pd.merge(population_df, bed_count, on='구군', how='left')
8
9 # 1000명당 병상 수 비율 계산
10 merged_bed_df['총 인구수'] = merged_bed_df['총 인구수'].str.replace(',', '').astype(int)
11 merged_bed_df['병상_비율'] = merged_bed_df['병상수 합계'] / merged_bed_df['총 인구수'] * 1000
12
13 # 병상 수 비율이 0 이상인 데이터만 필터링
14 filtered_bed_df = merged_bed_df[merged_bed_df['병상_비율'] > 0]
15 plt.figure(figsize=(12, 6))
16 sns.barplot(data=filtered_bed_df, x='구군', y='병상_비율', hue='구군', palette='Set1')
17 plt.axhline(12.8, color='lightgray', linestyle='--', linewidth=2, label='대한민국 평균 병상 수')
18
19 plt.xlabel('구군')
20 plt.ylabel('인구 대비 병상 수 비율 (1,000명당)')
21 plt.title('행정기관별 1,000명당 병상 수 비율')
22 plt.xticks(rotation=0)
23 plt.legend()
24
25 plt.show()
```

- 군위군을 제외한 7개의 구군 모두 oecd 평균 병상 수를 넘김
- 하지만 대한민국 평균 병상 수와 비교하면 동구, 수성구, 달성군, 군위군은 대한민국의 평균 병상 수에 못미침
- 대한민국 평균 병상 수에 못 미치는 구군에 병상 수 증가가 필요

대구 구군별 의료시설 분석

*첫 아이를 출산하는 평균 연령은 33.0세,
둘째 아이는 34.2세, 셋째 아이는 35.6세

산부인과의원



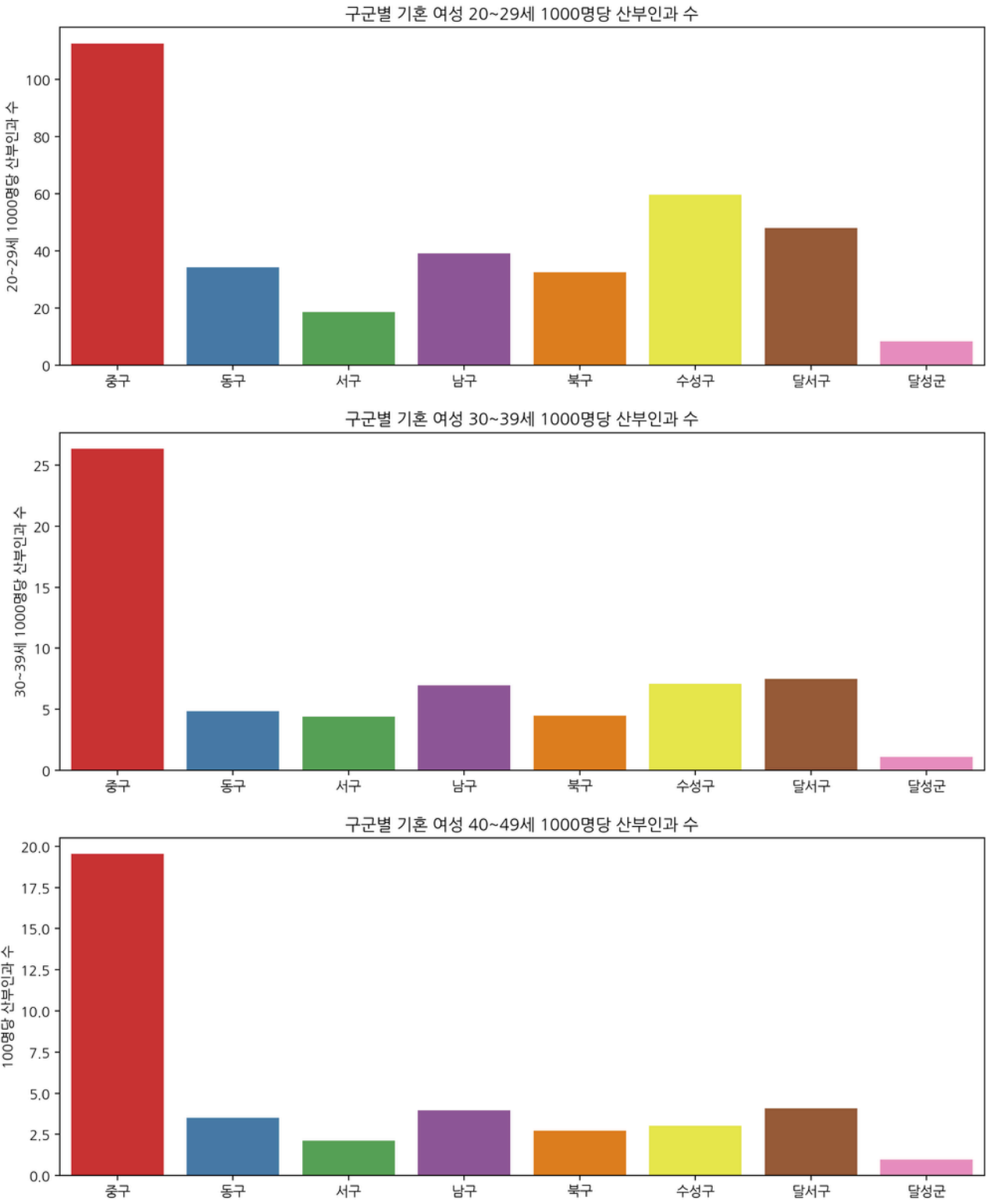
구군	20대 기혼 여성	30대 기혼 여성	40대 기혼 여성	병원수	병원 비율(1,000명) 20대/30대/40대
중구	142	607	819	16	112.67/26.36/19.54
동구	351	2,470	3,412	12	34.19/4.86/3.52
서구	162	682	1,409	3	18.51/4.4/2.12
남구	128	718	1,260	5	39.06/6.96/4
북구	430	3,117	5,165	14	32.55/4.49/2.71
수성구	335	2,830	6,577	20	59.7/7.06/3.04
달서구	542	3,470	6,383	26	47.97/7.49/4.07
달성군	364	2,682	3,291	3	8.24/1.11/0.97
구군	20대 여성	30대 여성	40대 여성	병원수	병원 비율(1,000명) 20대/30대/40대
군위군	390	491	722	0	0

- 가임기 기혼여성이 2번째로 많은 북구가 군위군을 제외한 대구시 구군중 가임기 기혼 여성이 가장 적은 중구보다 산부인과 수가 적은 모습을 보이며 전체적인 비율또한 대다수의 구군이 중구의 1/3의 모습을 보이는데 이또한 지역간 의료 불균형을 보임

대구 구군별 의료시설 분석

*첫 아이를 출산하는 평균 연령은 33.0세,
둘째 아이는 34.2세, 셋째 아이는 35.6세

산부인과의원

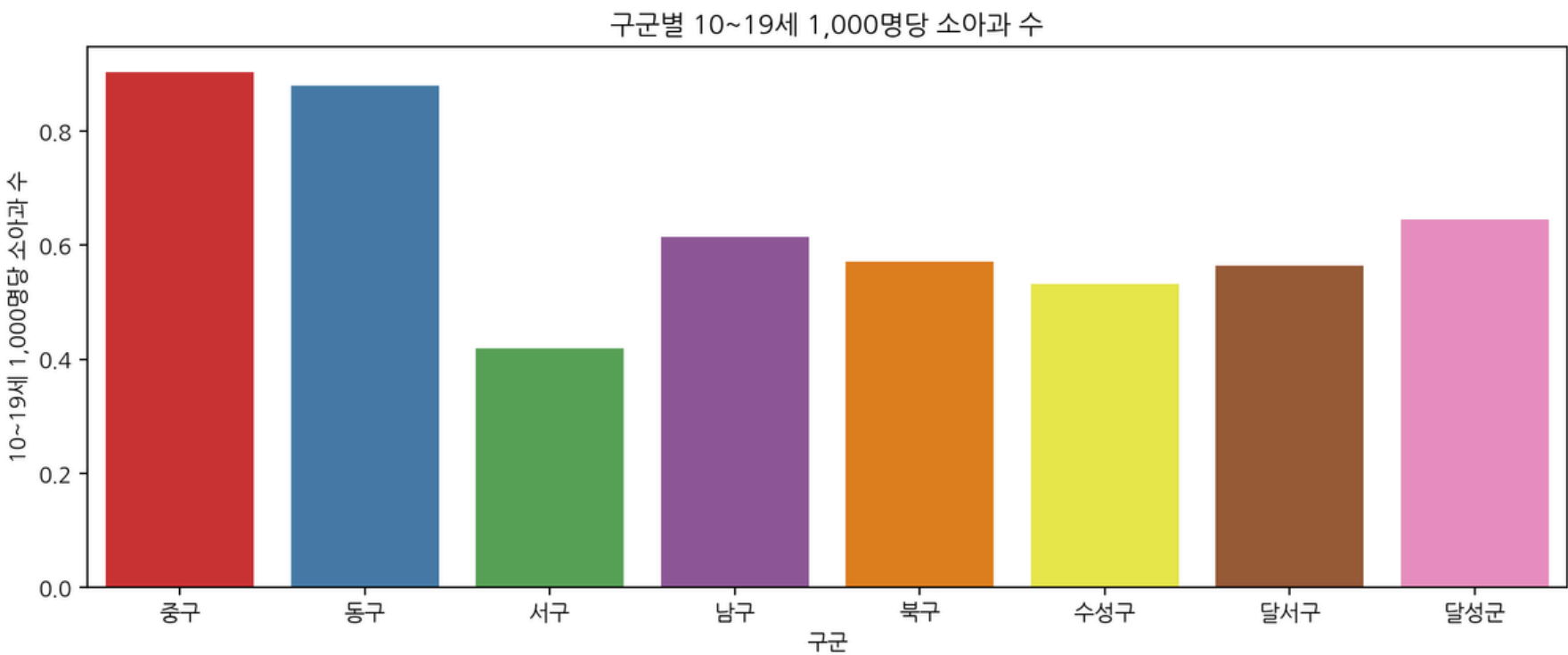
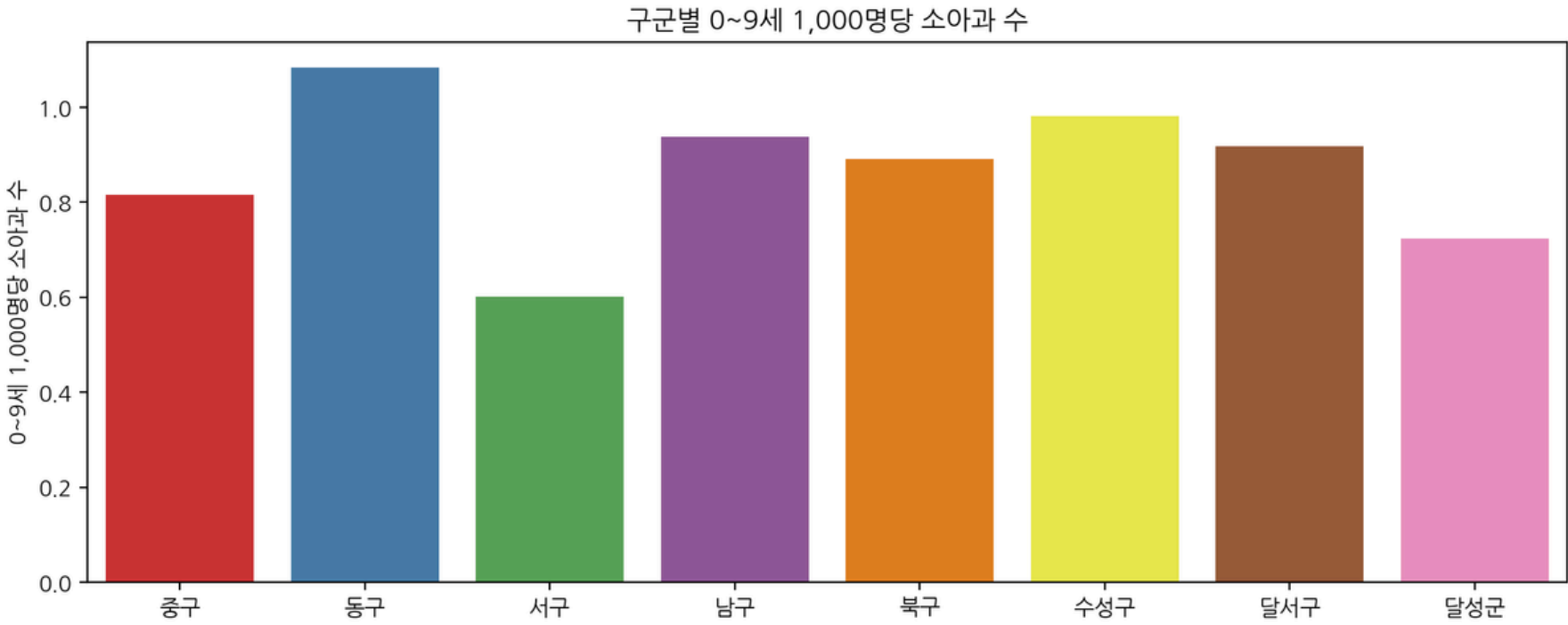


```
1 # 연령대별 100명당 산부인과 수 계산
2 age_groups = ['20~29세 ', '30~39세 ', '40~49세 ', '50~59세 ', '60~69세 ', '70~79세 ', '80대 이상 ']
3
4 for age_group in age_groups:
5     merged_df[f'{age_group} 1000명당 산부인과 수'] = (merged_df['산부인과 수'] / merged_df[age_group]) * 1000
6
7 # 예시: 특정 연령대에 대한 시각화
8 plt.figure(figsize=(12, 15))
9 plt.subplot(3, 1, 1)
10 plt.title('구군별 기혼 여성 20~29세 1000명당 산부인과 수')
11 sns.barplot(data=merged_df, x='구군', y='20~29세 1000명당 산부인과 수', hue='구군',palette='Set1')
12 plt.xlabel(' ')
13 plt.subplot(3, 1, 2)
14 plt.title('구군별 기혼 여성 30~39세 1000명당 산부인과 수')
15 sns.barplot(data=merged_df, x='구군', y='30~39세 1000명당 산부인과 수', hue='구군',palette='Set1')
16 plt.xlabel(' ')
17 plt.subplot(3, 1, 3)
18 plt.title('구군별 기혼 여성 40~49세 1000명당 산부인과 수')
19 sns.barplot(data=merged_df, x='구군', y='40~49세 1000명당 산부인과 수', hue='구군',palette='Set1')
20 plt.xlabel(' ')
21 plt.ylabel('100명당 산부인과 수')
22 plt.show()
23
```

- 가임기 기혼여성이 2번째로 많은 북구가 군위군을 제회한 대구시 구군중 가임기 기혼 여성이 가장 적은 중구보다 산부인과 수가 적은 모습을 보이며 전체적인 비율또한 대다수의 구군이 중구의 1/3의 모습을 보이는데 이또한 지역간 의료 불균형을 보임

대구 구군별 의료시설 분석

소아과의원

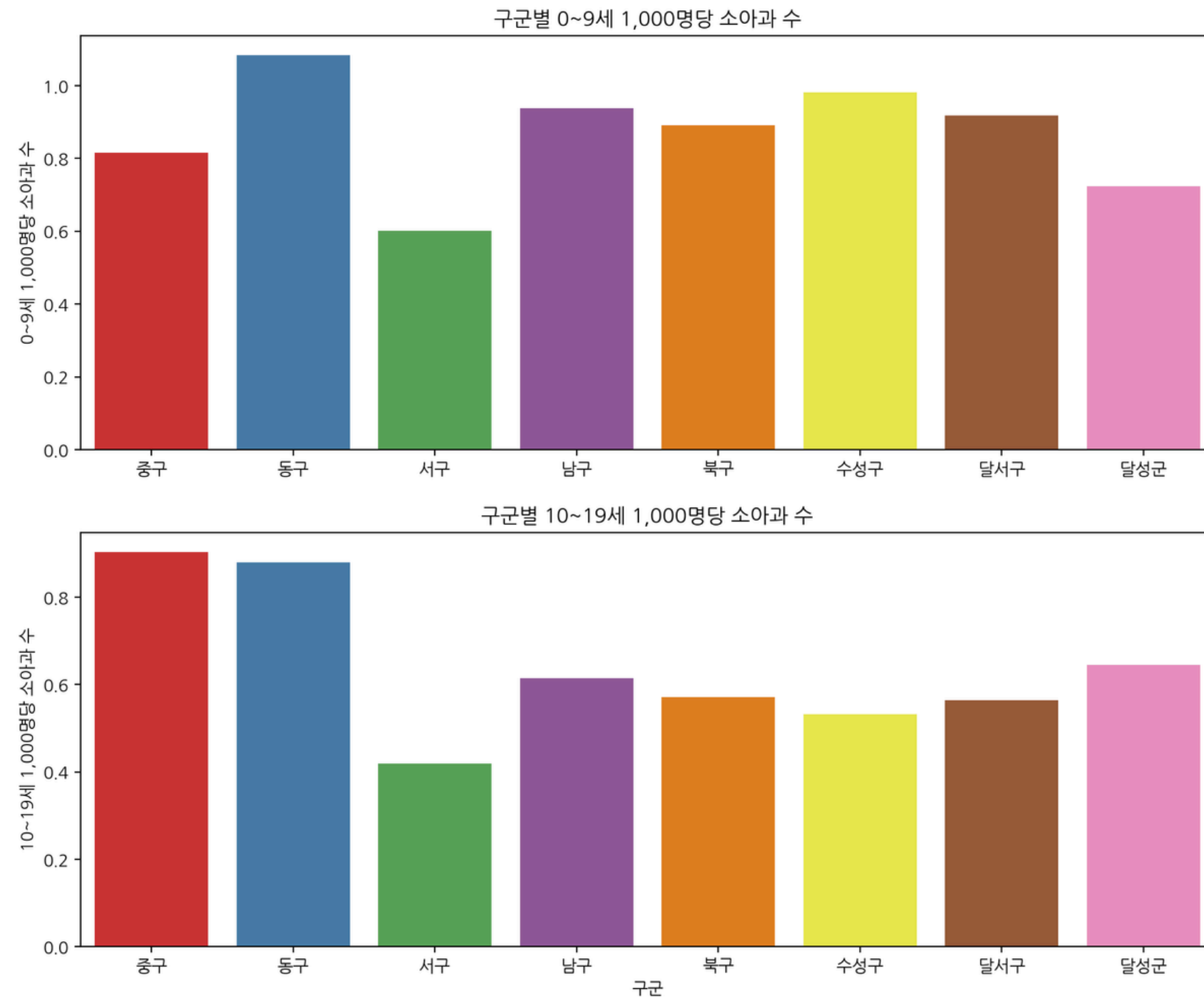


구군	0~9세	10~19세	소아과 수	병원 비율(1,000명) 10대 미만/10대
중구	7,357	6,643	6	0.82/0.9
동구	21,222	26,137	23	1.08/0.88
서구	6,647	9,530	4	0.6/0.41
남구	5,334	8,133	5	0.93/0.61
북구	24,674	38,499	22	0.89/0.57
수성구	25,486	46,915	25	0.98/0.53
달서구	30,533	49,620	28	0.91/0.56
달성군	23,496	26,331	17	0.72/0.64
군위군	492	860	0	0

- 군위군을 제외한 모든 구군의 전반적인 비율이 고르게 나타남
- 상대적으로 의료 불균형이 적음

대구 구군별 의료시설 분석

소아과의원

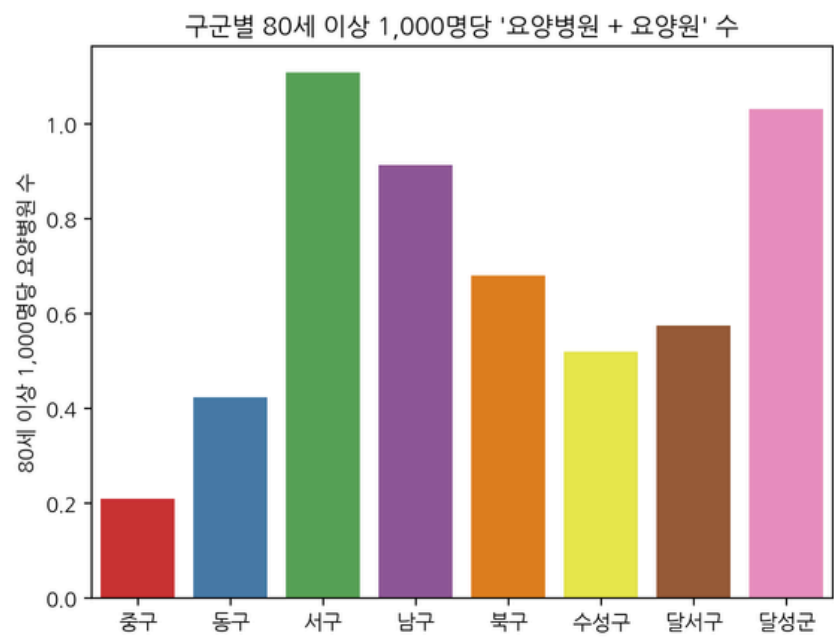
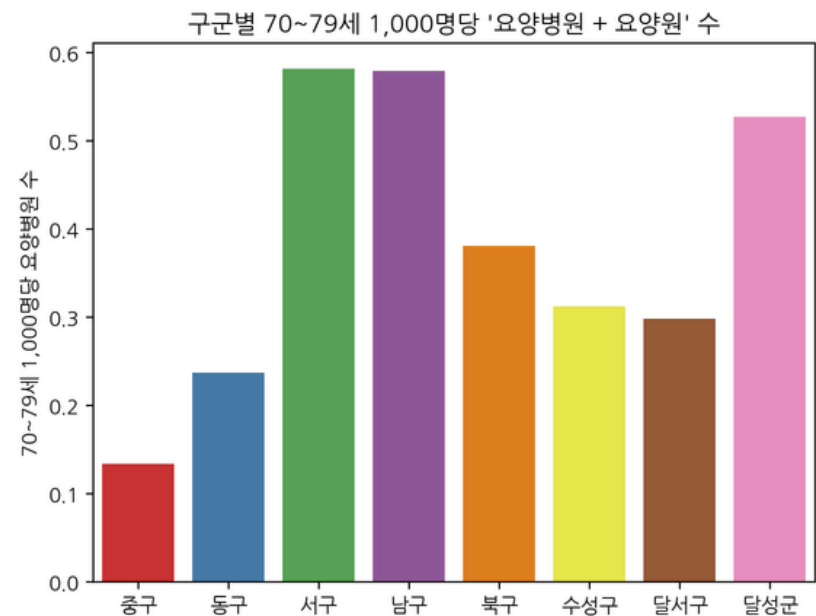
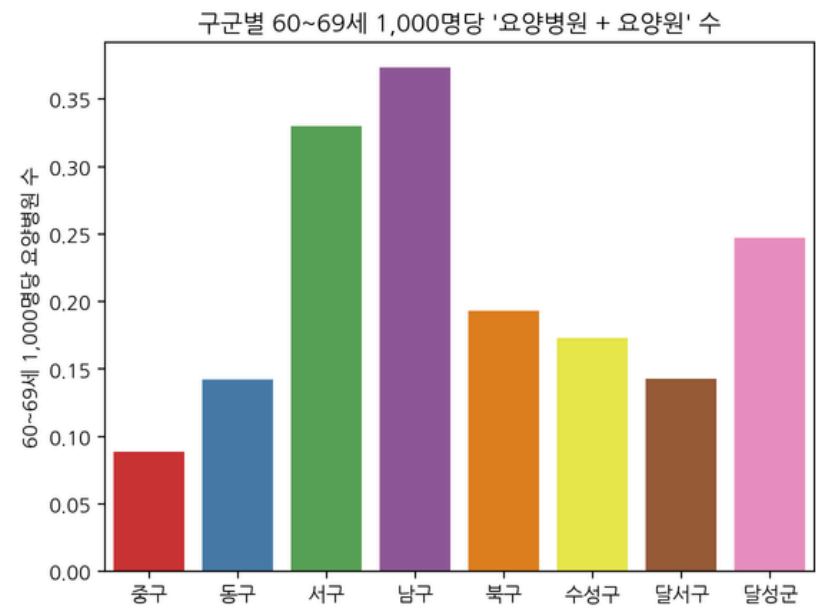


```
1 age_groups = ['0~9세 ', '10~19세 ', '20~29세 ', '30~39세 ', '40~49세 ', '50~59세 ', '60~69세 ', '70~79세 ']  
2 filtered_df = merged_df[merged_df['소아과 수'] > 0]  
3  
4 for age_group in age_groups:  
5     filtered_df[f'{age_group} 1000명당 소아과 수'] = (filtered_df['소아과 수'] / filtered_df[age_group]) * 1000  
6  
7 plt.figure(figsize=(12, 15))  
8 plt.subplot(3, 1, 1)  
9 plt.title('구군별 0~9세 1,000명당 소아과 수')  
10 sns.barplot(data=filtered_df, x='구군', y='0~9세 1000명당 소아과 수', hue = '구군', palette='Set1')  
11 plt.xlabel(' ')  
12 plt.ylabel('0~9세 1,000명당 소아과 수')  
13 plt.subplot(3, 1, 2)  
14 plt.title('구군별 10~19세 1,000명당 소아과 수')  
15 plt.ylabel('10~19세 1,000명당 소아과 수')  
16 sns.barplot(data=filtered_df, x='구군', y='10~19세 1000명당 소아과 수', hue = '구군', palette='Set1')  
17 plt.show()
```

- 군위군을 제외한 모든 구군의 전반적인 비율이 고르게 나타남
- 상대적으로 의료 불균형이 적음

대구 구군별 의료시설 분석

요양병원 + 요양원

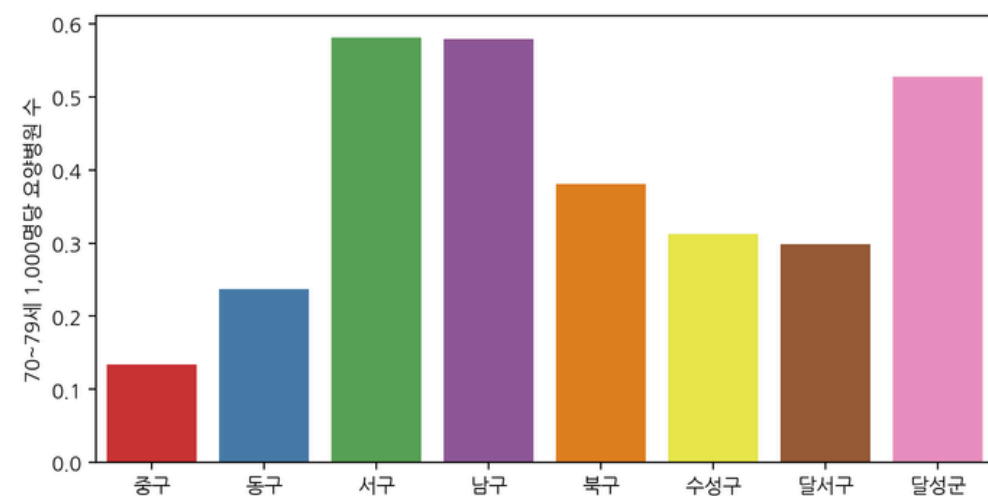
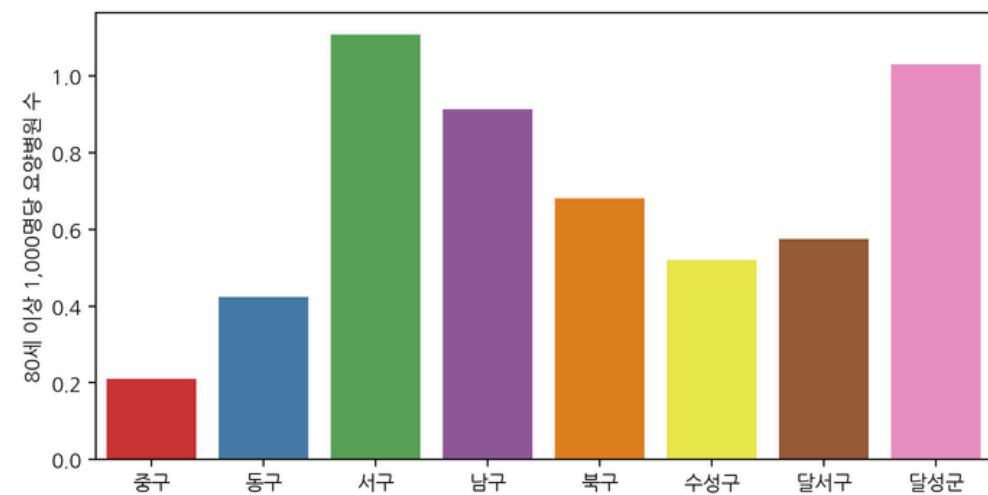
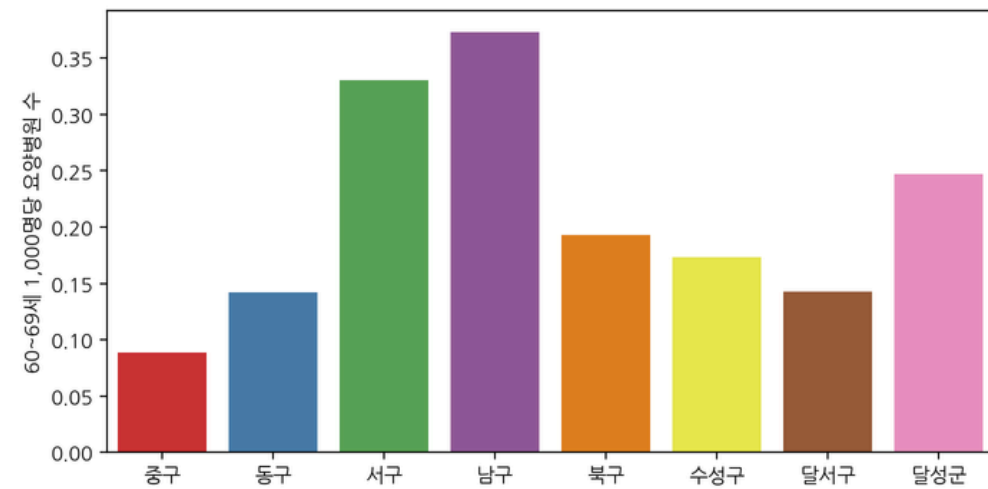


구군	60~69세	70~79세	80세 이상	요양병원 + 요양원 수	병원 비율(1,000명) 60대/70대/80대 이상
중구	11,270	7,452	4,770	1	0.09/0.13/0.2
동구	56,276	33,728	18,895	8	0.14/0.24/0.42
서구	33,316	18,916	9,925	11	0.33/0.58/1.1
남구	24,108	15,544	9,850	9	0.37/0.58/0.91
북구	62,144	31,510	17,641	12	0.19/0.38/0.68
수성구	57,777	31,972	19,256	10	0.17/0.31/0.51
달서구	84,025	40,231	20,872	12	0.14/0.3/0.57
달성군	36,419	17,072	8,730	9	0.25/0.53/1.03
군위군	22,649	4,096	3,328	0	0

- 다른 병원들과 다르게 중구의 요양병원 비율이 상대적으로 적음
- 요양병원 비율또한 최대 비율과 최소 비율이 5배 이상 차이나며 불균형이 나타남

대구 구군별 의료시설 분석

요양병원 + 요양원



```
1 age_groups = ['0~9세 ', '10~19세 ', '20~29세 ', '30~39세 ', '40~49세 ', '50~59세 ', '60~69세 ', '70~79세 ', '80세 이상 ']  
2  
3 for age_group in age_groups:  
4     merged_df[f'{age_group} 1000명당 요양병원 수'] = (merged_df['요양병원 수'] / merged_df[age_group]) * 1000  
5 filtered_df = merged_df[merged_df['요양병원 수'] > 0]  
6  
  
1 plt.title('구군별 60~69세 1,000명당 #요양병원 + 요양원# 수')  
2 #plt.figure(figsize=(8, 4))  
3 sns.barplot(data=filtered_df, x='구군', y='60~69세 1000명당 요양병원 수', hue='구군', palette='Set1')  
4 plt.ylabel('60~69세 1,000명당 요양병원 수')  
5 plt.xlabel('')
```

- 다른 병원들과 다르게 중구의 요양병원 비율이 상대적으로 적음
- 요양병원 비율또한 최대 비율과 최소 비율이 5배 이상 차이나며 불균형이 나타남

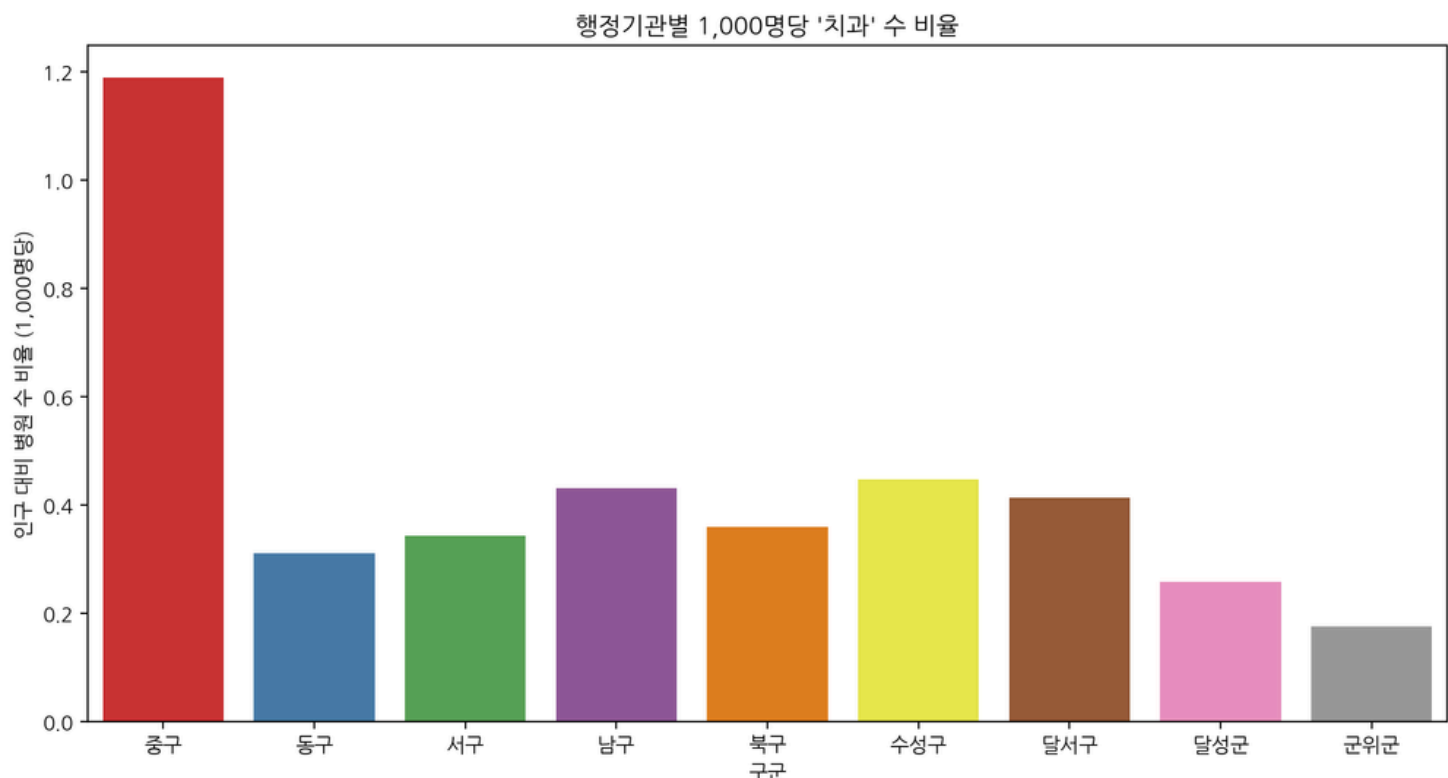
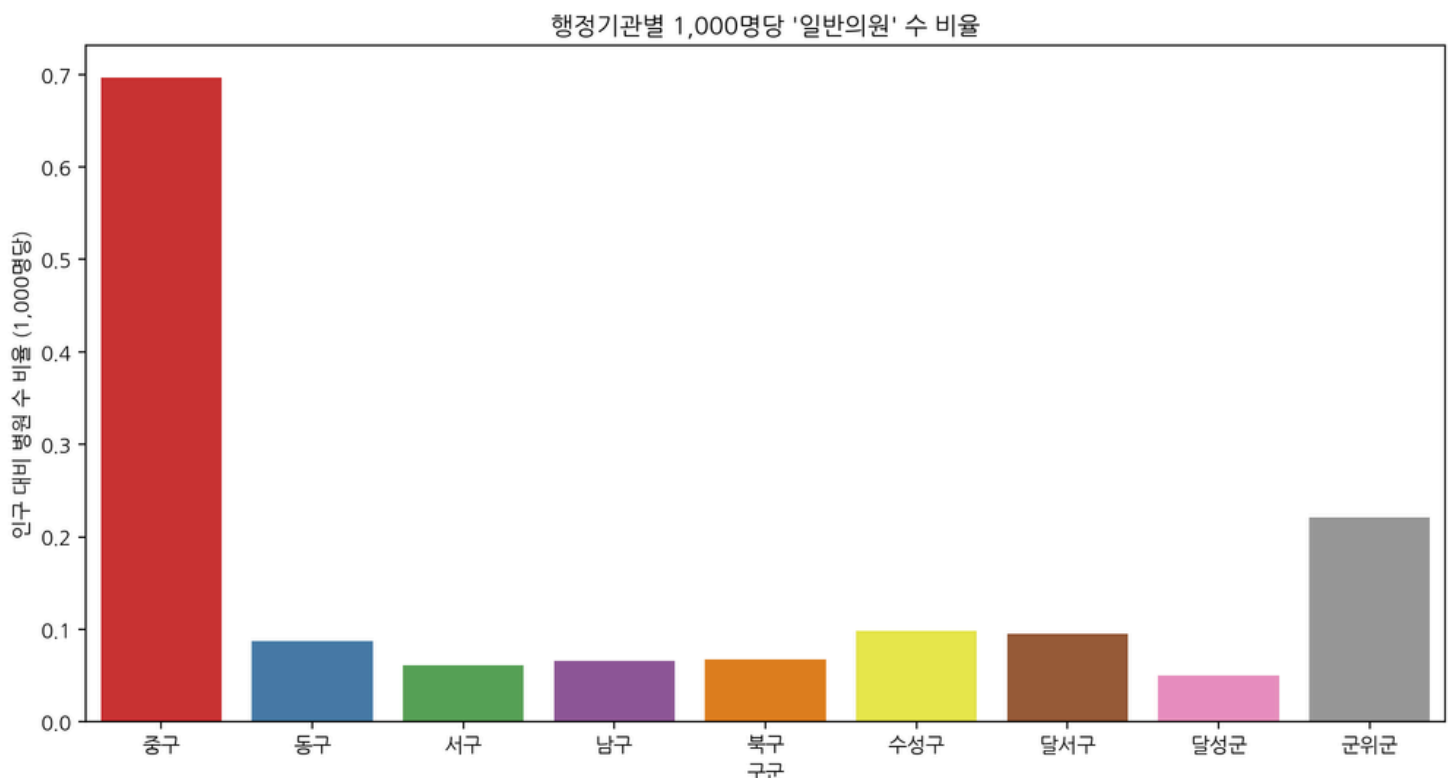
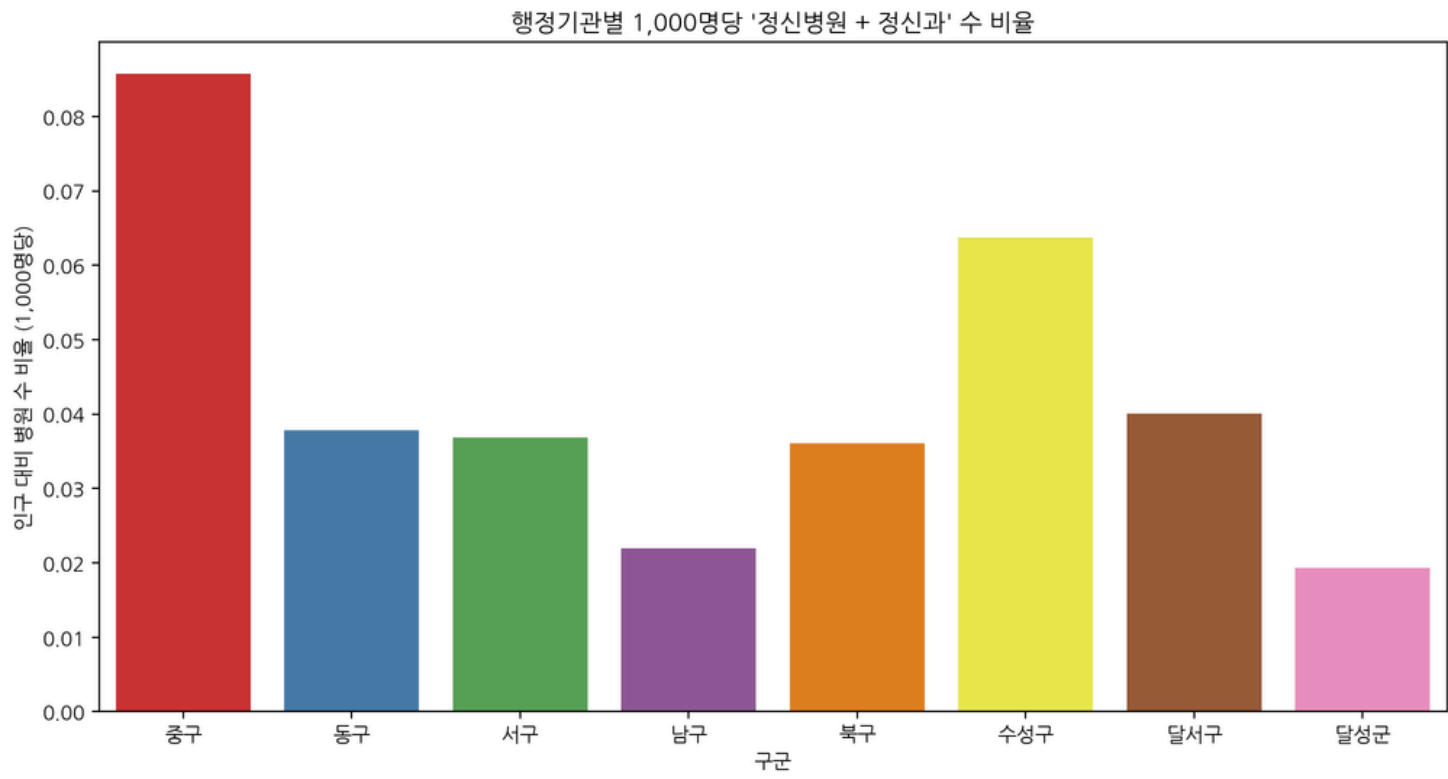
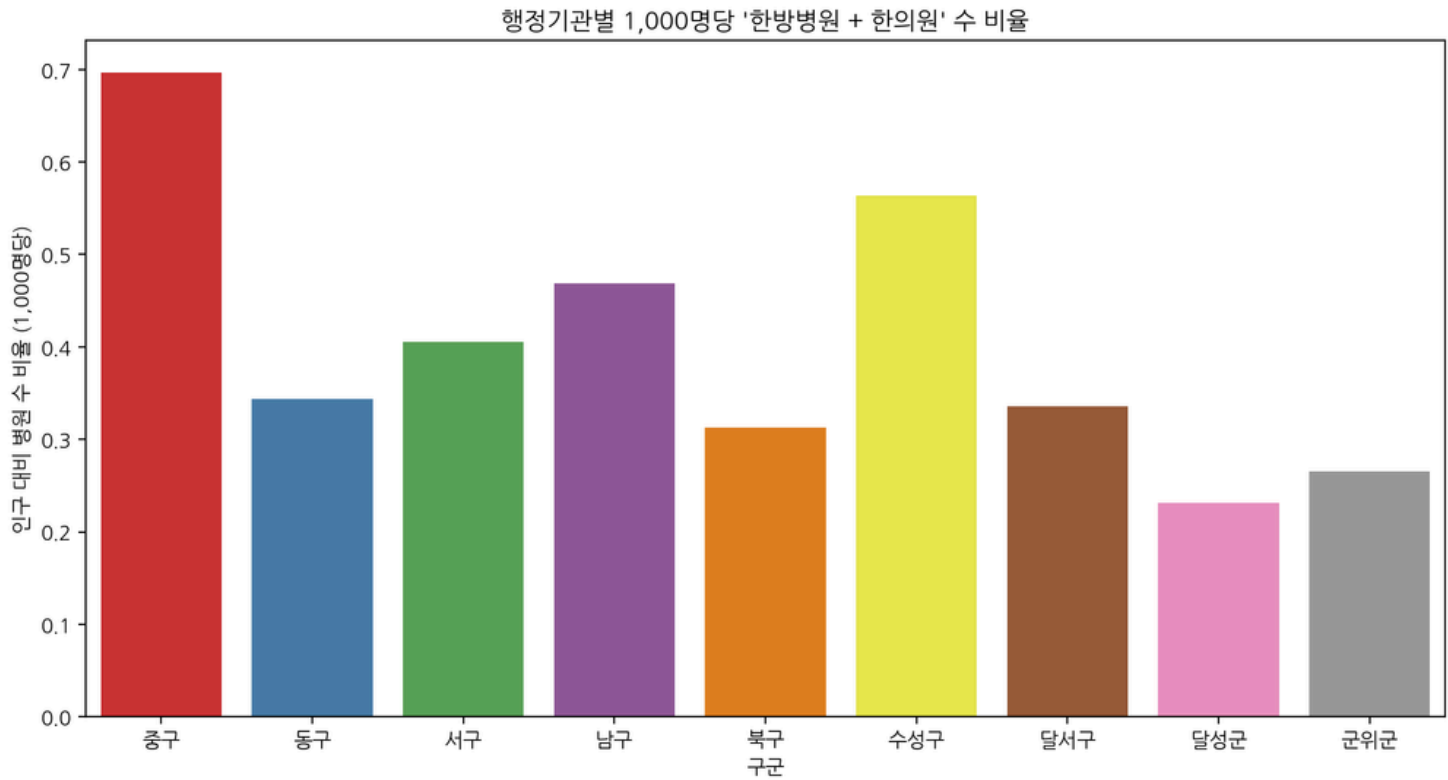
대구 구군별 의료시설 분석

분석한 과들을 알아보자

한방병원 + 한의원, 정신병원 + 정신과,
일반의원, 치과, 내과, 외과, 이비인후과,
안과, 정형외과, 마취통증의학과, 피부과,
재활의학과, 가정의학과, 신경외과,
영상의학과, 비뇨기과, 연합의원

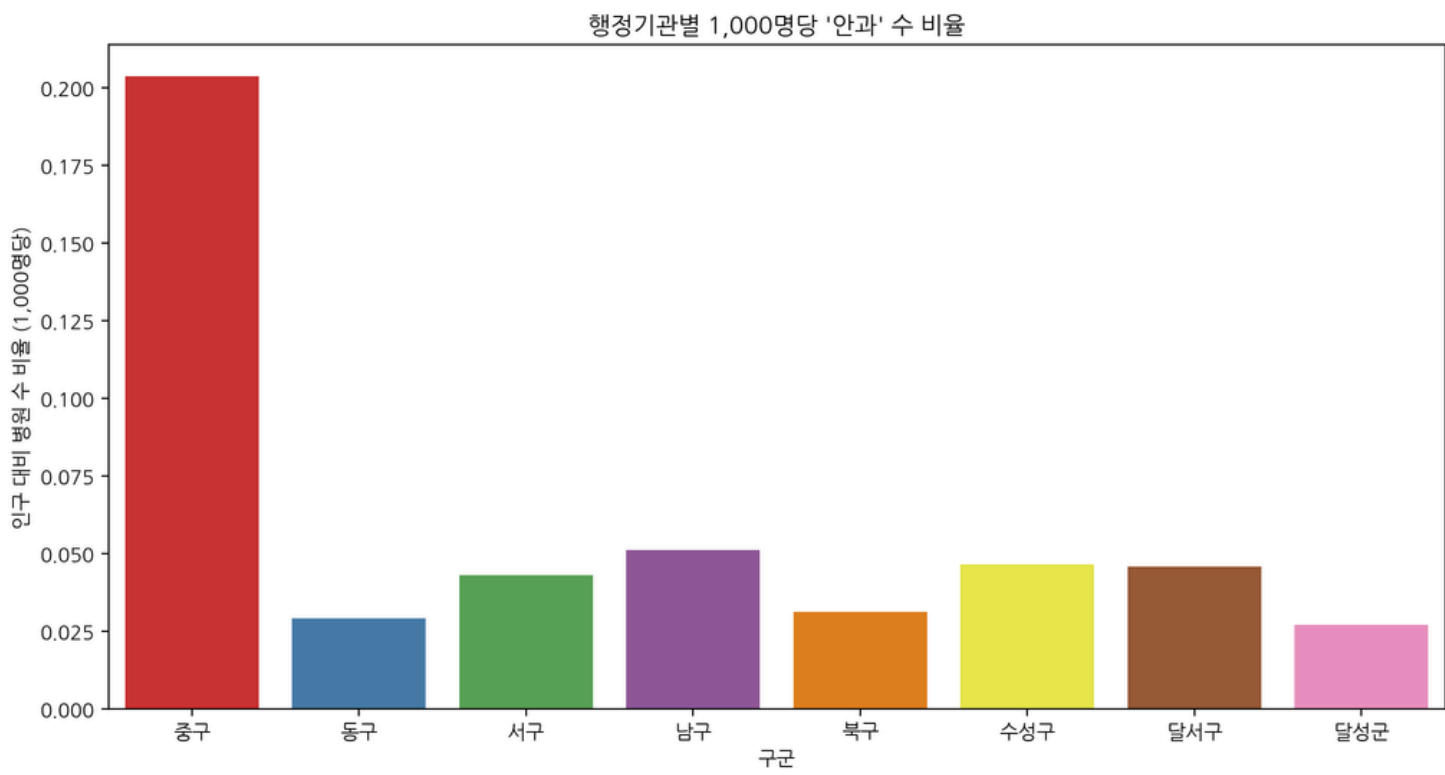
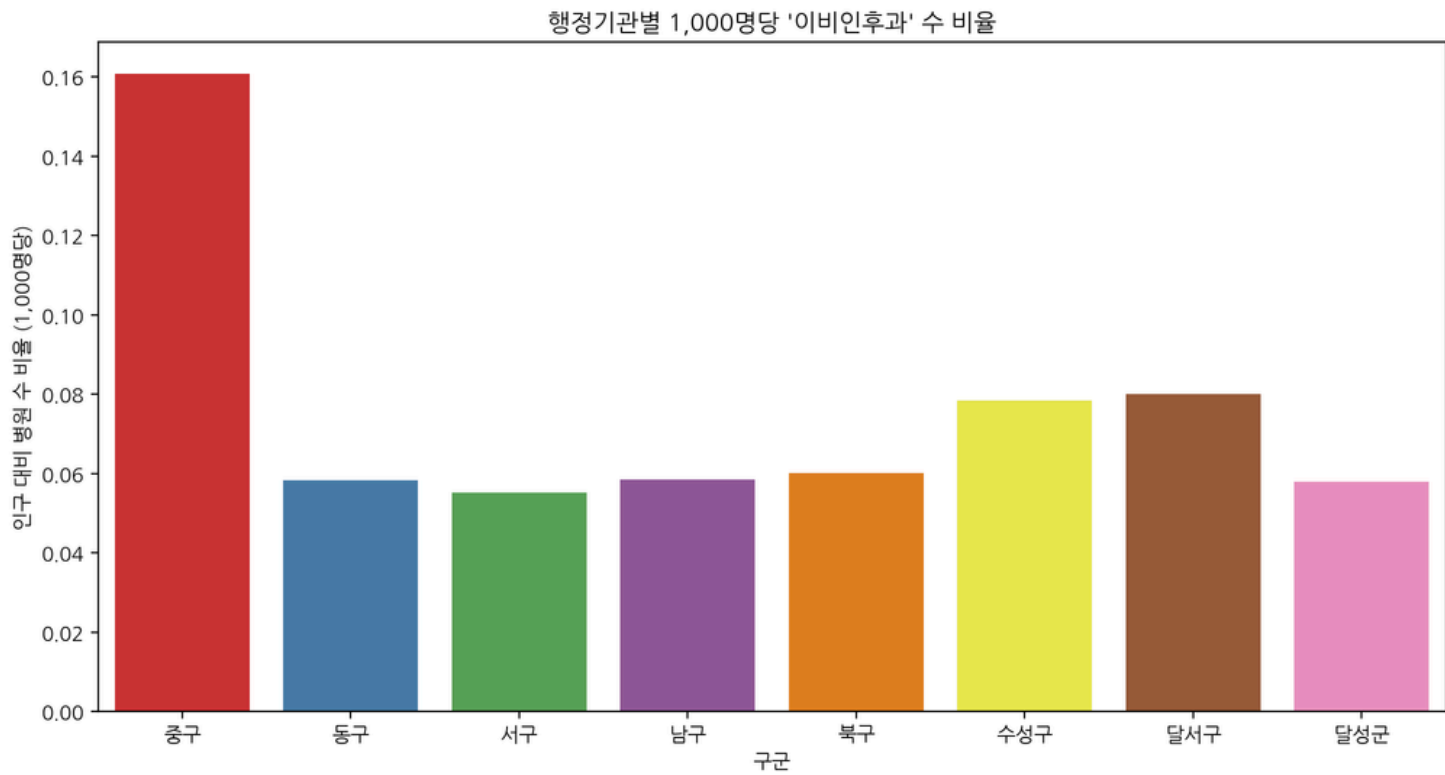
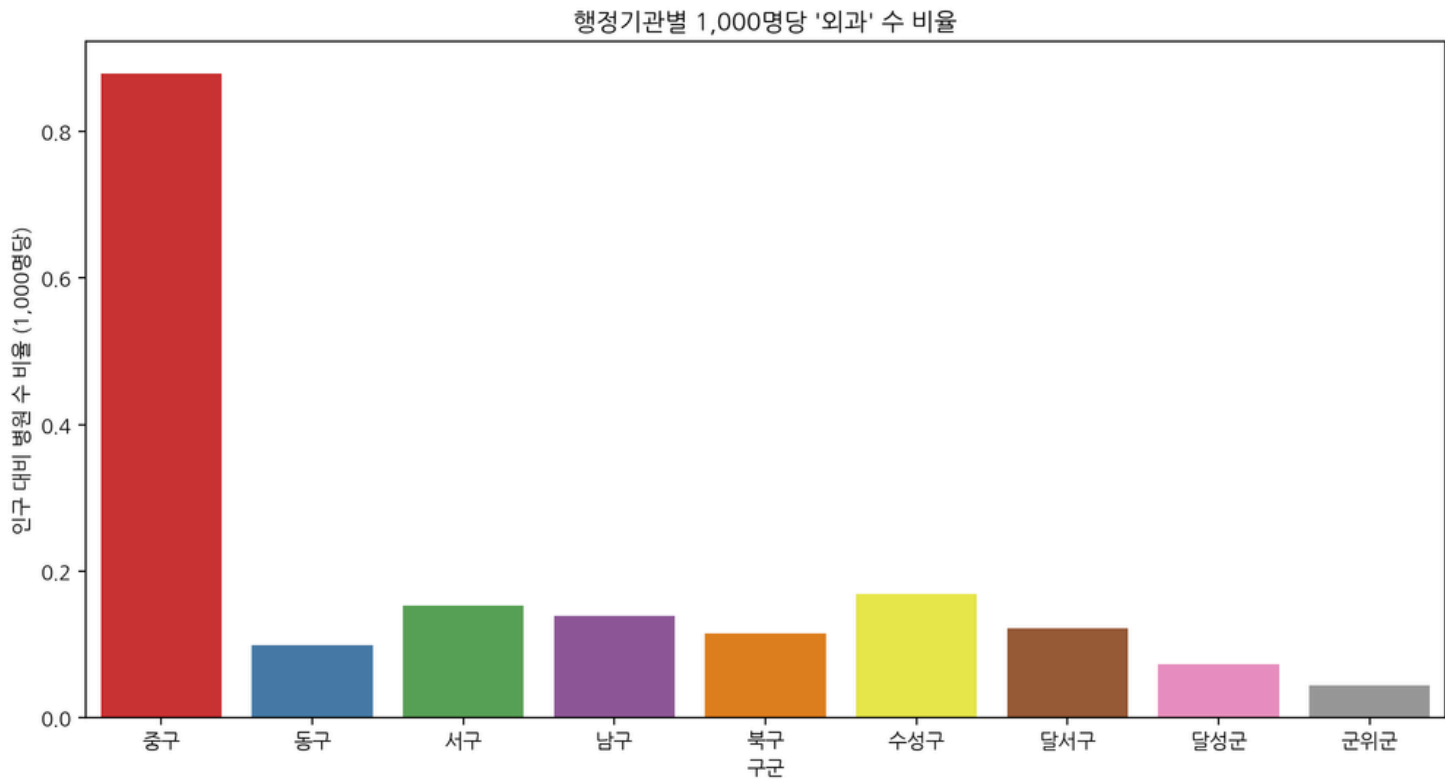
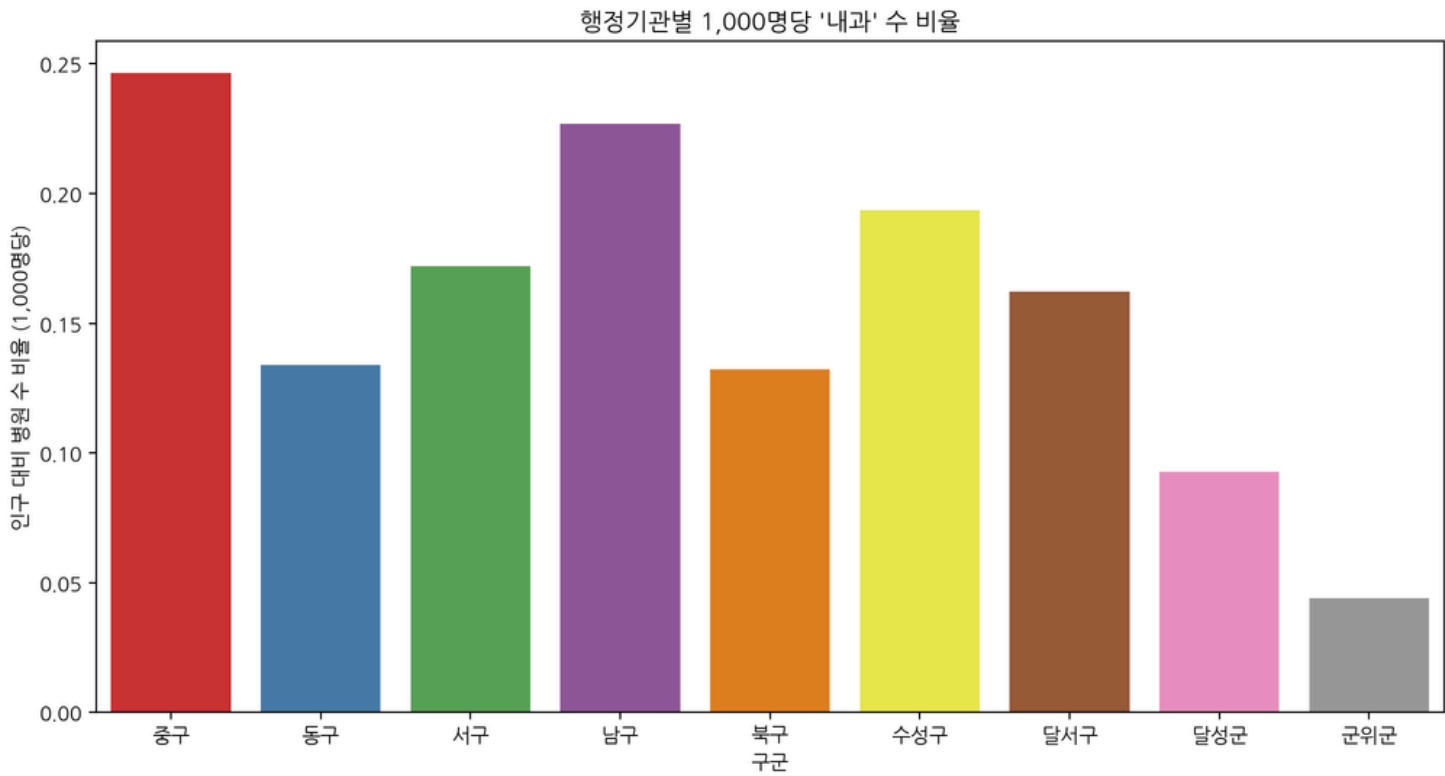
대구 구군별 의료시설 분석

한방병원 + 한의원, 정신병원 + 정신과, 일반의원, 치과



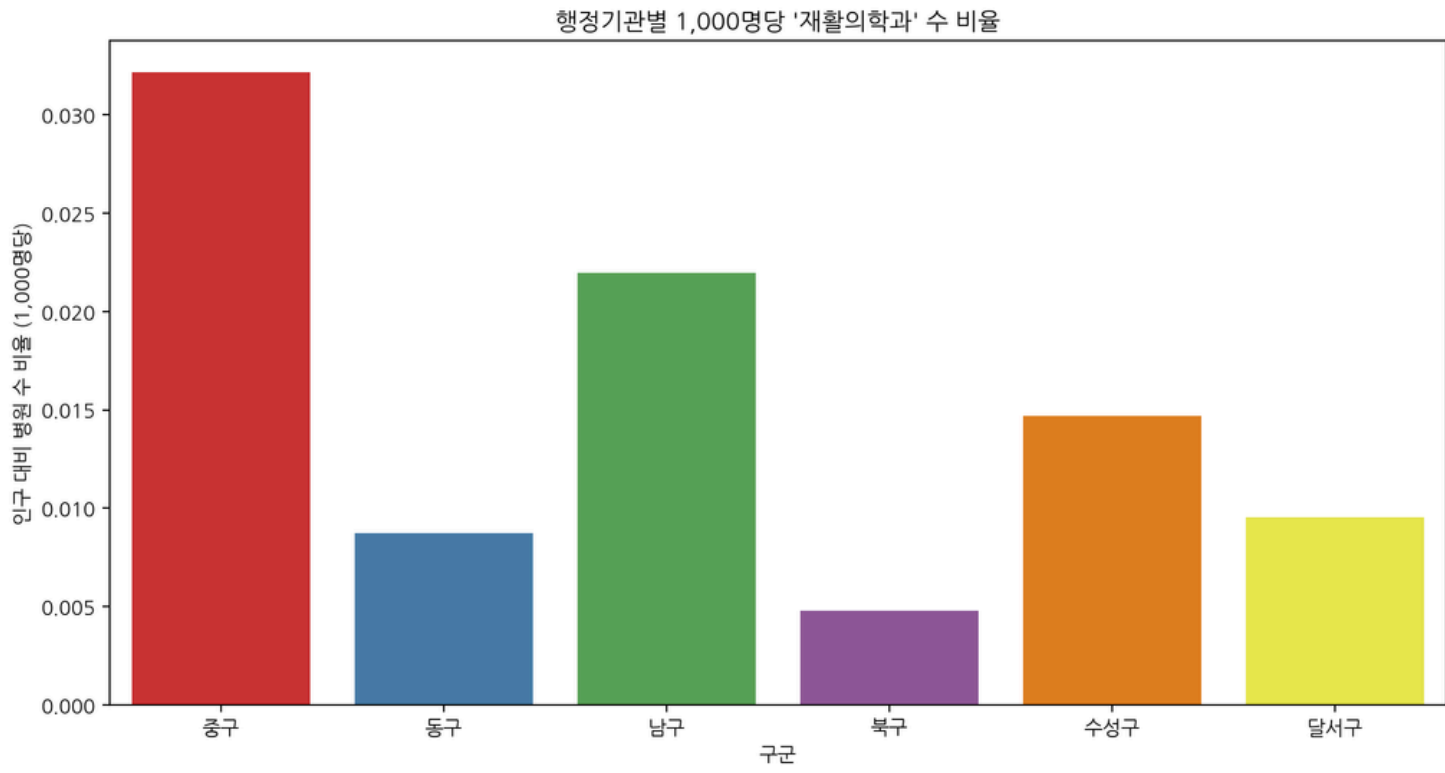
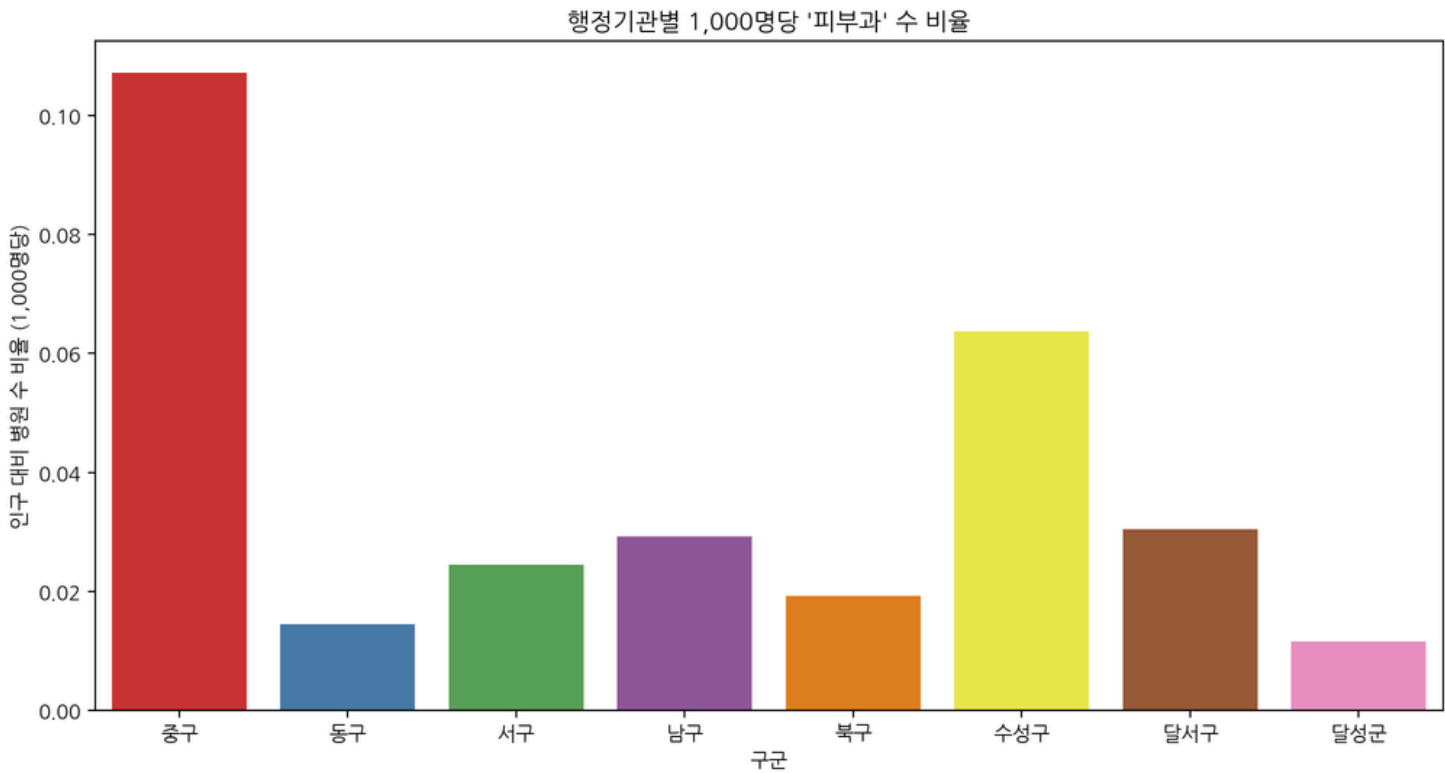
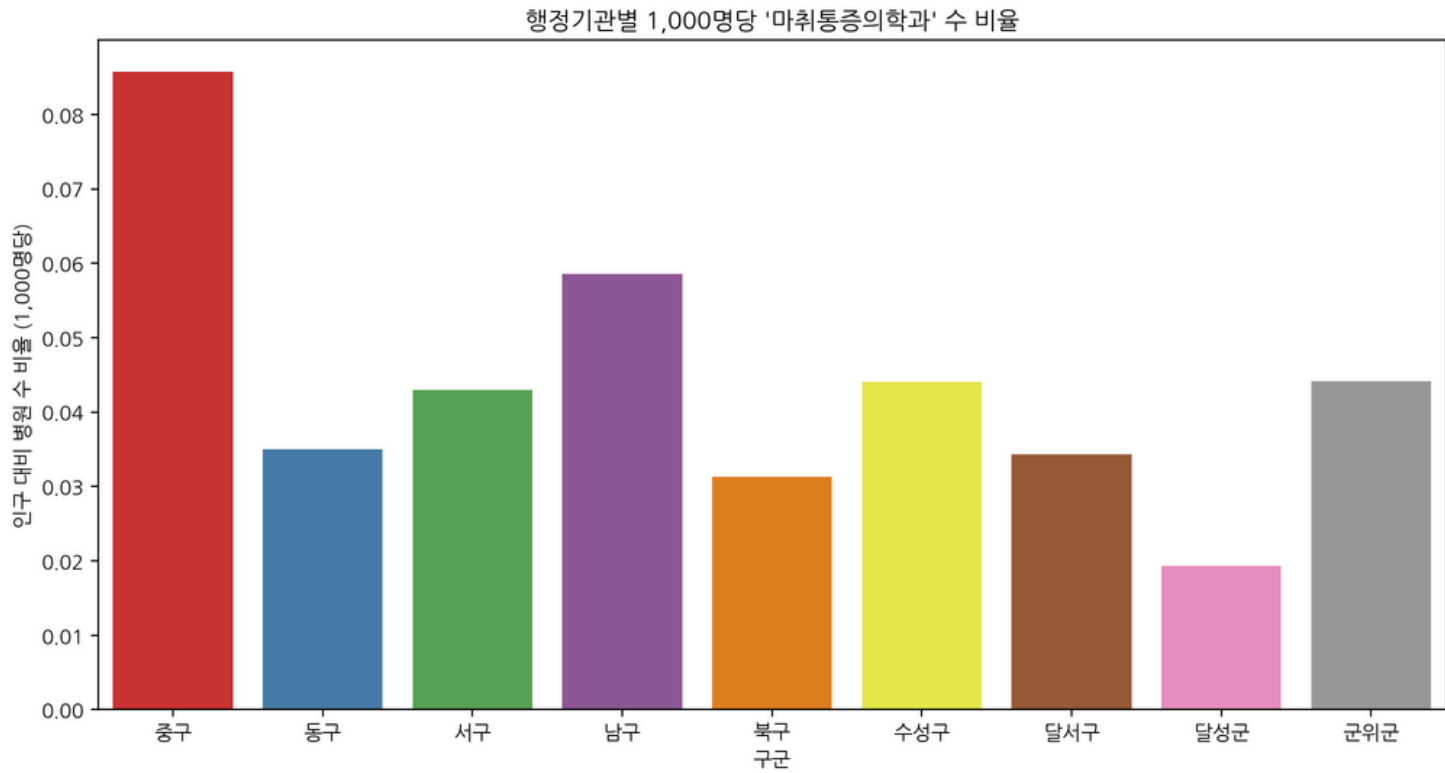
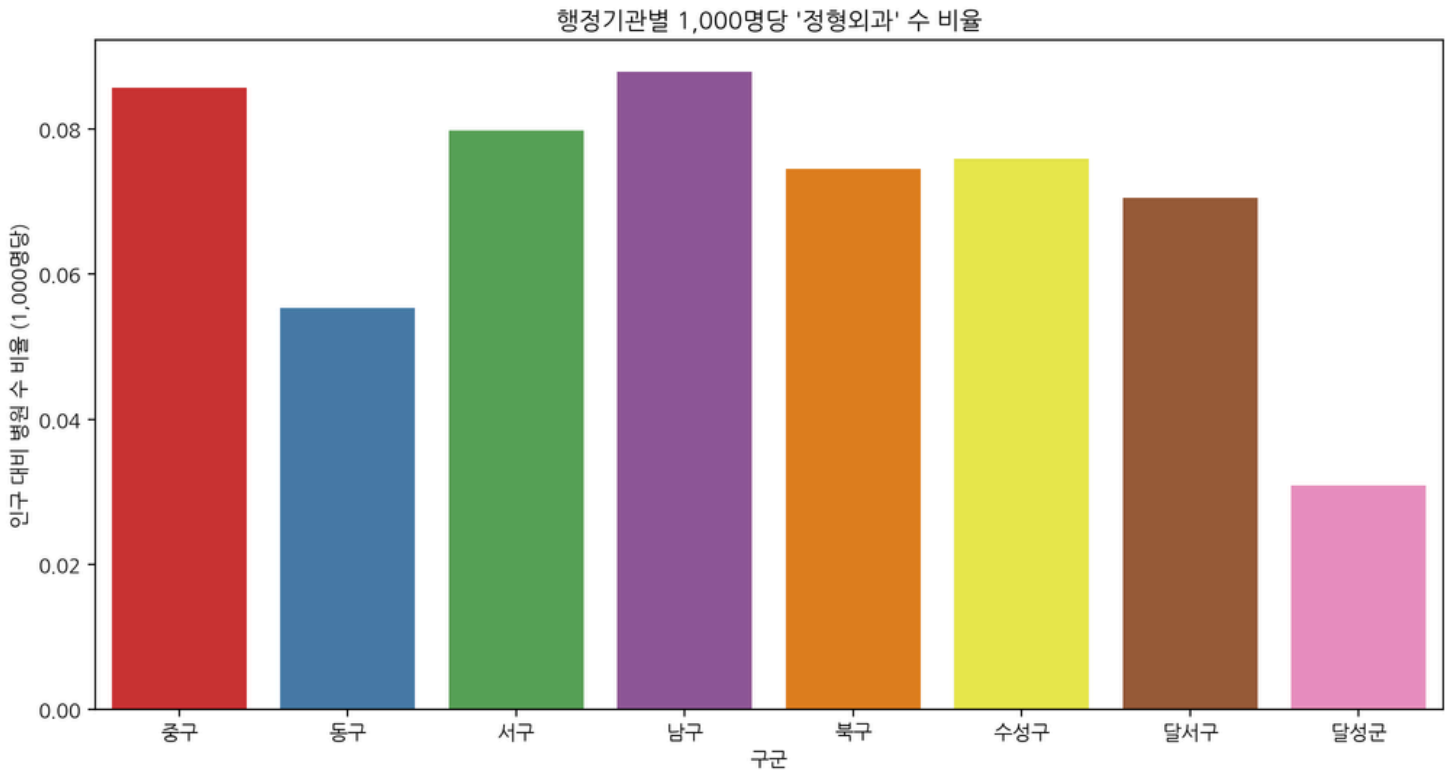
대구 구군별 의료시설 분석

내과, 외과, 이비인후과, 안과



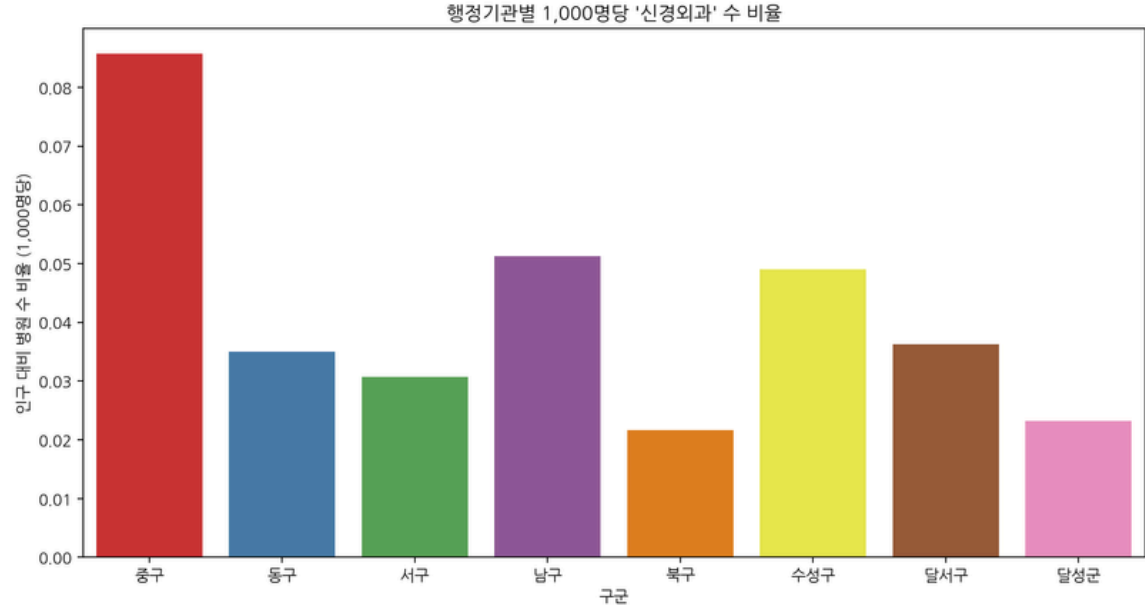
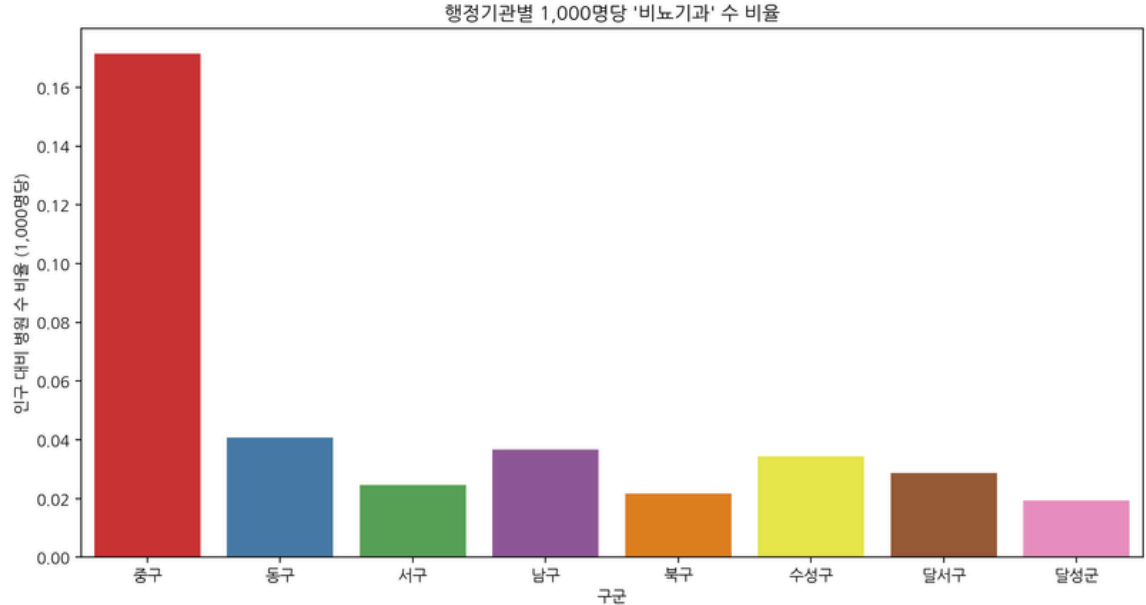
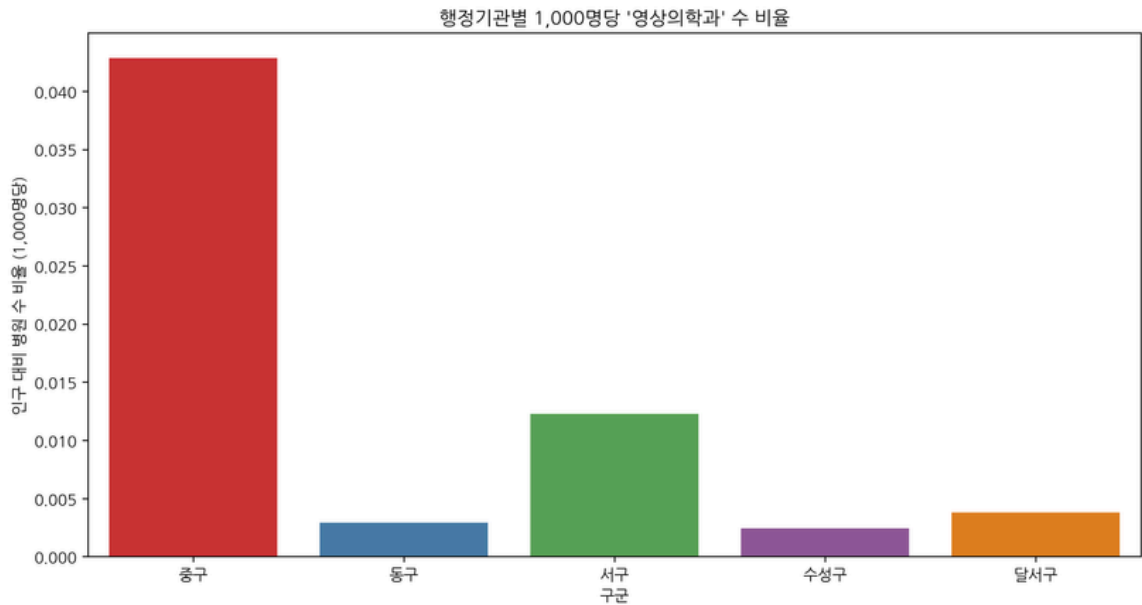
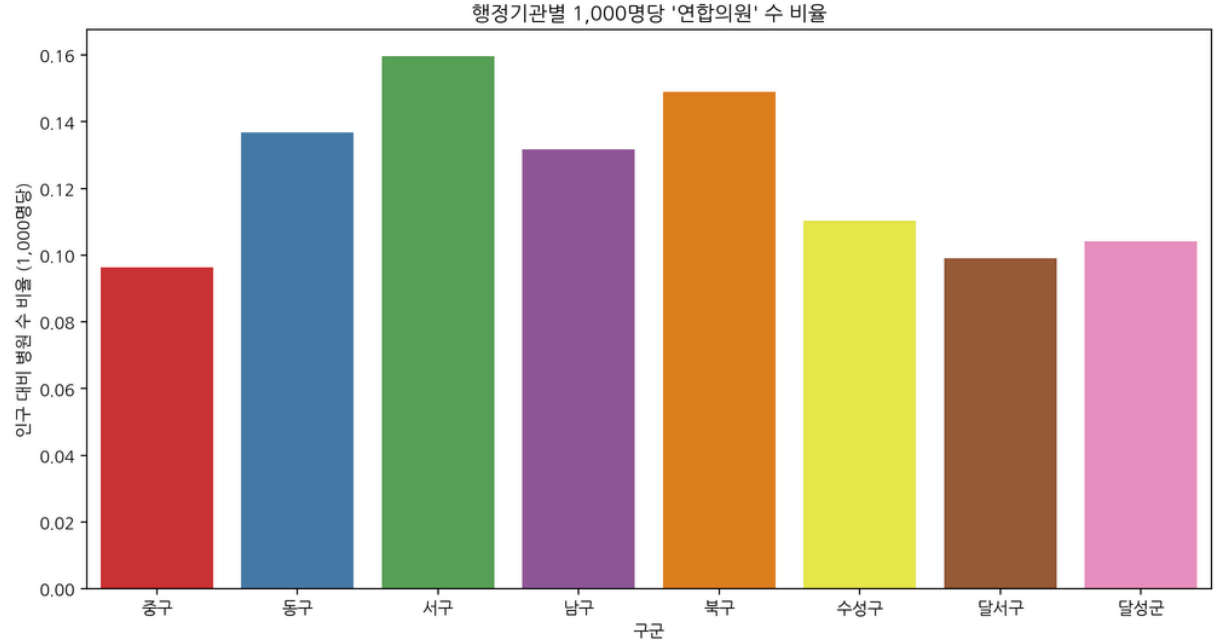
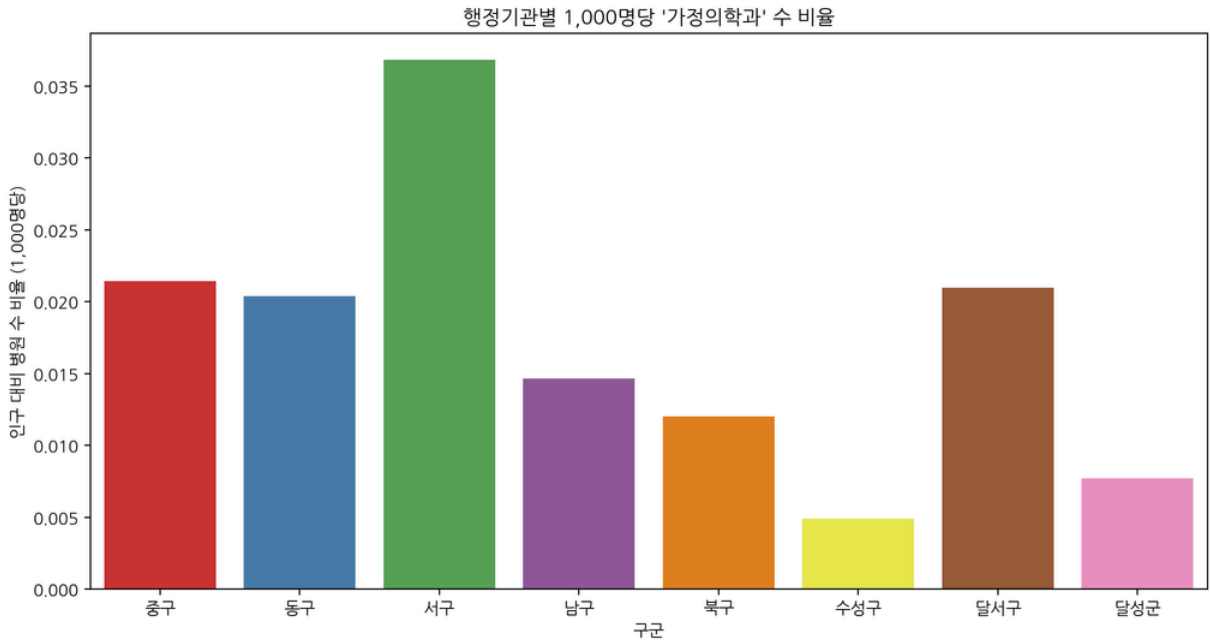
대구 구군별 의료시설 분석

정형외과, 마취통증의학과, 피부과, 재활의학과



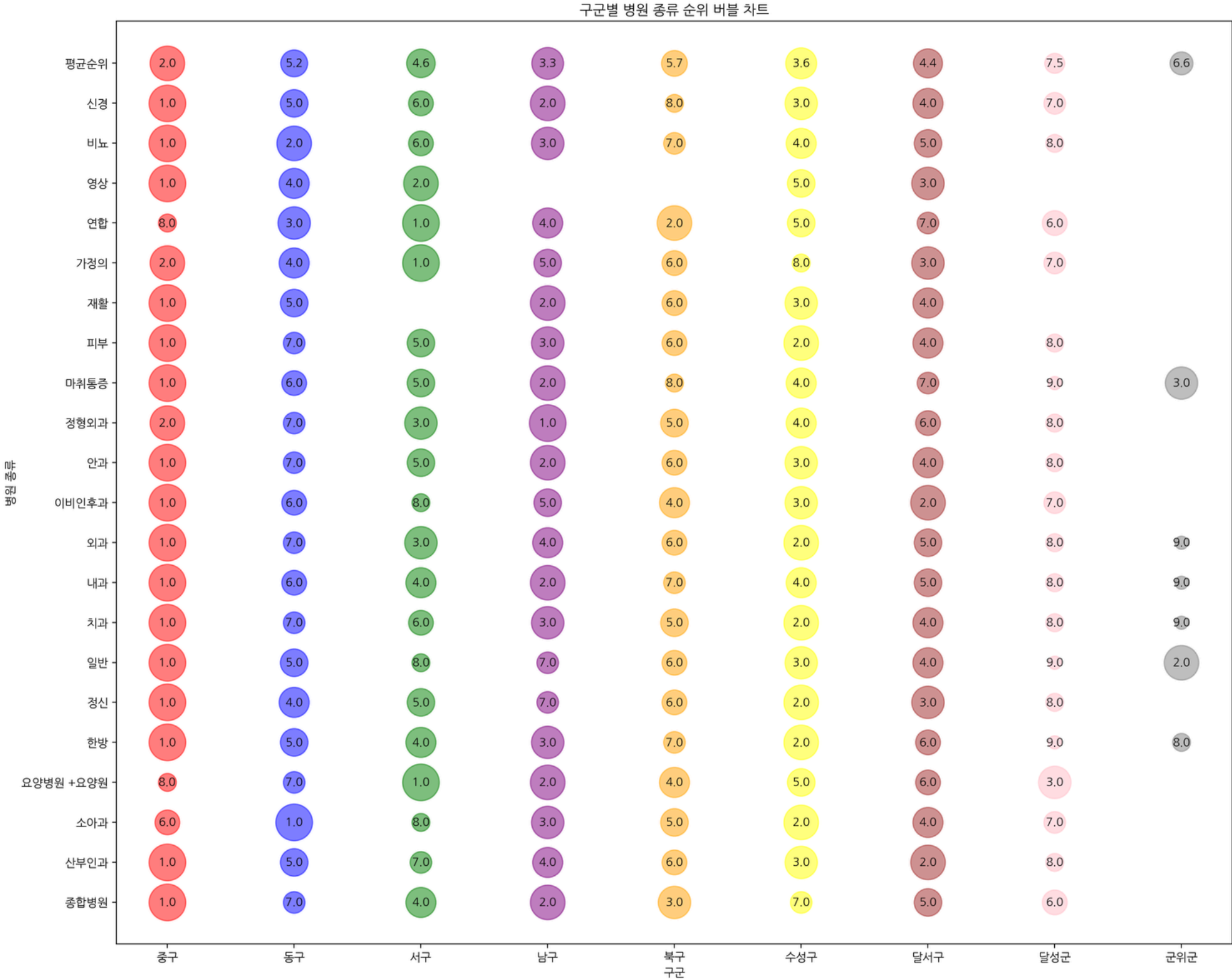
대구 구군별 의료시설 분석

가정의학과, 연합의원, 영상의학과, 비뇨기과, 신경외과



대구 구군별 의료시설 분석

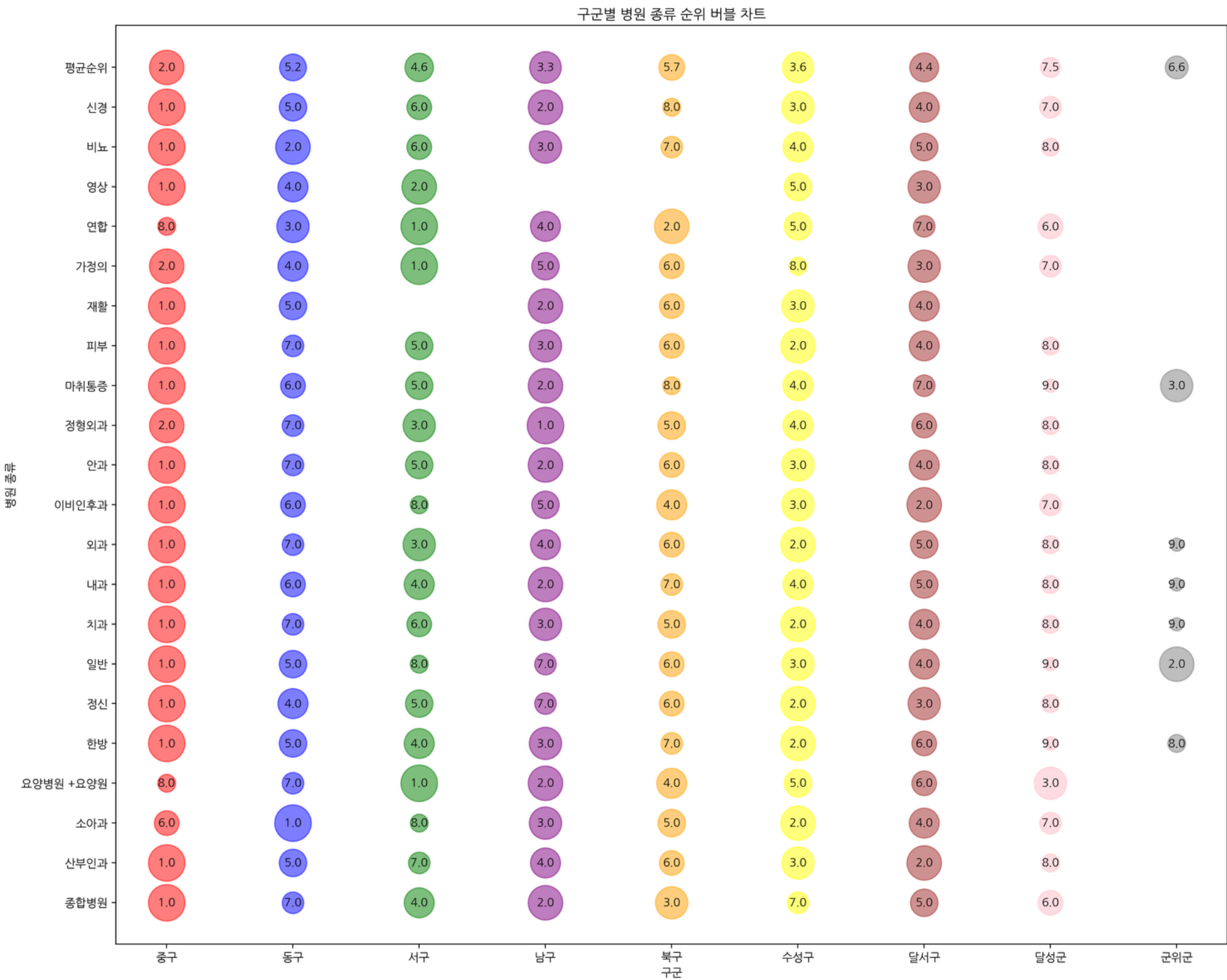
병원 종류별 순위 차트



구군	총인구수	평균순위(없는과 제외)	없는 병원 종류
중구	93,338	2	0
동구	343,392	5.2	0
서구	162,903	4.6	1개 / 재활의학과
남구	136,582	3.3	1개 / 영상의학과
북구	416,109	5.7	1개 / 영상의학과
수성구	408,215	3.6	5.0
달서구	524,448	4.4	8.0
달성군	259,024	7.5	2개 재활의학과 영상의학과
군위군	22,649	6.6	15개 종합병원, 산부인과, 소아과, 요양병원, 정신과,이비인후 과, 안과, 정형외과, 피부과, 재활의학과, 가정의학과, 연 합의원, 영상의학과, 비뇨기 과, 신경외과

대구 구군별 의료시설 분석

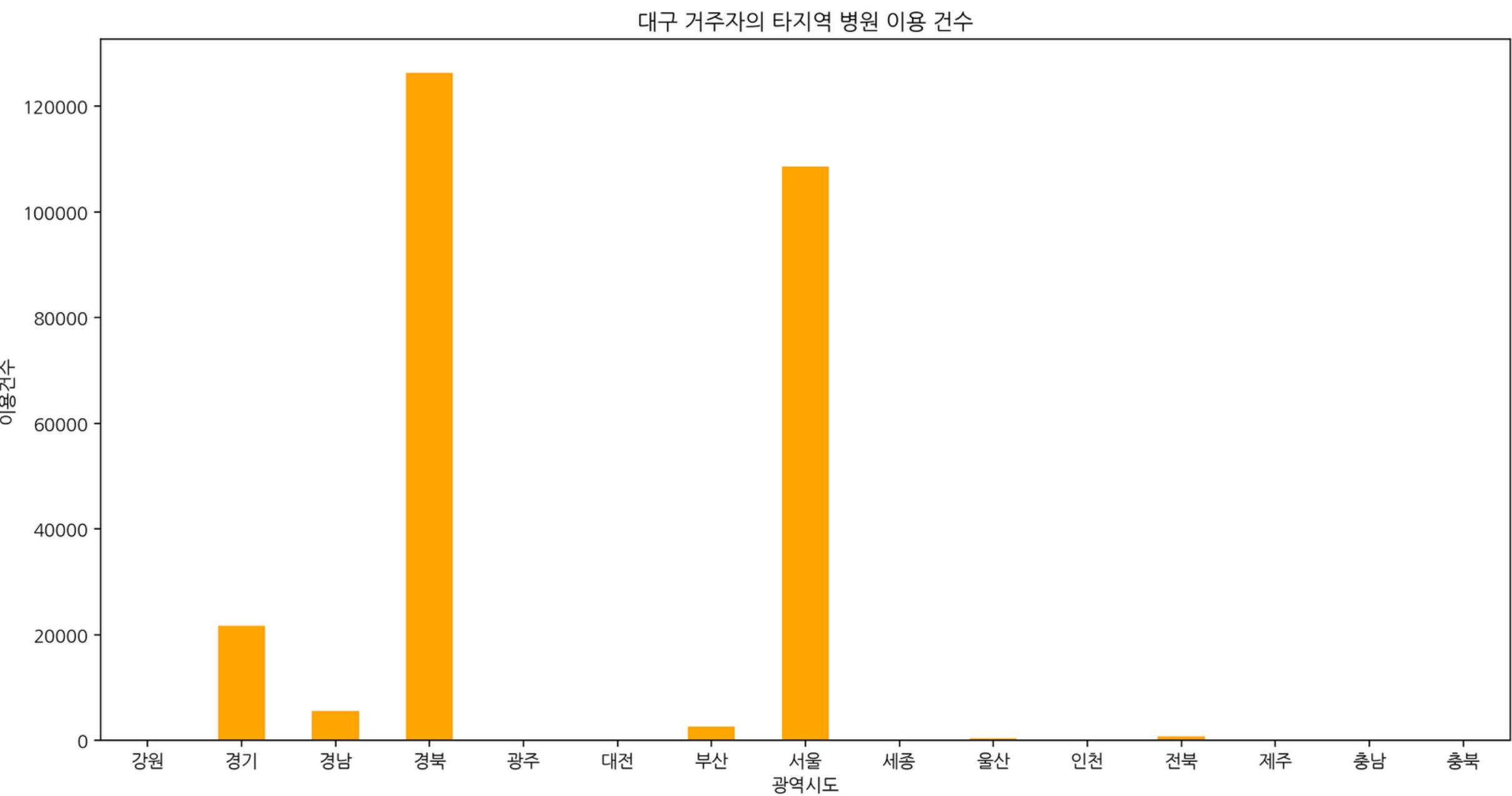
병원 종류별 순위 차트



```
1 columns_to_plot = ['병원종류', '중구', '동구', '서구', '남구', '북구', '수성구', '달서구', '달성군', '군위군']
2 df_plot = rank[columns_to_plot].set_index('병원종류')
3
4 # NaN 값을 0으로 대체
5 df_plot = df_plot.fillna(0)
6
7 # 데이터 타입을 float으로 변환
8 df_plot = df_plot.astype(float)
9
10 # 구군별 색상 정의
11 colors = {
12     '중구': 'red',
13     '동구': 'blue',
14     '서구': 'green',
15     '남구': 'purple',
16     '북구': 'orange',
17     '수성구': 'yellow',
18     '달서구': 'brown',
19     '달성군': 'pink',
20     '군위군': 'gray'
21 }
22
23 # 최대 값 구하기
24 max_value = max(df_plot.values.flatten())
25
26 # 버블 차트 그리기
27 fig, ax = plt.subplots(figsize=(15, 12))
28
29 for i, hospital in enumerate(df_plot.index):
30     for j, district in enumerate(df_plot.columns):
31         value = df_plot.loc[hospital, district]
32         if value > 0: # 0보다 큰 값만 표시
33             # 최대 크기에서의 비율에 따라 크기를 조정 (값이 작을수록 커짐)
34             size = (max_value - value + 1.2) * 109 # 1을 더해 0 이하가 되지 않도록
35             ax.scatter(j, i, s=size, alpha=0.5, color=colors[district])
36             ax.text(j, i, f'{value:.1f}', ha='center', va='center', color='black')
37
38 ax.set_xticks(range(len(df_plot.columns)))
39 ax.set_xticklabels(df_plot.columns)
40 ax.set_yticks(range(len(df_plot.index)))
41 ax.set_yticklabels(df_plot.index)
42
43 ax.set_title('구군별 병원 종류 순위 버블 차트')
44 ax.set_xlabel('구군')
45 ax.set_ylabel('병원 종류')
46
47 plt.tight_layout()
48 plt.show()
```

대구와 다른 지역간 분석

대구 시민들의 타지역 병원 이용 건수

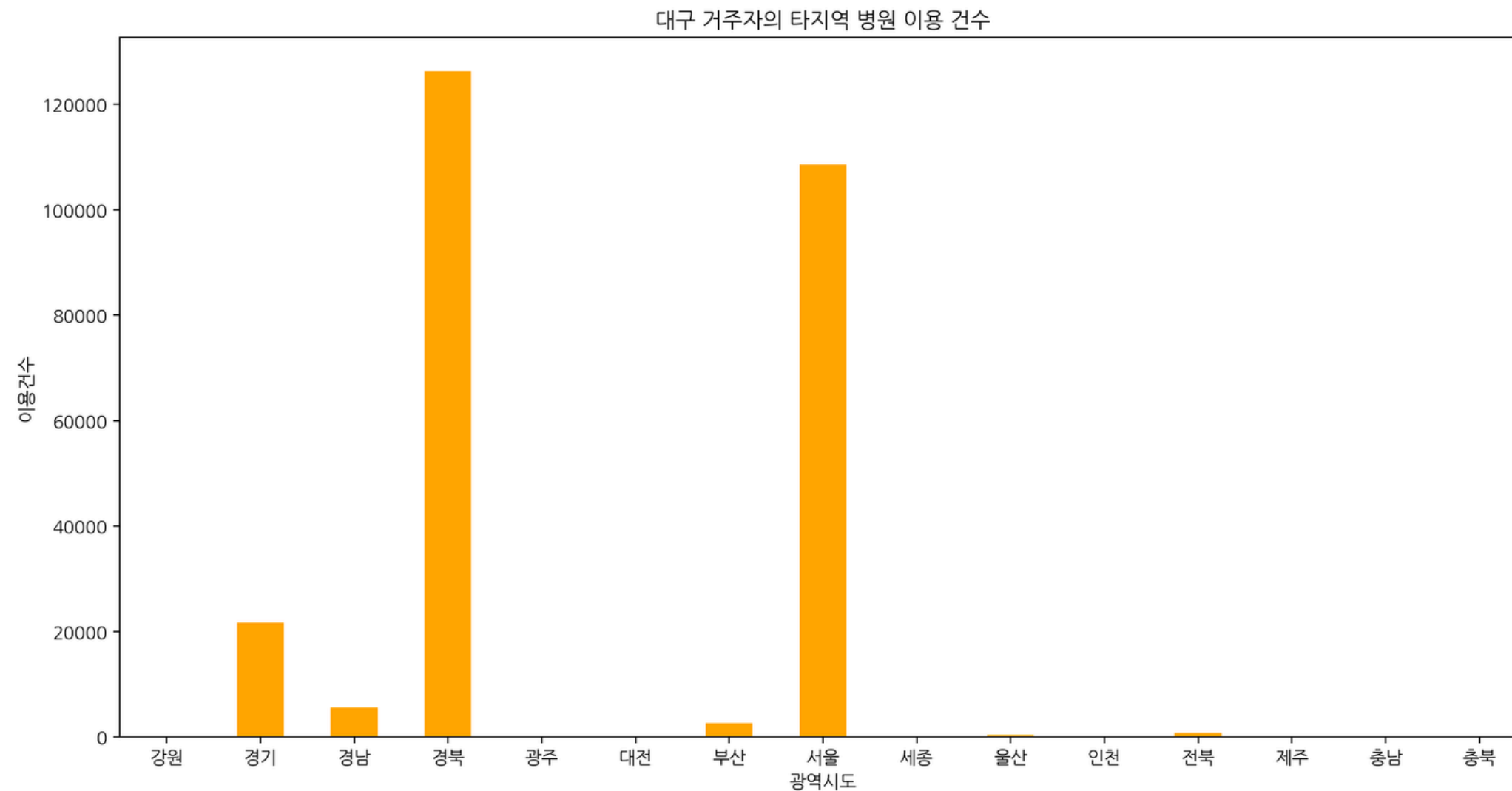


```
1 inner_p_hospital = hp[(hp['내외지인'] == 1)]
2 inner_p_outside_daegu = inner_p_hospital[inner_p_hospital['가맹점_광역시도'] != '대구']
3
4 plt.figure(figsize=(14, 7))
5
6 inner_p_outside_daegu.groupby('가맹점_광역시도')['CNT'].sum().plot(kind='bar', color='orange')
7 plt.title('대구 거주자의 타지역 병원 이용 건수')
8 plt.ylabel('이용건수')
9 plt.xlabel('광역시도')
10 plt.xticks(rotation=0)
11 plt.show()
```

- 대구 시민들의 타지역 병원 이용은 경북, 경남지역과 수도권에서 이용이 대다수

대구와 다른 지역간 분석

수도권 병원 선택이유?



수도권 병원 선택이유

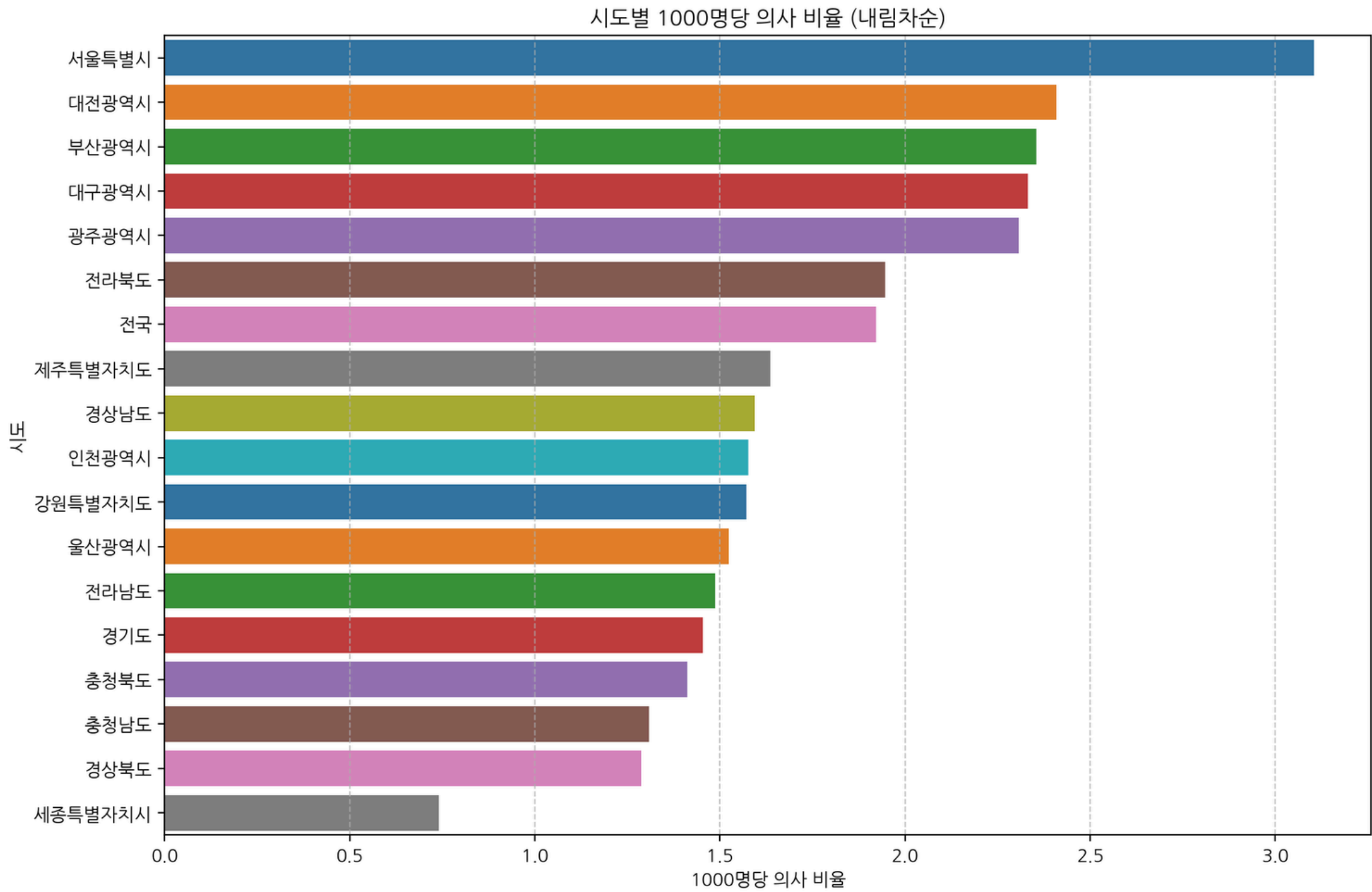
- 수도권 의사들의 전문성에 대한 신뢰가 높아서
- 높은 병원 인지도
- 전문기관
- 큰 병원 규모

지방 병원을 선택하지 않은 이유

- 의사 전문성에 대한 신뢰도가 낮아서
- 의료진 부족
- 작은 병원 규모
- 최신 의료 장비를 이용할 수 없음
- 체계적이지 않은 치료 절차

대구와 다른 지역간 분석

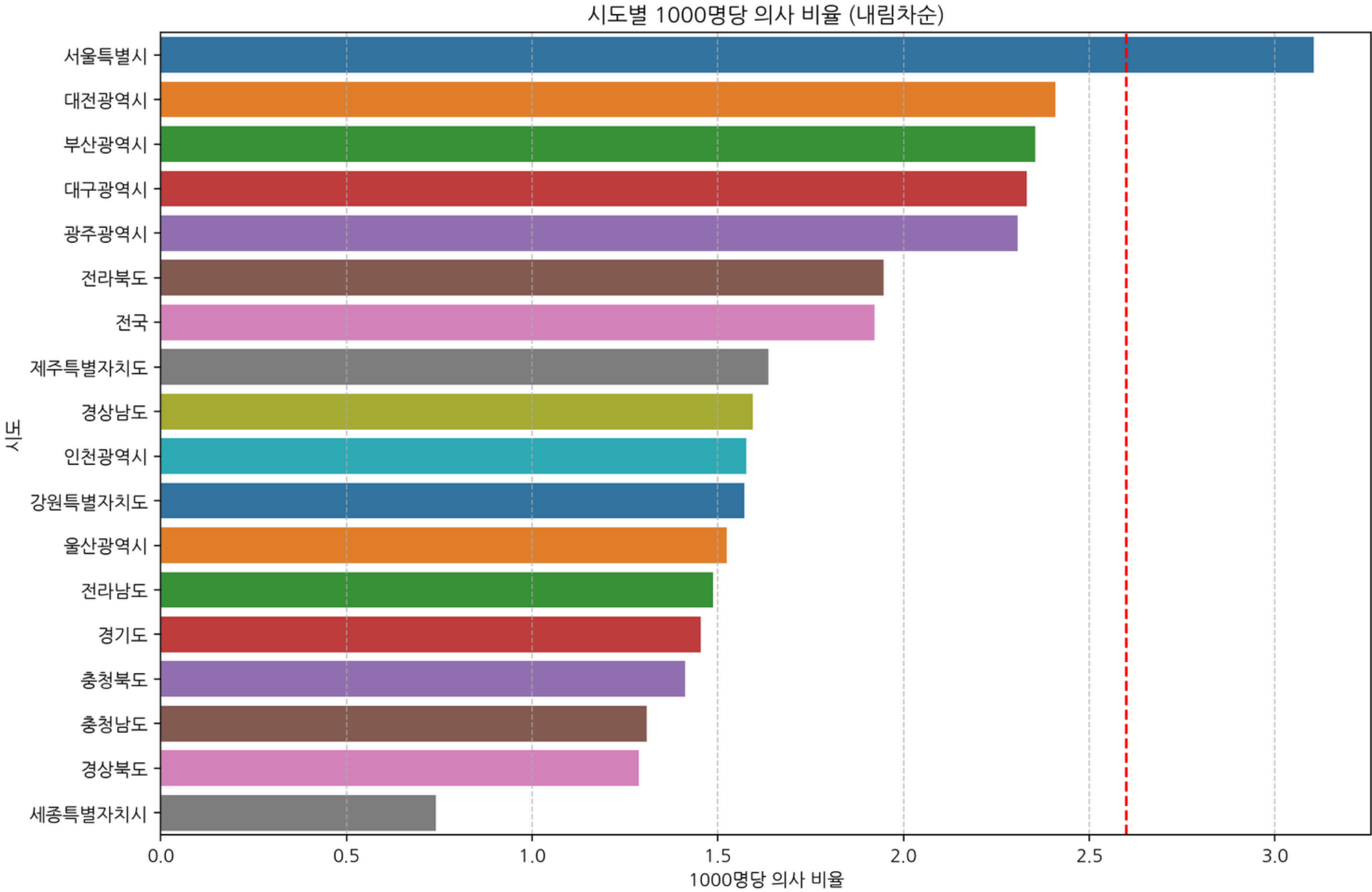
시도별 1000명당 의사 비율



- 대구의 1,000명당 의사 비율은 전국 시도 중 4번째로 높음

대구와 다른 지역간 분석

시도별 1000명당 의사 비율

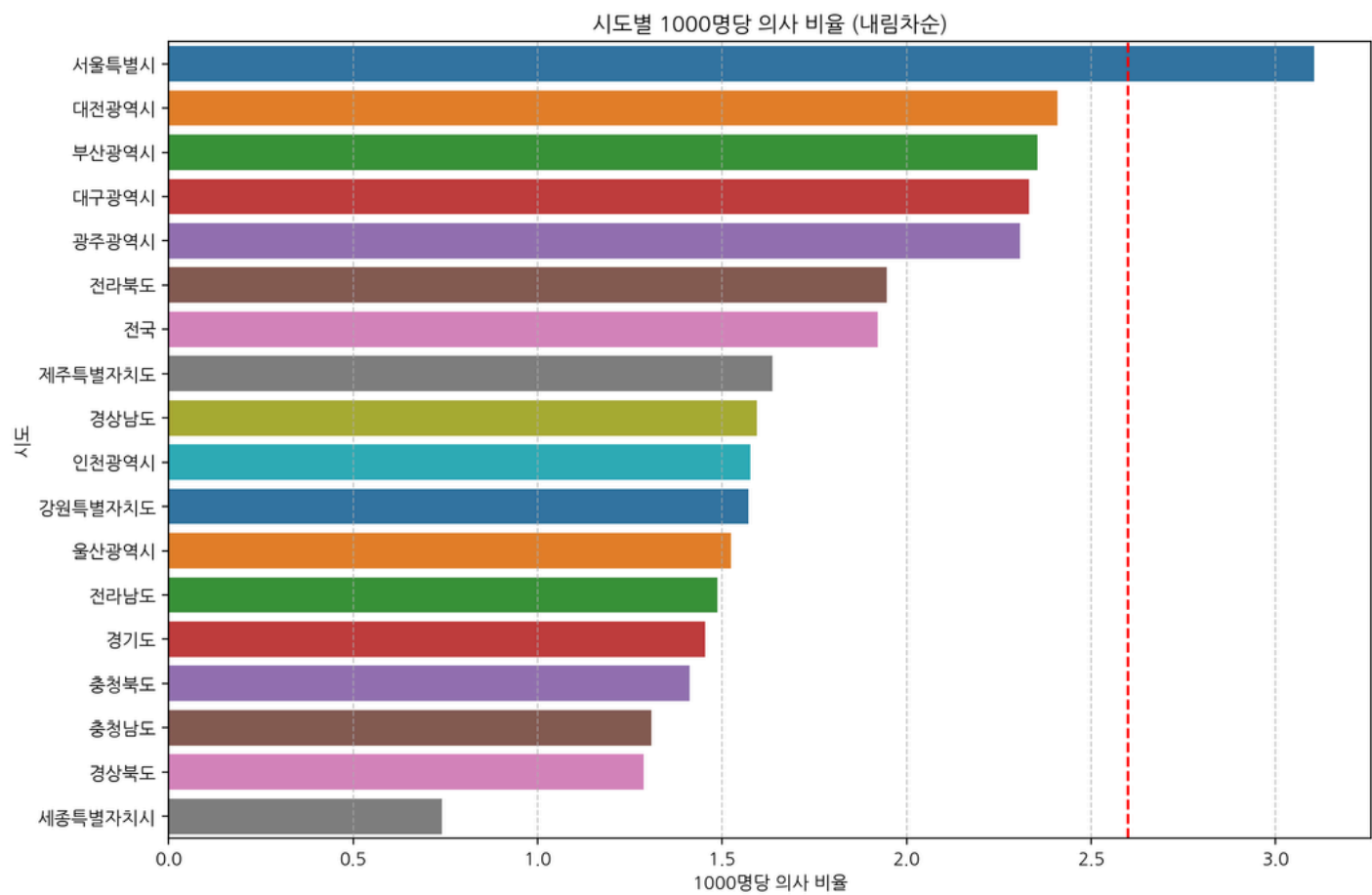


시도
-- OECD 평균 (1000명당 2.60명)

- 하지만 서울을 제외한 모든 시도가 oecd평균에 못미침
- 전국 상위권에 위치한 대구지만 oecd 평균을 넘기위해 더 많은 의사가 필요

대구와 다른 지역간 분석

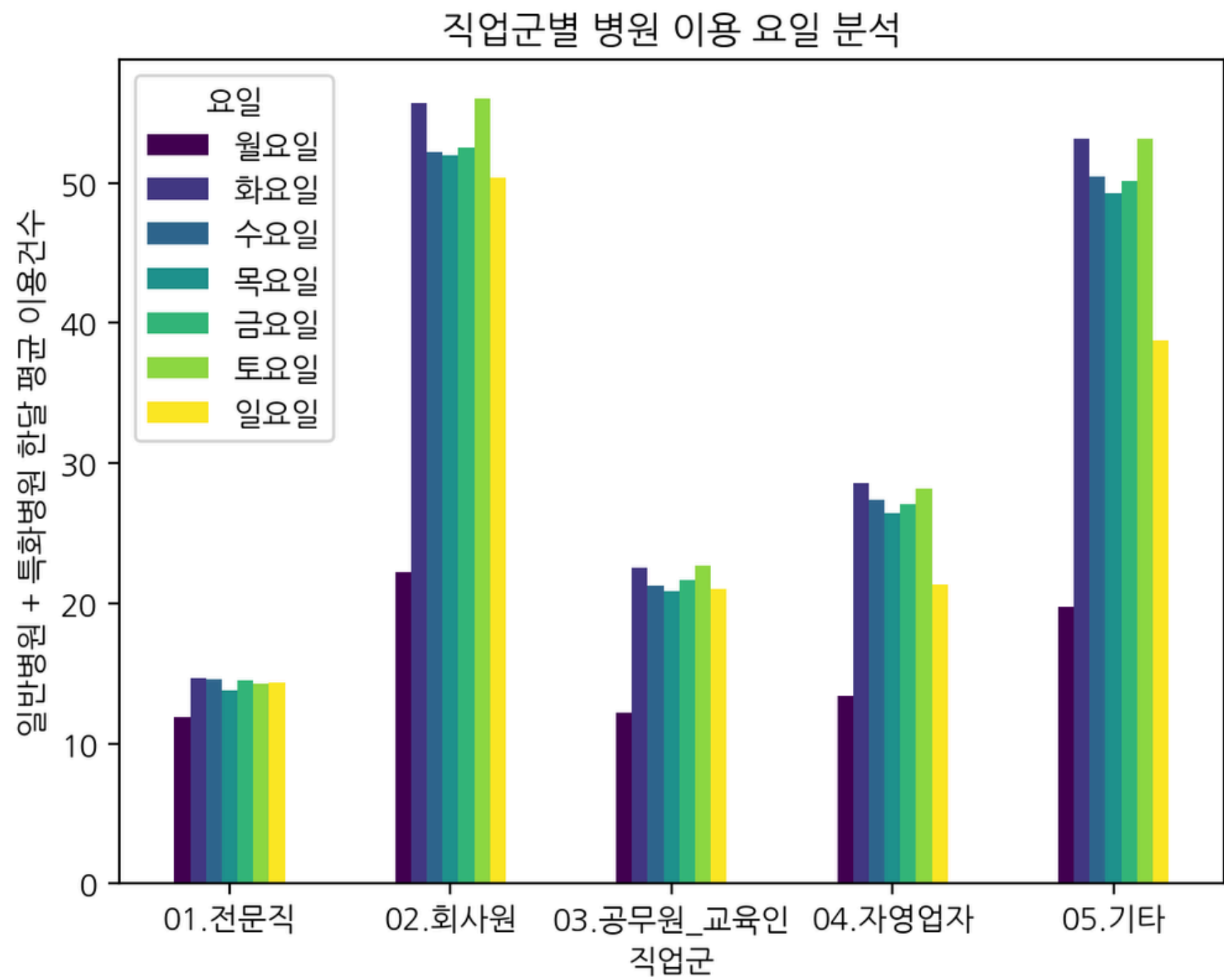
시도별 1000명당 의사 비율



```
1 merged_df = pd.merge(docc, kp, on='시도')
2
3 merged_df['1000명당 의사 비율'] = (merged_df['전체'] / merged_df['2023']) * 1000
4
5 oecd_average = 2.6 / 1
6
7 plt.figure(figsize=(12, 8))
8
9 sns.barplot(x='1000명당 의사 비율', y='시도', hue='시도', data=merged_df, sort_values('1000명당 의사 비율', ascending=False), dodge=False, palette="tab10")
10
11 plt.axvline(x=oecd_average, color='red', linestyle='--', label=f'OECD 평균 (1000명당 {oecd_average:.2f}명)')
12
13 plt.xlabel('1000명당 의사 비율')
14 plt.title('시도별 1000명당 의사 비율 (내림차순)')
15 plt.grid(True, axis='x', linestyle='--', alpha=0.7)
16
17 plt.legend(title='시도', bbox_to_anchor=(1.05, 1), loc='upper left')
18 plt.show()
```

직업군별 병원 방문 분석

직업군별 일반병원 + 특화병원 한달 이용건수 분석



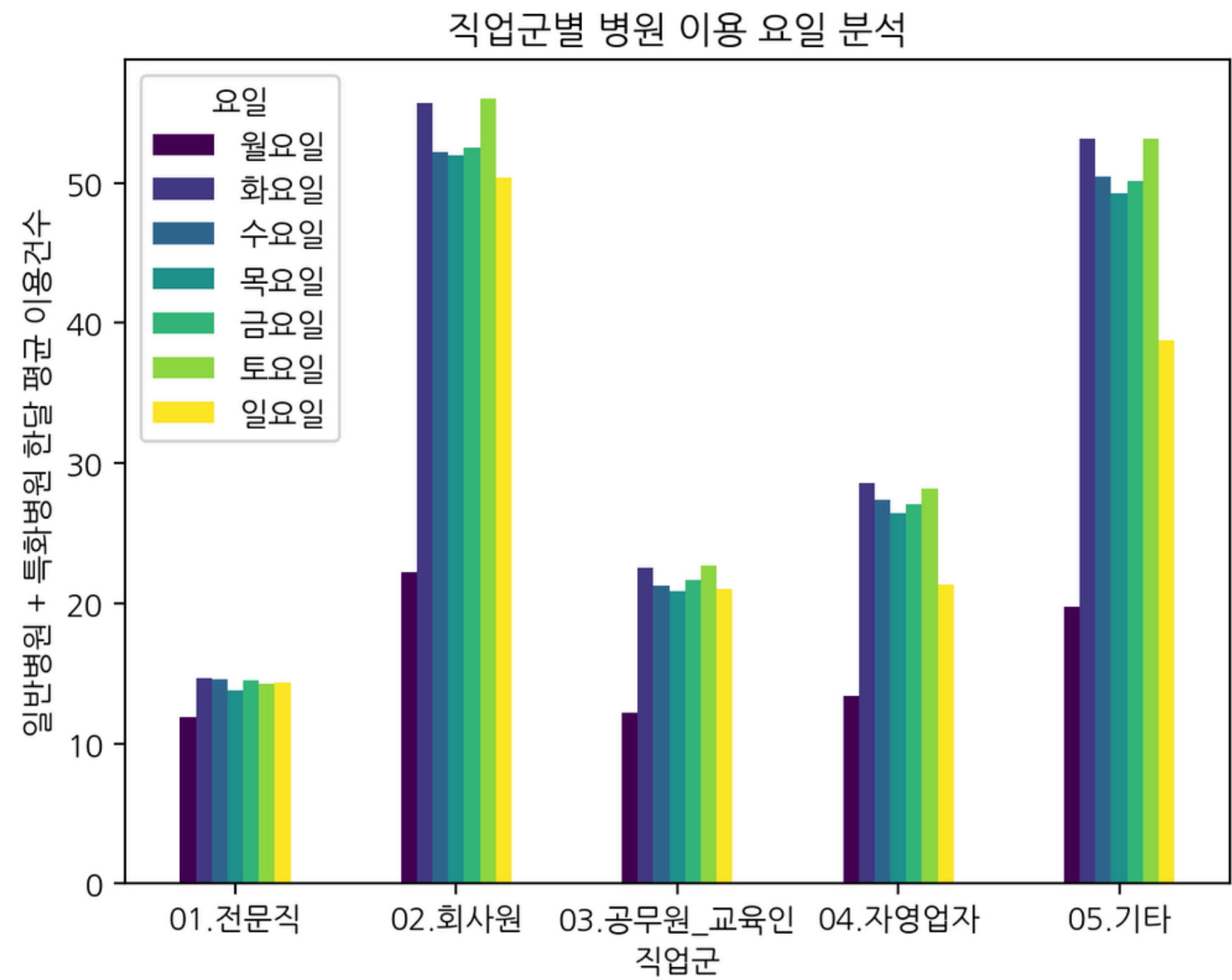
종합병원 및 의원 운영시간 (10곳 중 8곳)

- 월요일~금요일: 오전 8~9시 ~ 오후 5시30분~ 6시
- 토요일: 오전 9시 ~ 오후 2시
- 일요일 및 공휴일: 휴무

- 모든 직업군이 월요일보다 토요일,일요일 같은 주말에 병원을 이용하는 많음
- 이에따라 기존에 운영시간을 조정할 필요가 있어보임

직업군별 병원 방문 분석

직업군별 일반병원 + 특화병원 한달 이용건수 분석



```
1 week_order = ['월요일', '화요일', '수요일', '목요일', '금요일', '토요일', '일요일']
2 hp1 = hp.groupby(['직업군', 'WEEK_GROUP'])['CNT'].mean().unstack()
3
4 hp1 = hp1.reindex(columns=week_order, level='WEEK_GROUP')
5
6 plt.figure(figsize=(12, 8))
7 hp1.plot(kind='bar', stacked=False, cmap='viridis')
8 plt.title('직업군별 병원 이용 요일 분석')
9 plt.ylabel('일반병원 + 특화병원 한달 평균 이용건수')
10 plt.xlabel('직업군')
11 plt.legend(title='요일')
12 plt.xticks(rotation=0)
13 plt.show()
```

“

결론

대구 지역의 의료 서비스 현황과 주민들의 건강 소비 패턴에 대한 종합적인 이해를 도출할 수 있었음. 대구시는 전체적으로 양호한 의료 인프라를 보유하고 있지만, 구군별로 상당한 의료 불균형이 존재, 특정 의료 서비스(예: 산부인과, 소아과 등)의 접근성이 취약한 지역이 여전히 존재함을 확인할 수 있었음
특히, 직업군별 병원 이용 패턴 분석에서는 대부분의 직업군에서 주말에 병원 이용 빈도가 높다는 점이 확인되었으며, 이는 기존 병원 운영 시간의 조정 필요성을 시사

”

“

제언

- 의료 인프라 확충: 대구 내 의료 접근성이 취약한 지역에 대한 종합병원 및 전문 병원의 확충이 필요 특히 산부인과, 소아과와 같은 필수 의료 서비스의 공급을 강화해야 함
- 의료 서비스 운영 시간 조정: 주말에 병원 이용이 집중되는 경향을 반영하여, 병원 운영 시간을 유연하게 조정하는 방안을 검토해야함 이를 통해 다양한 직업군의 주민들이 더욱 쉽게 의료 서비스를 이용할 수 있을 것으로 기대됨
- 정책적 대응: 대구 시민들의 타지역 병원 이용을 줄이기 위해, 대구 내 병원들의 전문성 강화와 최신 의료 장비 도입 등을 통해 병원에 대한 신뢰도를 높이는 것이 필요함 또한, 의료진의 전문성 향상을 위한 지속적인 교육과 지원이 필수적임

”

“

구체적인 활용 방안 I, II

- 직장인의 병원 이용 문제
 - 병원 운영시간인 오전 8~6시에 병원 이용이 어려움 (특히 회사원, 자영업자)
- 해결 방안
 - 주말 진료 시간 확장
 - 평일 야간 진료 운영
 - 병원 간 협력 체계 구축 (진료과목 분담)
- 지역간 의료 불균형
 - 병상 수와 전문 의료기관 부족 (산부인과, 소아과 등)
- 해결 방안
 - 필수 진료과목을 갖춘 종합병원 설립
 - 기존 병원의 시설 확충

”

“

구체적인 활용 방안 III, IIII

- 특정 직업군 맞춤형 의료 정책
 - 직업군에 따른 병원 이용 패턴 반영한 의료 서비스 제공
- 예시
 - 특정 직업군이 많은 지역에 맞춤형 의료 운영 시간 도입
- 타지역 병원 이용 증가 문제
 - 대구 시민이 전문성을 이유로 타지역 병원 이용
- 해결 방안
 - 대구 내 병원의 전문 진료 분야 강화
 - 최신 의료 장비 도입 및 의료진 교육 강화

”

“

현실적 한계 및 추가 고려사항

- 정책 실행의 경제적 한계: 제언된 정책 중 의료 인프라 확충이나 의료진 교육 강화는 상당한 재정적 투자가 필요함 대구 지역 예산과 국가 지원이 이를 뒷받침할 수 있는지 여부를 고려해야 함
- 지역 주민들의 수용성: 병원 운영 시간을 확대하거나 전문 진료 분야를 강화하는 과정에서 주민들의 수요와 실제 이용 패턴이 정책적 변화에 부합하는지 지속적인 모니터링이 필요합니다.
수요 예측이 부정확할 경우, 예상치 못한 결과가 발생할 수 있음
- 전문 의료진 확보의 어려움: 대구 내에서 전문 의료진을 유치하는 것은 다른 대도시와의 경쟁에서 다소 어려울 수 있습니다. 이를 위해 대구 지역만의 의료진 유입 전략이 필요하며, 장기적인 인재 육성 계획도 함께 고려해야 함

”

제 6회 대구 빅데이터 분석 경진대회

감사합니다

세명컴퓨터고등학교 박진우

