
강원소방용수

곽명빈 · 김주형 · 김도훈 · T A N G O C H A

최적압자선정

CONTENTS

- 01** 개요
- 상황 분석
 - 제안

- 02** 데이터
- 데이터 수집
 - 데이터 분석

- 03** 결론 및 활용방안
- 결과 설명
 - 활용 방안
 - 기대 효과

팀원 역할

팀 대표

20173204 곽명빈

슬라이드 12 ~ 29

데이터수집, 데이터 EDA, 전처리 및 분석, 결과보고서

팀원

20173218 김주형

슬라이드 30 ~ 45

데이터수집, 분석 결과 해석 및 시각화, 결론 및 한계점

팀원

20181203 김도훈

슬라이드 1 ~ 11

데이터수집, 자료조사(상황분석 및 제안), PPT 제작 (전체)

팀원

20198095 TANGOC HA

슬라이드

데이터수집, 자료조사, 결과보고서

01

개요

- 상황분석
- 제안

01 개요

상황 분석

[ISSUE] 동해안 덮친 최악 산불... 서울 면적 3분의 1 초토화

서울 면적 3분의 1 불타... 역대 두 번째 피해

최근 화두된 강원지역을 비롯한 역대급 산불

강원 강릉과 영월 등 동해안 곳곳에서 발생한 산불



01개요

상황 분석

인제소방서, 2022년 소방용수시설 운영위원회 개최

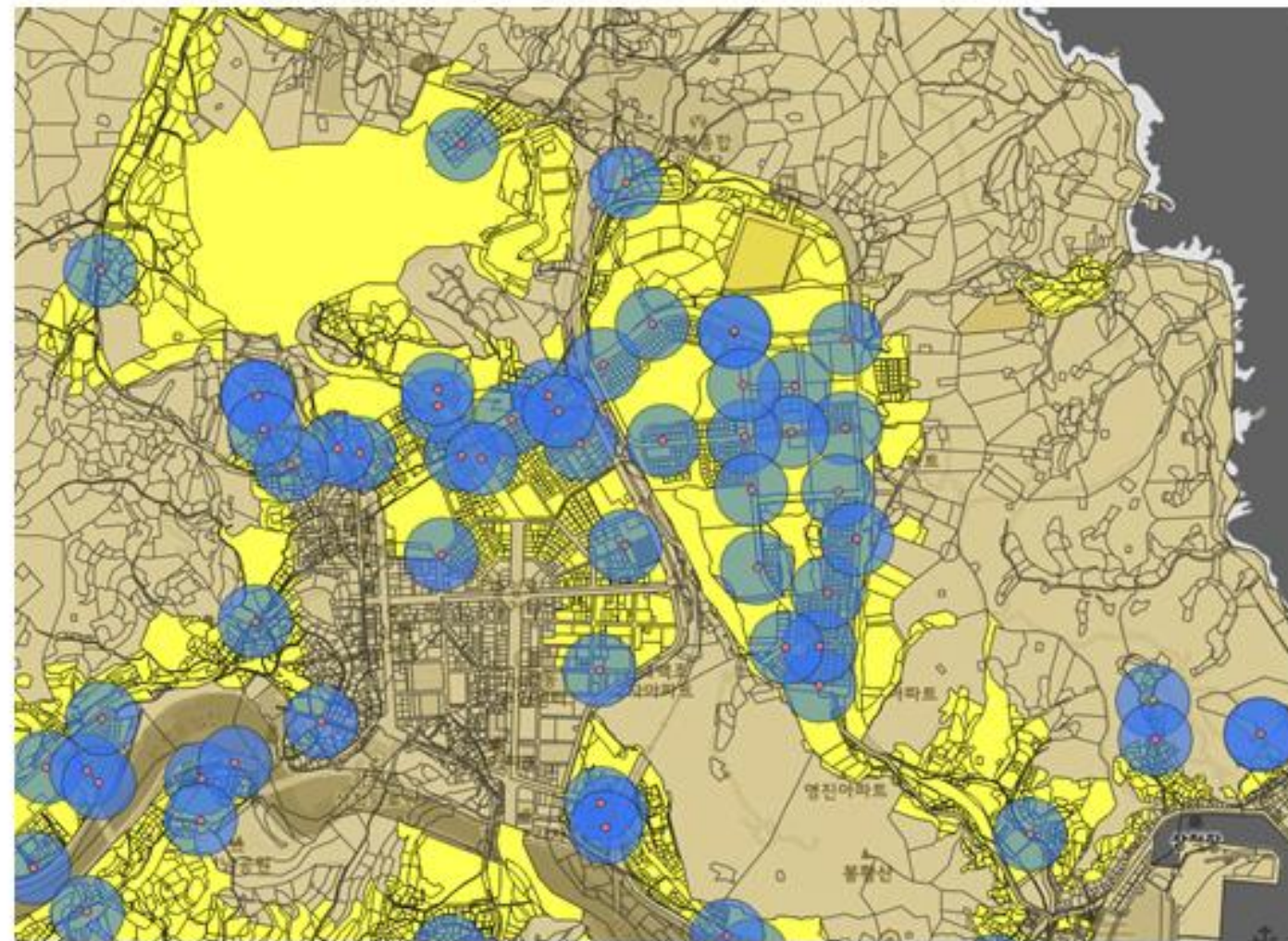
정현희 기자 | 기사입력 2022/04/28 [11:30]

“제2의 강원산불 막는다”...강원지역, 소방 용수 보강 추진

소방청, 빅데이터 활용 소방용수 취약지역 분석

소방 용수 시설 운용 및 보강 관심

강원지역 및 소방청에서는 소방용수시설의 효율적 관리를 위해 운영위원회 개최, 빅데이터를 활용하여 소방용수시설을 보강하는 계획을 추진



소방용수시설 및 상수관 설치 취약지역(삼척시) / 사진 = 소방청 제공.

01 개요

제안

화재취약지점과 최대한 가까이 위치해 있을 수 있게
소방용수시설의 최적의 위치를 선정하는 목표로 함.





공공데이터를 통한
강원도 화재취약
지역 파악



데이터 파악 후
취약점 파악 및
결과 도출



**소방용수시설
최적입지 선정**

02

데이터

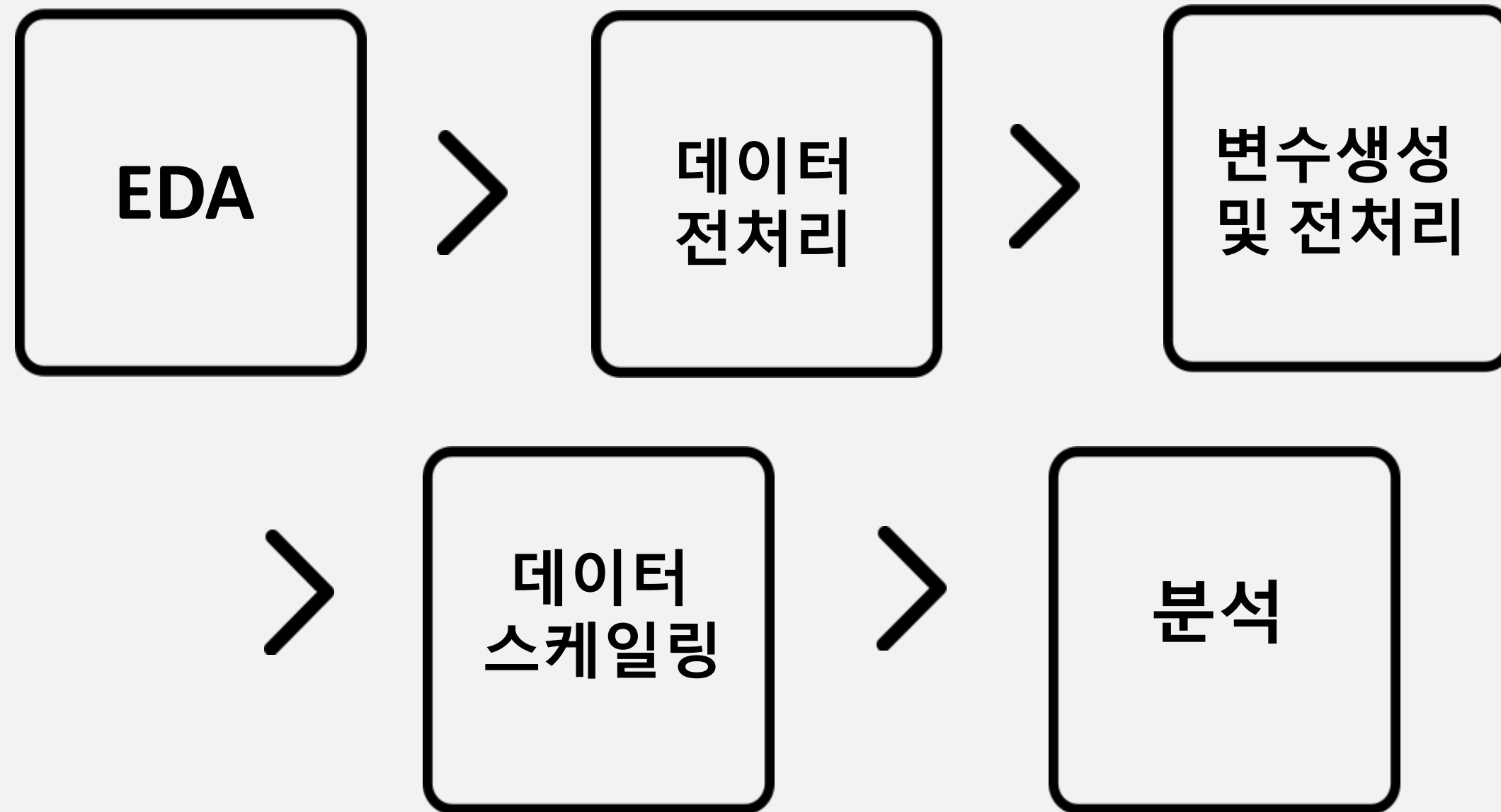
- 데이터 수집
- 데이터 분석

02 데이터

데이터 수집

데이터	데이터 설명	특성	출처
강원도 소방본부 소방용수시설	소방용수시설 좌표	시군구명 위도경도 배관 ⋮	
구조 종별 신고 재난 데이터	구조종별 신고 재난 데이터 좌표	사고원인 위도경도 현장도착시,분 ⋮	소방안전 빅데이터플랫폼
화재 종별 신고 재난 데이터	화재 종별 신고 재난 데이터	사고원인 위도경도 현장도착시,분 ⋮	소방안전 빅데이터플랫폼
강원도 읍면동별 주민등록인구	강원도 읍면동별 주민등록인구	시군 행정구역 총인구수 ⋮	
강원도 건물통합정보	건물공간정보	-	

데이터 분석 과정



02 데이터

1. EDA

지도 시각화 1 – Python

소방용수시설 및 재난데이터 군집확인

```
import pandas as pd
import folium as g
from folium.plugins import MarkerCluster
```

```
alarm = pd.read_csv('./data/강원도_강원도소방본부_소방용수시설.csv', encoding = 'cp949')
df = pd.read_csv('./data/구조 종별 신고.재난 데이터.csv', encoding='utf-8')
```

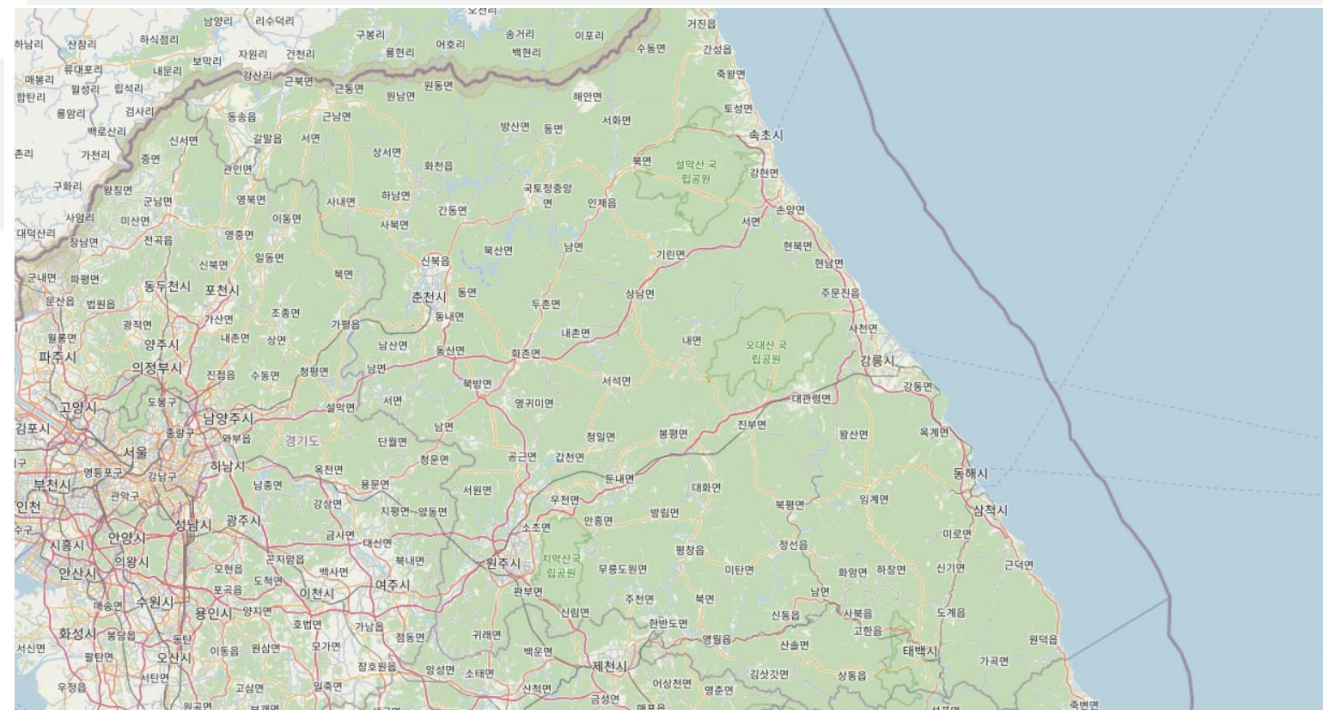
필요 라이브러리 정의 후 데이터 로드

folium, pandas

```
m = g.Map(
    ... location=[37.8013, 128.3745],
    ... zoom_start = 9
)

m.add_child(g.LatlngPopup()) ..... # 지도에서 클릭하는 곳의 경도·위도·좌표를 표시

m
```

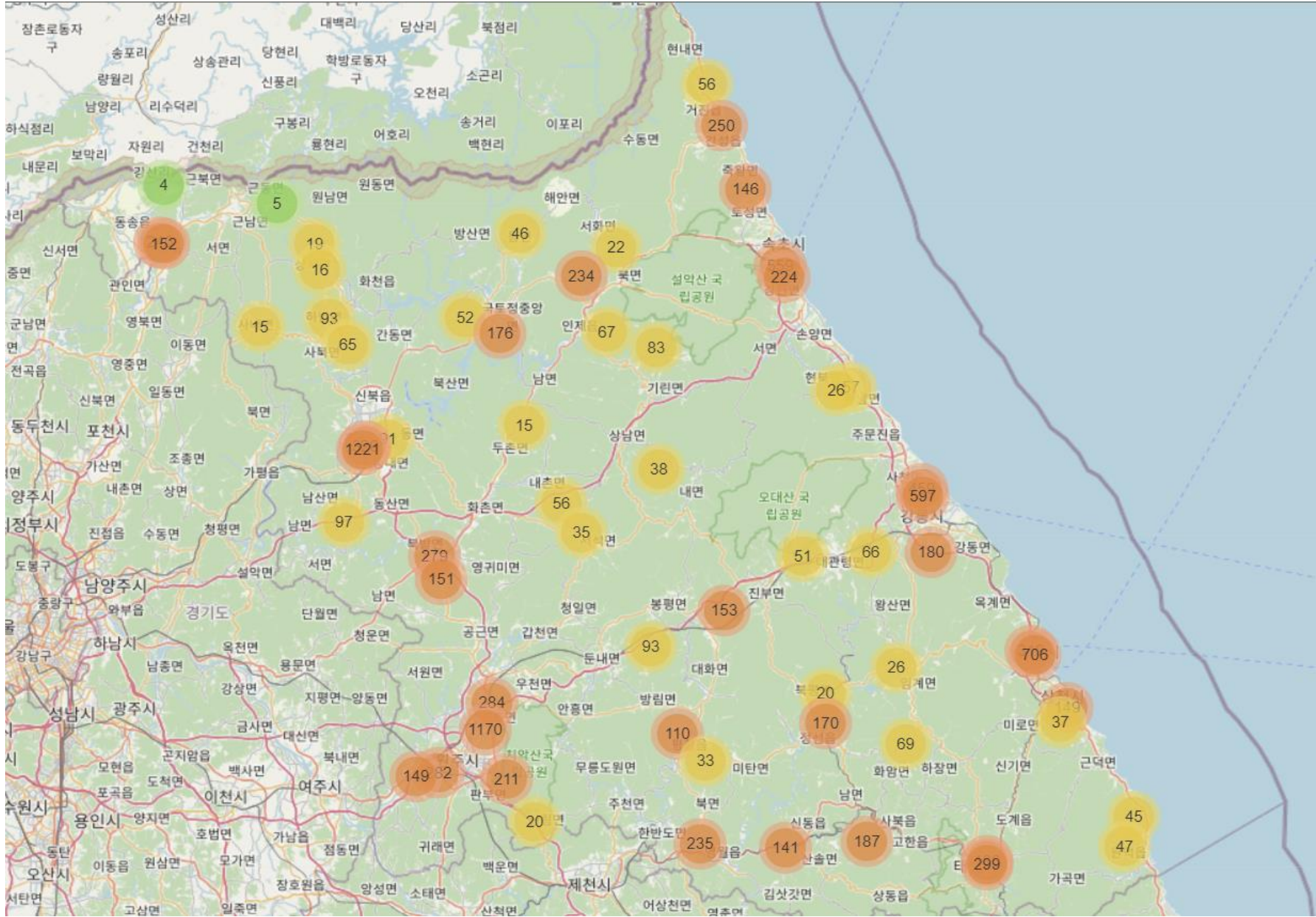


강원도 중심 좌표와 zoom 설정

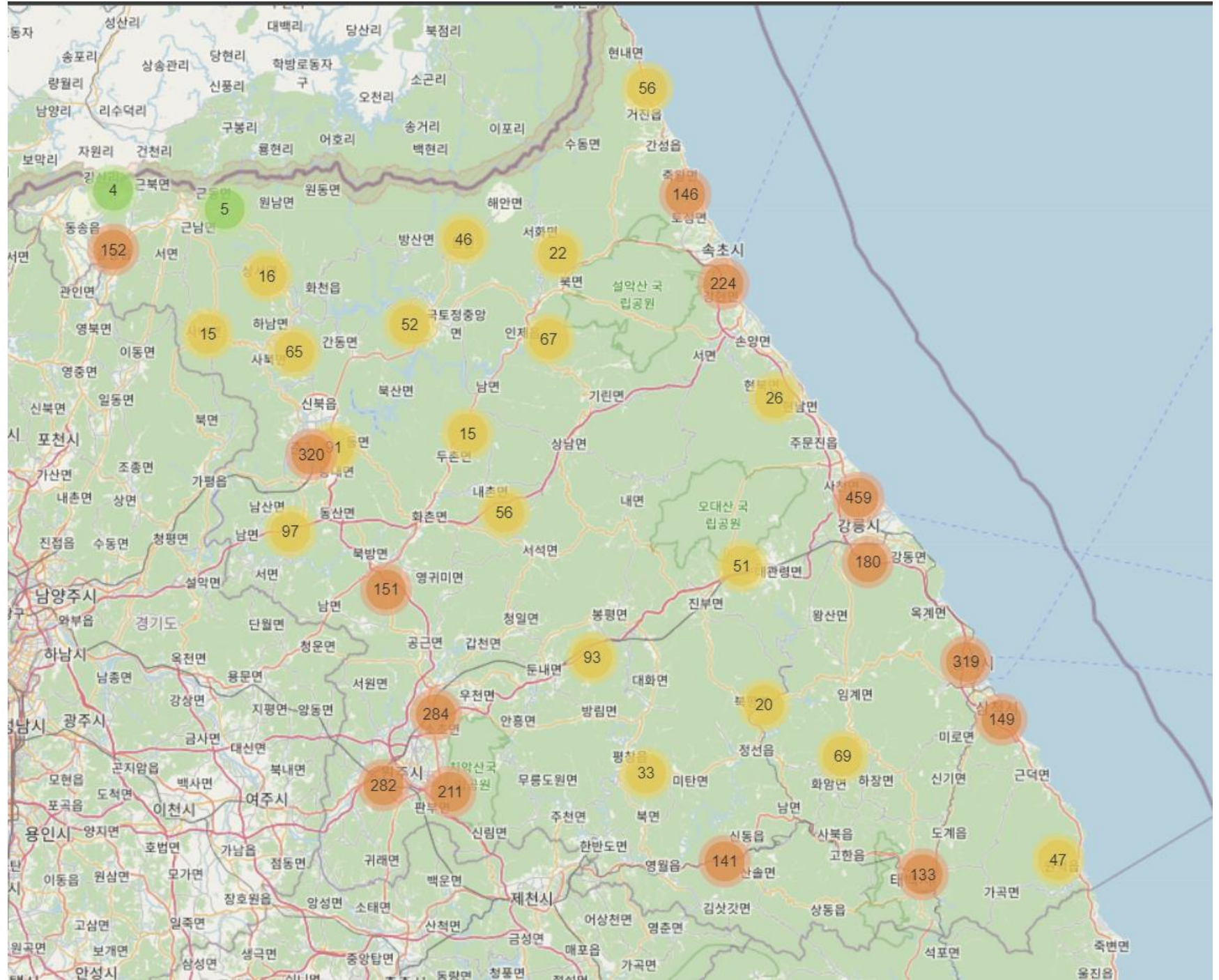
```
def plot(data):
    # 리스트를 이용해 여러 행의 데이터를 위,경도로 묶음
    locations = list(zip(data.LOC_INFO_X, data.LOC_INFO_Y))

    cluster = MarkerCluster(locations = locations)
    # 지도에 클러스터를 추가.
    cluster.add_to(m)
```

MarkerCluster 함수 생성



소방용수시설, 화재발생위치 군집



-> 소방용수시설, 화재발생위치 군집 대략적 파악

02 데이터

2 .데이터 전처리

```
# 필요 라이브러리 정의
import pandas as pd
```

✓ 0.8s

Python

전처리 필요 라이브러리 정의

```
# 데이터 정의
```

강원도 소방용수시설 데이터 = 'alarm'

```
alarm = pd.read_csv('./data/강원도_강원도소방본부_소방용수시설.csv', encoding = 'cp949')
```

```
df = pd.read_csv('./data/구조_종별_신고_재난_데이터.csv', encoding = 'utf-8')
```

강원도 구조 종별 신고, 재난 데이터 = 'df'

```
# 데이터 화재관련 값만 추출
```

```
df = df[(df['ACDNT_CAUSE'] == '기타화재') | (df['ACDNT_CAUSE'] == '화약류') | (df['ACDNT_CAUSE'] == '방화')
        | (df['ACDNT_CAUSE'] == '폭발물') | (df['ACDNT_CAUSE'] == '화재확인')]
```

df 화재관련 값 추출 => df 재지정

```
# 누락값 확인 및 누락값이 있는 열 삭제
```

```
df.isnull().sum()
```

df 누락값 확인 후 제거

```
df = df.dropna(axis = 1)
```

✓ 0.3s

Python

02 데이터

2 .데이터 전처리

alarm.head()

✓ 0.7s

Python

	시설번호	시설유형코드	시군구명	시군구코드	소재지번주소	위도	경도	안전센터명	보호틀유무	사용가능여부	설치연도	배관깊이	출수압력	배관지름	관할기관명	데이터기준일자
0	강촌001호	1	춘천시	42110	강원도 춘천시 남산면 창촌리 145-1	37.786189	127.649237	강촌안전센터	N	Y	2007	0.7	7.0	100	춘천소방서	2021-11-19
1	강촌002호	1	춘천시	42110	강원도 춘천시 남산면 창촌리 147-1	37.786725	127.647929	강촌안전센터	N	Y	2013	0.7	8.0	100	춘천소방서	2021-11-19
2	강촌003호	1	춘천시	42110	강원도 춘천시 남산면 창촌리 126-9	37.785523	127.646686	강촌안전센터	Y	Y	2013	0.7	8.0	100	춘천소방서	2021-11-19
3	강촌004호	1	춘천시	42110	강원도 춘천시 남산면 창촌리 219-15	37.784756	127.643488	강촌안전센터	Y	Y	2013	0.8	8.0	100	춘천소방서	2021-11-19
4	강촌005호	1	춘천시	42110	강원도 춘천시 남산면 창촌리 238-2	37.781550	127.643992	강촌안전센터	Y	Y	2013	0.7	5.5	100	춘천소방서	2021-11-19

alarm 데이터 확인

df.head()

✓ 0.1s

Python

	MSFRTN_RESC_REPRT_NO	ACDNT_CAUSE	PRCS_RESULT_SE_NM	DCLR_YMD	DCLR_TM	DCLR_YR	SEASON_SE_NM	QTR_SE	DCLR_MNTH	DCLR_DAY	...	SIDO_NM	SIGI
3	20204207506S00069	기타화재	기타	20201011	143700	2020	가을	4	10	11	...	강원도	
65	20214205202S00423	기타화재	오인(구조)	20210619	115100	2021	여름	2	6	19	...	강원도	
68	20214205202S00129	기타화재	오인(구조)	20210414	183000	2021	봄	2	4	14	...	강원도	
80	20204205103S00075	기타화재	안전조치	20201201	134800	2020	겨울	4	12	1	...	강원도	
101	20204205202S00074	기타화재	인명검색	20200212	92900	2020	겨울	1	2	12	...	강원도	

5 rows × 36 columns

df 데이터 확인

=> 공통 행을 만들어야 하므로 읍면동별 분석을 위한 ‘alarm’의 읍면동 데이터 추가

02 데이터

2 .데이터 전처리

```
# 소재지번호주소를 이용하여 읍면동 columns을 만들어야함
alarm.head()
# 소재지번호주소의 7번째 이후의 데이터를 만들어서 저장
alarm['EMD_NM'] = alarm['소재지번호주소'].str[7:]
alarm.head(5)

alarm['EMD_NM'] = alarm['EMD_NM'].str.lstrip() # 앞 공백을 제거
alarm['EMD_NM'].str.split(" ").head() # 공백을 기준으로 분할
dong = alarm['EMD_NM'].str.split(" ", expand=True) # expand = True 데이터 프레임 형태로 저장
dong

# 동의 첫번째 열을 EMD_NM로 지정
alarm['EMD_NM'] = dong[0]
alarm.head()

alarm['EMD_NM'].unique() # 일부 동이 띄어쓰기가 안되있어서 나누어 지지않음
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace(pat=r'^\w', repl='', regex=True) # 특수문자 제거
alarm['EMD_NM'].unique() # 숫자를 정규표현식을 이용하려 했지만 중앙로 n가 등 특수한 경우가 존재하여 제거 x 하나하나 지워야겠음
```

✓ 0.7s

Python

강원도 소방용수시설 데이터(alarm)에 읍면동 데이터 x, 소재지번호주소 이용

소재지번호주소
강원도 춘천시 남산면 창촌리 145-1
강원도 춘천시 남산면 창촌리 147-1
강원도 춘천시 남산면 창촌리 126-9
강원도 춘천시 남산면 창촌리 219-15
강원도 춘천시 남산면 창촌리 238-2
강원도 춘천시 남산면 창촌리 300-15
강원도 춘천시 남산면 창촌리 300-11

소재지번호주소				
강원도	춘천시	남산면	창촌리	145-1
강원도	춘천시	남산면	창촌리	147-1
강원도	춘천시	남산면	창촌리	126-9
강원도	춘천시	남산면	창촌리	219-15
강원도	춘천시	남산면	창촌리	238-2
강원도	춘천시	남산면	창촌리	300-15
강원도	춘천시	남산면	창촌리	300-11



EMD_NM	
남산면	창촌리
145-1	
남산면	창촌리
147-1	
남산면	창촌리
126-9	
남산면	창촌리
219-15	
남산면	창촌리
238-2	



공백 기준 분할 후 'dong' 저장

	0	1	2	3
0	남산면	창촌리	145-1	None
1	남산면	창촌리	147-1	None
2	남산면	창촌리	126-9	None
3	남산면	창촌리	219-15	None
4	남산면	창촌리	238-2	None
...

dong[0] = 'EMD_NM' 저장

소재지번호주소의 7번째 이후의 단어 = 'EMD_NM'

02 데이터

2 .데이터 전처리

```
# 전처리시 문제가 있던행을 변경
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('죽헌동93317', '죽헌동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('교동18991', '교동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('금학동73', '금학동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('성남동1016', '성남동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('성남동1114', '성남동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('성남동11310', '성남동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('발한동3099', '발한동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('이로동1021', '이로동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('이로동194', '이로동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('구미동4821', '구미동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('단구동1567', '단구동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('단구동15797', '단구동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('단구동15881', '단구동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('태장동1293', '태장동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('대포동957', '대포동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('포남동109721', '포남동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('발한동238', '발한동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('목호진동2447', '목호진동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('단구동15681', '단구동')
alarm['EMD_NM'] = alarm['EMD_NM'].str.replace('북평동128', '북평동')
```

전처리 문제되는 행 변경

```
# 필요없는 데이터 columns 제거
del alarm['시설유형코드']
del alarm['시설번호']
del alarm['시군구코드']
del alarm['데이터기준일자']
```

불필요 열 제거

```
# 시군구읍으로 해야함 => 중복되는 게 있음
```

```
alarm['Full_NM'] = alarm['시군구명'] + ' ' + alarm['EMD_NM']
```

```
df['Full_NM'] = df['SIGUNGU_NM'] + ' ' + df['EMD_NM']
```

열이름 중복, 두 데이터에 ['Full_NM'] 새로운 열 생성

3. 변수생성 및 전처리

```
# 개수를 데이터프레임으로 변환
water = pd.DataFrame(alarm['Full_NM'].value_counts())
fire = pd.DataFrame(df['Full_NM'].value_counts())

# 인덱스를 초기화
water = water.reset_index()
fire = fire.reset_index()

# columns 이름 변경
water.rename(columns={'Full_NM': '용수시설개수'}, inplace=True)
fire.rename(columns={'Full_NM': '화재발생횟수'}, inplace=True)

# 공통된 행 합치기(index 기준으로)
dat = pd.merge(water, fire, on = 'index')

dat.rename(columns={'index': 'Full_NM'}, inplace=True)

dat
✓ 0.8s
```

각 읍면동별 용수시설의 개수, 화재발생 횟수데이터 groupby

읍면동 기준 공통 행 merge

	Full_NM	용수시설개수	화재발생횟수
0	홍천군 홍천읍	183	63
1	횡성군 횡성읍	141	62
2	영월군 영월읍	137	54
3	원주시 지정면	127	60
4	태백시 황지동	112	49
...

읍면동별 용수시설개수, 화재발생횟수 데이터

02 데이터

3. 변수생성 및 전처리

```
# 위에서 결측값이 있는데 제거했더니 도착시간이 안나와서 새로 load
df = pd.read_csv('./data/구조 종별 신고.재난 데이터.csv', encoding= 'utf-8')

# 데이터 화재관련 값만 추출

df = df[(df['ACDNT_CAUSE'] == '기타화재') | (df['ACDNT_CAUSE'] == '화약류') | (df['ACDNT_CAUSE'] == '방화')
        | (df['ACDNT_CAUSE'] == '폭발물') | (df['ACDNT_CAUSE'] == '화재확인')]

df['Full_NM'] = df['SIGUNGU_NM'] + ' ' + df['EMD_NM']

# 시간*60 + 분으로 해서 하나의 데이터로 만들었음
s_time = (df['DSP_HOUR']*60) + (df['DSP_MIN'])
a_time = (df['SPT_ARVL_HOUR']*60) + (df['SPT_ARVL_MIN'])

# 도착시간에 결측치가 있어서 0분으로 처리
a_time = a_time.fillna(s_time)

# 걸린시간
time = a_time - s_time

time = pd.DataFrame(time)
df['time'] = time

✓ 0.3s
```

구조 종별 신고, 재난 데이터 -> 시간 항목 추출

시간과 분의 단위가 달라 하나의 단위인 '분'으로 통일

=> 시간*60 + 분

도착시간 - 출발시간 = 걸린시간

```
# EMD_NM 별 화재도착시간 만들기
# 평균화재도착시간으로 하려했는데 23시출발 - 0시 도착이면 값이 이상해져서 중앙값으로 하였음.
time = pd.DataFrame(df.groupby(df['Full_NM'])['time'].median())
time = time.reset_index()
time

✓ 0.6s
```

읍면동별 화재도착시간 생성

그룹별 중앙값으로 'time' 생성

02 데이터

3 . 변수생성 및 전처리

	Full_NM	time
0	강릉시 강동면	13.0
1	강릉시 강문동	5.0
2	강릉시 견소동	6.0
3	강릉시 교동	5.0
4	강릉시 구정면	8.0
...

시간 변수 데이터

```
dat = pd.merge(dat, time, on='Full_NM')
dat
✓ 0.7s
```

시간데이터, 기존 데이터 merge

```
fire = pd.read_csv('./data/화재 종별 신고.재난 데이터.csv', encoding='utf-8')
fire.head()

fire['Full_NM'] = fire['SIGUNGU_NM'] + ' ' + fire['EMD_NM']

# 화재발생위치로부터의 거리
dist = pd.DataFrame(fire.groupby(fire['Full_NM'])['SPT_FRSTT_DIST'].mean())
dist = dist.reset_index()
dist
dat = pd.merge(dat, dist, on='Full_NM')
dat
✓ 0.1s
```

화재 발생위치로 부터의 거리 파생변수 추가

읍면동별 소방서부터의 거리 = 'dist'

기존 데이터, 거리 merge

3. 변수생성 및 전처리

```
#배관깊이
depth = pd.DataFrame(alarm.groupby(alarm['Full_NM'])['배관깊이'].mean())
depth = depth.reset_index()
depth

#출수압력
pressure = pd.DataFrame(alarm.groupby(alarm['Full_NM'])['출수압력'].mean())
pressure = pressure.reset_index()
pressure

#배관지름
diameter = pd.DataFrame(alarm.groupby(alarm['Full_NM'])['배관지름'].mean())
diameter = diameter.reset_index()
diameter

# 데이터간 합
dat = pd.merge(dat, depth, on='Full_NM')
dat = pd.merge(dat, pressure, on='Full_NM')
dat = pd.merge(dat, diameter, on='Full_NM')
dat

dat['EMD_NM'] = dat['Full_NM'].str[4:]
dat['SGG_NM'] = dat['Full_NM'].str[:3]
```

✓ 0.7s

배관관련 파생변수 추가

```
people = pd.read_csv('./data/강원도_읍면동별_주민등록인구.csv', encoding='cp949')
people.head()
```

✓ 0.1s

	시군	행정구역(동읍면)별	5세별	총인구수 (명)	남자인구수 (명)	여자인구수 (명)
0	춘천시	신북읍	계	7503	3763	3740
1	춘천시	동면	계	19415	9571	9844
2	춘천시	동산면	계	1451	781	670
3	춘천시	신동면	계	2604	1351	1253
4	춘천시	남면	계	1081	588	493

주민등록인구 데이터 로드

(데이터관련 문제가 있어 엑셀로 시군항목 추가)

```
dat['EMD_NM'] = dat['Full_NM'].str[4:]
dat['SGG_NM'] = dat['Full_NM'].str[:3]
```

✓ 0.6s

```
dat.head()
```

✓ 0.1s

	Full_NM	용수시설개수	화재발생횟수	time	TIME_UNIT_WS	TIME_UNIT_WD	SPT_FRSTT_DIST	배관깊이	출수압력	배관지름	EMD_NM	SGG_NM
0	홍천군 홍천읍	183	63	5.0	1.444444	152.857143	4.380952	0.997268	4.621311	100.0	홍천읍	홍천군
1	횡성군 횡성읍	141	62	5.0	2.290323	180.645161	4.800000	1.007092	5.184397	65.0	횡성읍	횡성군
2	영월군 영월읍	137	54	5.0	1.333333	143.518519	4.000000	0.936715	4.670803	100.0	영월읍	영월군
3	원주시 지정면	127	60	10.0	1.583333	177.333333	21.612903	0.675591	5.541732	100.0	지정면	원주시
4	태백시 황지동	112	49	4.0	1.387755	115.714286	2.219512	0.716161	4.650000	65.0	황지동	태백시

읍면동, 시군구 파생변수 추가

```
del people['5세별']
del people['남자인구수 (명)']
del people['여자인구수 (명)']
```

✓ 0.6s

불필요 열 제거

3. 변수생성 및 전처리

```
people.rename(columns={'행정구역(동읍면)별': 'EMD_NM'}, inplace=True)
people.rename(columns={'시군': 'SGG_NM'}, inplace=True)

people['Full_NM'] = people['SGG_NM'] + ' ' + people['EMD_NM']
people.head()
```

✓ 0.1s

	SGG_NM	EMD_NM	5세별	총인구수 (명)	남자인구수 (명)	여자인구수 (명)	Full_NM
0	춘천시	신북읍	계	7503	3763	3740	춘천시 신북읍
1	춘천시	동면	계	19415	9571	9844	춘천시 동면
2	춘천시	동산면	계	1451	781	670	춘천시 동산면
3	춘천시	신동면	계	2604	1351	1253	춘천시 신동면
4	춘천시	남면	계	1081	588	493	춘천시 남면

EMD_NM, SGG_NM 항목 추가

EMD_NM + SGG_NM = Full_NM

```
dat = pd.merge(dat, people, on='Full_NM')
dat.head()
```

✓ 0.8s

	Full_NM	용수시설개수	화재발생횟수	time	TIME_UNIT_WS	TIME_UNIT_WD	SPT_FRSTT_DIST	배관깊이	출수압력	배관지름	EMD_NM	SGG_NM	총인구수 (명)
0	홍천군 홍천읍	183	63	5.0	1.444444	152.857143	4.380952	0.997268	4.621311	100.0	홍천읍	홍천군	33754
1	횡성군 횡성읍	141	62	5.0	2.290323	180.645161	4.800000	1.007092	5.184397	65.0	횡성읍	횡성군	20976
2	영월군 영월읍	137	54	5.0	1.333333	143.518519	4.000000	0.936715	4.670803	100.0	영월읍	영월군	20883
3	원주시 지정면	127	60	10.0	1.583333	177.333333	21.612903	0.675591	5.541732	100.0	지정면	원주시	29115
4	태백시 황지동	112	49	4.0	1.387755	115.714286	2.219512	0.716161	4.650000	65.0	황지동	태백시	5970

Dat, people merge

```
# 전처리된 데이터를 tidy_data로 저장
dat.to_csv('tidy_data.csv', encoding='cp949', index=False)
```

✓ 0.6s

02데이터

4. 데이터 스케일링

```
# 전처리된 데이터를 tidy_data로 저장
dat.to_csv('tidy_data.csv', encoding='cp949', index=False)
```

✓ 0.6s

```
dat = pd.read_csv('tidy_data.csv', encoding='cp949')
dat.head()
```

✓ 0.7s

```
#데이터 스케일링을 위해 수치형, 범주형 데이터 분리
cat_dat = dat[['Full_NM', 'EMD_NM', 'SGG_NM']]
cat_dat.head()
```

✓ 0.4s

	Full_NM	EMD_NM	SGG_NM
0	홍천군 홍천읍	홍천읍	홍천군
1	횡성군 횡성읍	횡성읍	횡성군
2	영월군 영월읍	영월읍	영월군
3	원주시 지정면	지정면	원주시
4	태백시 황지동	황지동	태백시

```
#데이터 스케일링을 위해 수치형, 범주형 데이터 분리
num_dat = dat.drop(columns=['Full_NM', 'EMD_NM', 'SGG_NM'])
num_dat.head()
```

✓ 0.2s

	용수시설개수	화재발생횟수	time	TIME_UNIT_WS	TIME_UNIT_WD	SPT_FRSTT_DIST	배관깊이	출수압력	배관지름	총인구수 (명)
0	183	63	5.0	1.444444	152.857143	4.380952	0.997268	4.621311	100.0	33754
1	141	62	5.0	2.290323	180.645161	4.800000	1.007092	5.184397	65.0	20976
2	137	54	5.0	1.333333	143.518519	4.000000	0.936715	4.670803	100.0	20883
3	127	60	10.0	1.583333	177.333333	21.612903	0.675591	5.541732	100.0	29115
4	112	49	4.0	1.387755	115.714286	2.219512	0.716161	4.650000	65.0	5970

```
# 두개의 열을 새로운 계산을 위하여 따로 처리
data_count = dat[['용수시설개수', '화재발생횟수']]
data_count.head()
```

✓ 0.3s

	용수시설개수	화재발생횟수
0	183	63
1	141	62
2	137	54
3	127	60
4	112	49

‘화재예방지수’ 파생변수 생성 위해 화재대비 용수시설개수 필요

수치형, 범주형 분할 후 스케일링

4. 데이터 스케일링

```
# MinMaxScaler
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
scaler.fit(num_dat)
df_norm = scaler.transform(num_dat)
df_norm = pd.DataFrame(df_norm, columns=num_dat.columns)
df_norm.head()
```

✓ 0.6s

	용수시설개수	화재발생횟수	time	TIME_UNIT_WS	TIME_UNIT_WD	SPT_FRSTT_DIST	배관깊이	출수압력	배관지름	총인구수 (명)
0	1.000000	1.000000	0.142857	0.140351	0.394892	0.043193	0.138530	0.472174	1.0	0.695115
1	0.769231	0.983871	0.142857	0.407470	0.535015	0.049475	0.140482	0.619892	0.0	0.426308
2	0.747253	0.854839	0.142857	0.105263	0.347801	0.037481	0.126500	0.485158	1.0	0.424352
3	0.692308	0.951613	0.500000	0.184211	0.518315	0.301543	0.074621	0.713634	1.0	0.597526
4	0.609890	0.774194	0.071429	0.122449	0.207597	0.010787	0.082681	0.479700	0.0	0.110632

MinMaxScaler 이용 후 점수화

```
# 용수시설개수와 화재발생횟수의 정규화 마지막 행 제거
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
scaler.fit(data_count)
df_count = scaler.transform(data_count)
df_count = pd.DataFrame(df_count, columns=data_count.columns)
df_count = df_count.drop(143, axis=0)
df_count.head()
```

✓ 0.1s

	용수시설개수	화재발생횟수
0	1.000000	1.000000
1	0.770492	0.984127
2	0.748634	0.857143
3	0.693989	0.952381
4	0.612022	0.777778

마지막행(임의의 값)제거

```
df_norm['용수시설개수'] = df_count['용수시설개수']  
df_norm['화재발생횟수'] = df_count['화재발생횟수']
```

✓ 0.1s

- # 시간과 거리의 개념은 반대되는 개념이므로 -1 후에 음수로 변환
인구또한 적을수록 화재위험도가 낮으므로 -1 후에 음수로 변환

```
df['time'] = df['time'] - 1  
df['SPT_FRSTT_DIST'] = df['SPT_FRSTT_DIST'] - 1  
df['총인구수 (명)'] = df['총인구수 (명)'] - 1
```

```
df['time'] = -df['time']  
df['SPT_FRSTT_DIST'] = -df['SPT_FRSTT_DIST']  
df['총인구수 (명)'] = -df['총인구수 (명)']
```

✓ 0.5s

시간, 거리, 인구 값처리

02 데이터

4. 데이터 스케일링

```
df['화재예방지수'] = (df['용수시설개수'] / df['화재발생횟수'])

df.describe()
```

✓ 0.8s Python

	용수시설개수	화재발생횟수	time	TIME_UNIT_WS	TIME_UNIT_WD	SPT_FRSTT_DIST	배관깊이	출수압력	배관지름	총인구수 (명)	화재예방지수
count	143.000000	143.000000	1.430000e+02	143.000000	143.000000	143.000000	143.000000	143.000000	143.000000	143.000000	143.000000
mean	0.215102	0.355977	6.743257e-01	0.270859	0.509030	0.771018	0.162064	0.389376	0.806404	0.859308	0.675261
std	0.170640	0.222545	2.130029e-01	0.203772	0.188938	0.190974	0.109886	0.202940	0.364155	0.181084	0.501058
min	0.005464	0.015873	1.110223e-16	0.000000	0.000000	-0.000000	0.000000	0.000000	0.000000	-0.000000	0.031297
25%	0.101093	0.190476	5.714286e-01	0.121953	0.395252	0.671869	0.120112	0.240961	0.922908	0.839932	0.398770
50%	0.163934	0.301587	7.142857e-01	0.231579	0.506549	0.809352	0.140315	0.359706	1.000000	0.931462	0.602459
75%	0.278689	0.507937	8.571429e-01	0.363597	0.614604	0.929350	0.178808	0.488918	1.000000	0.961871	0.817623
max	1.000000	1.000000	1.000000e+00	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	4.810672

```
df['화재예방지수'] = (df['화재예방지수']) + (df['time']) + (df['SPT_FRSTT_DIST'])

df.describe()
```

✓ 0.9s Python

	용수시설개수	화재발생횟수	time	TIME_UNIT_WS	TIME_UNIT_WD	SPT_FRSTT_DIST	배관깊이	출수압력	배관지름	총인구수 (명)	화재예방지수
count	143.000000	143.000000	1.430000e+02	143.000000	143.000000	143.000000	143.000000	143.000000	143.000000	143.000000	143.000000
mean	0.215102	0.355977	6.743257e-01	0.270859	0.509030	0.771018	0.162064	0.389376	0.806404	0.859308	2.120604
std	0.170640	0.222545	2.130029e-01	0.203772	0.188938	0.190974	0.109886	0.202940	0.364155	0.181084	0.598446
min	0.005464	0.015873	1.110223e-16	0.000000	0.000000	-0.000000	0.000000	0.000000	0.000000	-0.000000	0.766888
25%	0.101093	0.190476	5.714286e-01	0.121953	0.395252	0.671869	0.120112	0.240961	0.922908	0.839932	1.740146
50%	0.163934	0.301587	7.142857e-01	0.231579	0.506549	0.809352	0.140315	0.359706	1.000000	0.931462	2.069218
75%	0.278689	0.507937	8.571429e-01	0.363597	0.614604	0.929350	0.178808	0.488918	1.000000	0.961871	2.519086
max	1.000000	1.000000	1.000000e+00	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	5.723078

‘화재예방지수’ 파생변수 생성

화재예방지수 = (용수시설개수 / 화재발생횟수) + 시간 + 거리

```
df['용수시설지수'] = df['출수압력'] + df['배관지름']
```

✓ 0.6s

‘용수시설지수’ 파생변수 생성

용수시설지수 = 출수압력 + 배관지름

```
final_all = pd.concat([cat_dat, final_num], axis=1)
```

✓ 0.5s

```
final_all.head()
```

✓ 0.3s

	Full_NM	EMD_NM	SGG_NM	화재예방지수	용수시설지수	총인구수 (명)
0	홍천군 홍천읍	홍천읍	홍천군	2.813950	1.472174	0.304885
1	횡성군 횡성읍	횡성읍	횡성군	2.590587	0.619892	0.573692
2	영월군 영월읍	영월읍	영월군	2.693068	1.485158	0.575648
3	원주시 지정면	지정면	원주시	1.927146	1.713634	0.402474
4	태백시 황지동	황지동	태백시	2.704669	0.479700	0.889368

final_all = 화재예방지수 + 용수시설지수 + 인구 + 읍면동

4. 데이터 스케일링

```
final_num['Final Score'] = final_num['화재예방지수'] + final_num['용수시설지수'] + final_num['총인구수 (명)']  
final_num
```

	화재예방지수	용수시설지수	총인구수 (명)	Final Score
0	2.813950	1.472174	0.304885	4.591009
1	2.590587	0.619892	0.573692	3.784170
2	2.693068	1.485158	0.575648	4.753873
3	1.927146	1.713634	0.402474	4.043253
4	2.704669	0.479700	0.889368	4.073738
...

‘Final Score’ 파생변수 생성

Final Score = 모든변수 합

```
final_data = pd.concat([cat_dat, final_num], axis = 1)  
  
del final_data['화재예방지수']  
del final_data['용수시설지수']  
del final_data['총인구수 (명)']
```

✓ 0.5s

```
final_data.sort_values(by='Final Score').head(5)
```

✓ 0.6s

	Full_NM	EMD_NM	SGG_NM	Final Score
142	삼척시 하장면	하장면	삼척시	2.305948
40	횡성군 둔내면	둔내면	횡성군	2.741306
93	정선군 남면	남면	정선군	2.877478
65	영월군 주천면	주천면	영월군	2.917247
138	강릉시 왕산면	왕산면	강릉시	3.112850

Final score 제외 수치형변수 제거

final score를 내림차순으로 정렬하여 최종적 입지선정

삼척군 하장면, 횡성군 둔내면, 정선군 남면, 영월군 주천면, 강릉시 왕산면


```
df_l = df[(df['Full_NM'] == '삼척시 하장면') | (df['Full_NM'] == '횡성군 둔내면') | (df['Full_NM'] == '정선군 남면')
|         | (df['Full_NM'] == '영월군 주천면') | (df['Full_NM'] == '강릉시 왕산면')]

qgis = df_l[['Full_NM', 'SIGUNGU_NM', 'EMD_NM', 'LOC_INFO_X', 'LOC_INFO_Y']]
qgis.dropna(axis=0)
# qgis.to_csv('qgisdata.csv', encoding='cp949')
```

✓ 0.1s Python

	Full_NM	SIGUNGU_NM	EMD_NM	LOC_INFO_X	LOC_INFO_Y
5	삼척시 하장면	삼척시	하장면	128.9242	37.3575
127	횡성군 둔내면	횡성군	둔내면	128.2436	37.5006
138	횡성군 둔내면	횡성군	둔내면	128.2086	37.5125
161	횡성군 둔내면	횡성군	둔내면	128.1865	37.4764
226	영월군 주천면	영월군	주천면	128.2699	37.2253
231	삼척시 하장면	삼척시	하장면	128.8698	37.3925

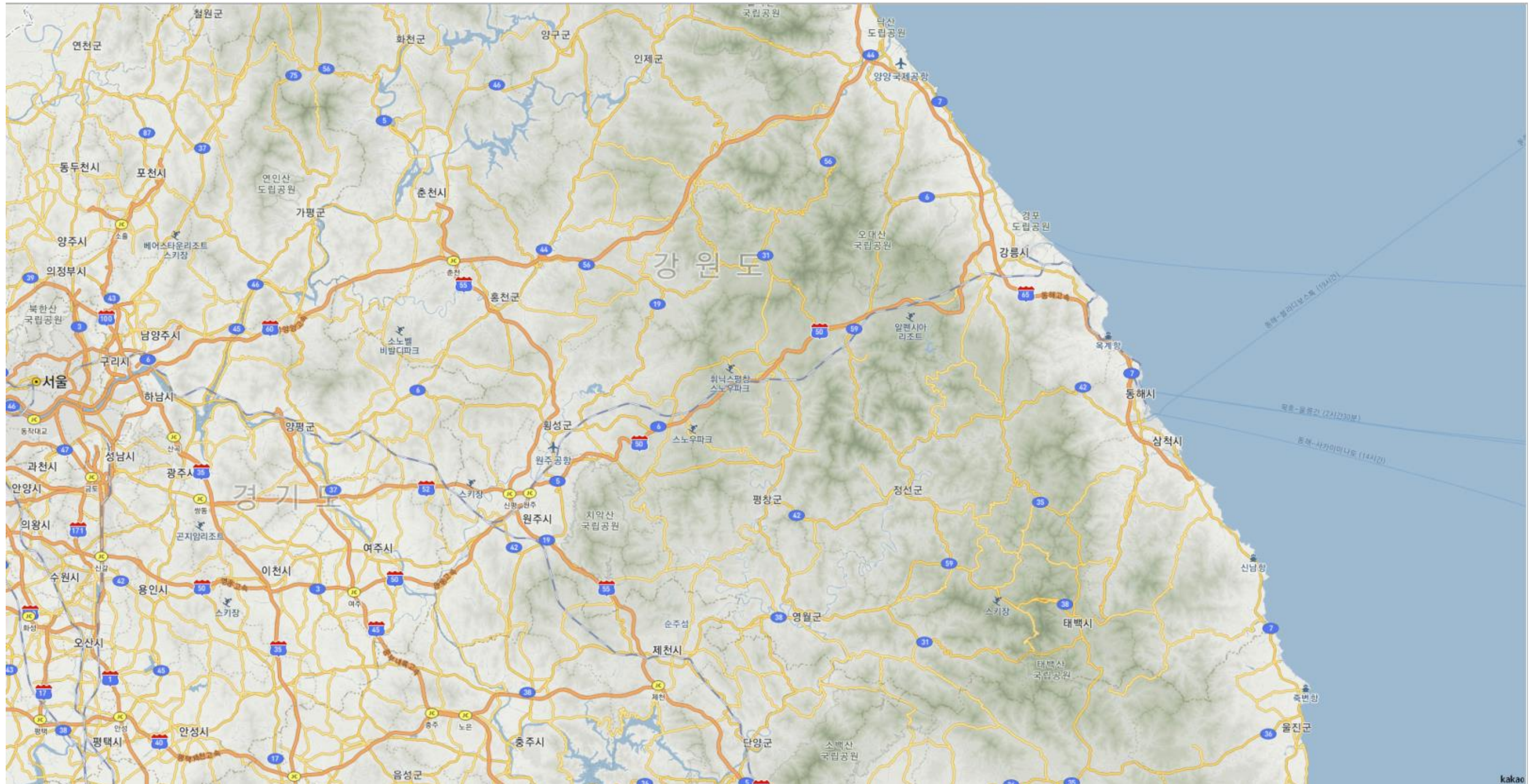
```
alarm1 = alarm[(alarm['Full_NM'] == '삼척시 하장면') | (alarm['Full_NM'] == '횡성군 둔내면') | (alarm['Full_NM'] == '정선군 남면')
|              | (alarm['Full_NM'] == '영월군 주천면') | (alarm['Full_NM'] == '강릉시 왕산면')]

qgis = alarm1[['Full_NM', 'EMD_NM', '위도', '경도']]
qgis.dropna(axis=0)
# qgis.to_csv('qgisdata_alarm.csv', encoding='cp949')
```

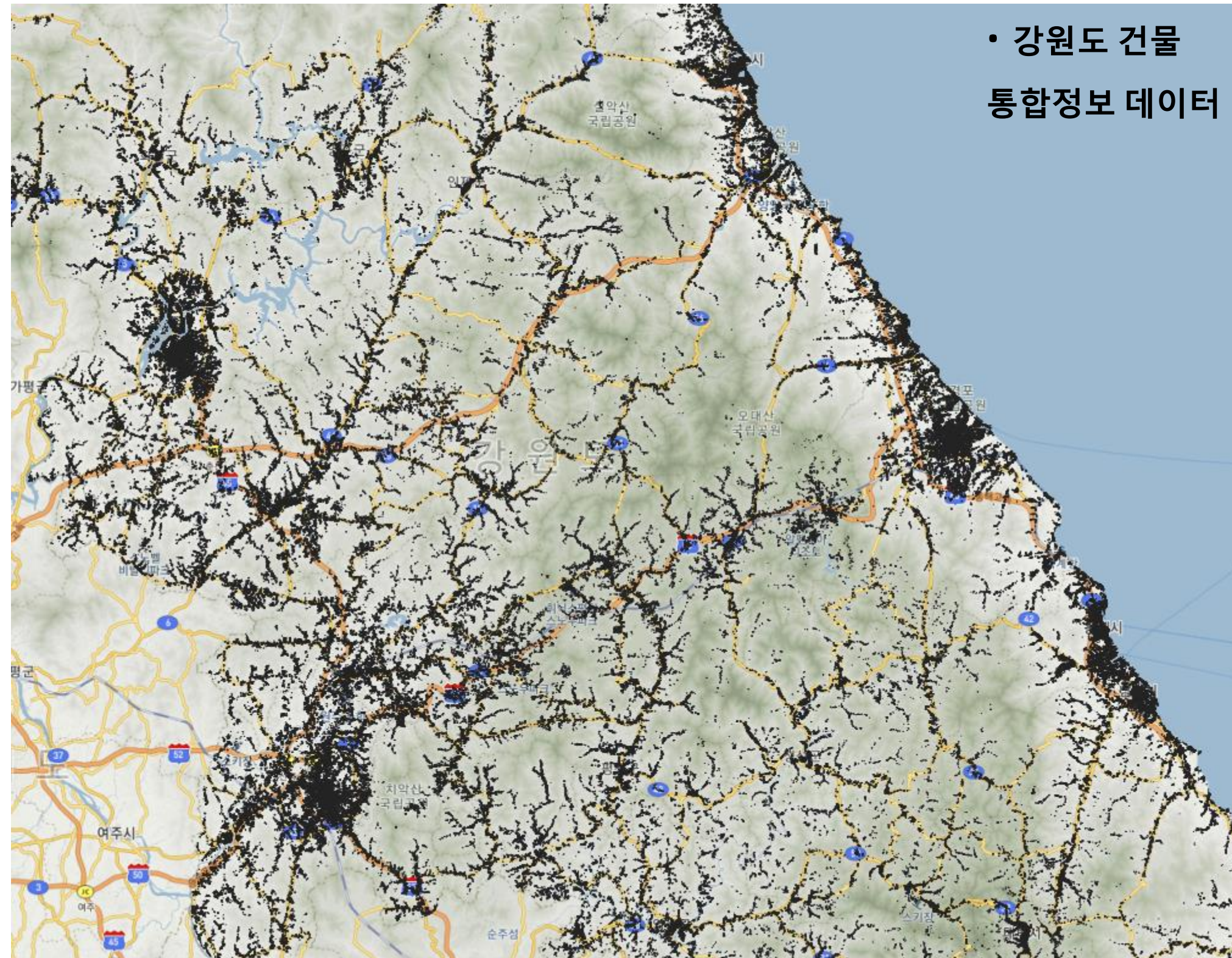
✓ 0.7s Python

	Full_NM	EMD_NM	위도	경도
1433	횡성군 둔내면	둔내면	37.519203	128.201199
1434	횡성군 둔내면	둔내면	37.512914	128.214327
1435	횡성군 둔내면	둔내면	37.514725	128.216582
1436	횡성군 둔내면	둔내면	37.513427	128.217923
1437	횡성군 둔내면	둔내면	37.512626	128.216204
...

입지선정 top5 용수시설좌표, 화재발생위치 좌표 생성

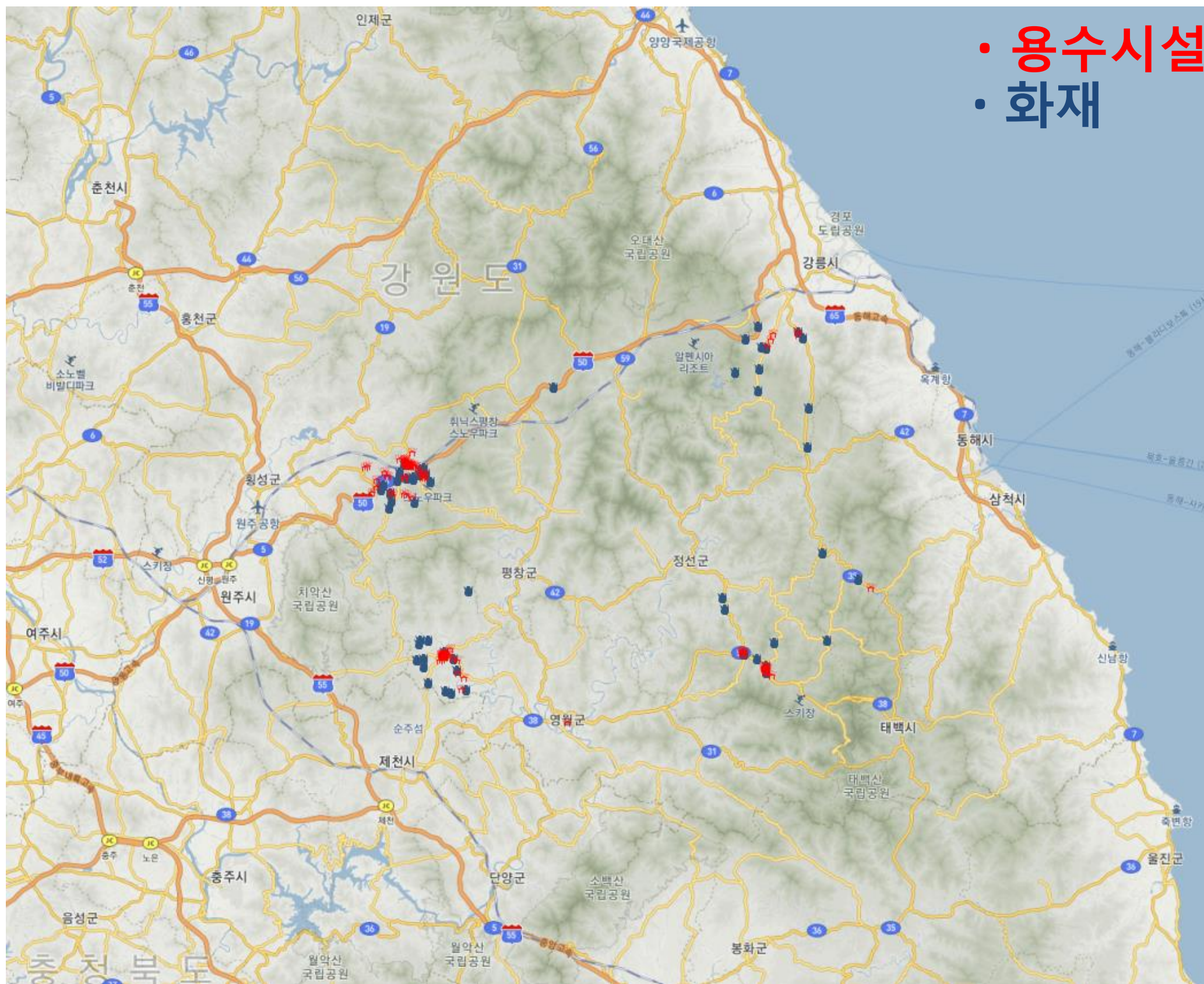


QGIS 에 Kakao Map 로드



강원도 건물 통합정보 데이터 로드

5. 분석



TOP 5 용수시설 및 화재

	Full_NM	EMD_NM	SGG_NM	Final Score
142	삼척시 하장면	하장면	삼척시	2.305948
40	횡성군 둔내면	둔내면	횡성군	2.741306
93	정선군 남면	남면	정선군	2.877478
65	영월군 주천면	주천면	영월군	2.917247
138	강릉시 왕산면	왕산면	강릉시	3.112850

상위 5개 읍면동

YEAR = left("USEAPR_DAY", 4)

둔내면 — 필드 계산기

☐ 선택한 객체 0 개만 업데이트

☒ 새로운 필드 생성 ☐ 기존 필드를 갱신

☐ 가상 필드 생성

산출 필드 이름: YEAR

산출 필드 유형: 정수 (정수형)

산출 필드 길이: 10 정밀도: 3

표현식 함수 편집기

left("USEAPR_DAY", 4)

객체: \$5300000000

미리보기: NULL

row_number

- 날짜와 시간
- 도형
- 래스터
- 레코드와 속.
- 맵
- 문자열
- 배열
- 변수
- 변환
- 색상
- 수학

이 레이어 상에서 정보를 편집하고 있지만 레이어가 현재 편집 모드가 아닙니다. '확인'을 클릭하면 자동적으로 편집 모드로 전환될 것입니다.

확인 취소 도움말

노후 불량 건축물 판단 기준

제4조(노후 · 불량건축물)

① 영 제2조제3항제1호에 따라 노후·불량건축물로 보는 기준은 다음 각 호와 같다.

1. 공동주택

가. 철근콘크리트·철골콘크리트·철골철근콘크리트 및 강구조인 공동주택: 별표 1에 따른 기간

<별표 1> 철근콘크리트 · 철골콘크리트 · 철골철근콘크리트 및 강구조 공동주택의 노후 · 불량건축물 기준 (제4조제1항제1호 관련)

나. 가목 이외의 공동주택: 20년

2. 공동주택 이외의 건축물

가. 철근콘크리트·철골콘크리트·철골철근콘크리트 및 강구조 건축물(「건축법 시행령」 별표 1 제1호에 따른 단독주택을 제외한다): 30년

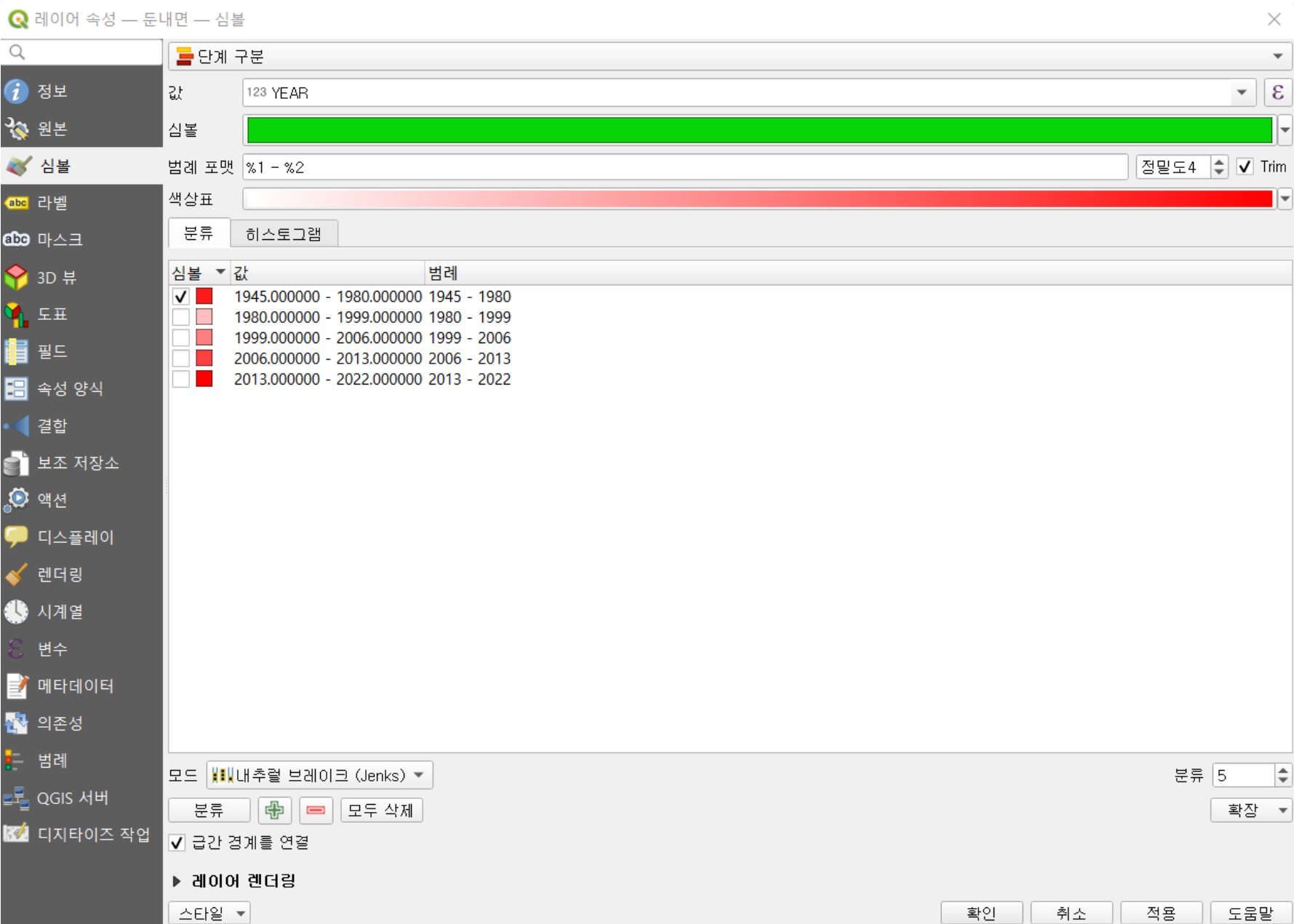
나. 가목 이외의 건축물: 20년

② 영 제2조제2항제1호에 따른 노후·불량건축물은 건축대지로서 효용을 다할 수 없는 과소필지 안의 건축물로서 2009년 8월 11일 전에 건축된 건축물을 말한다.

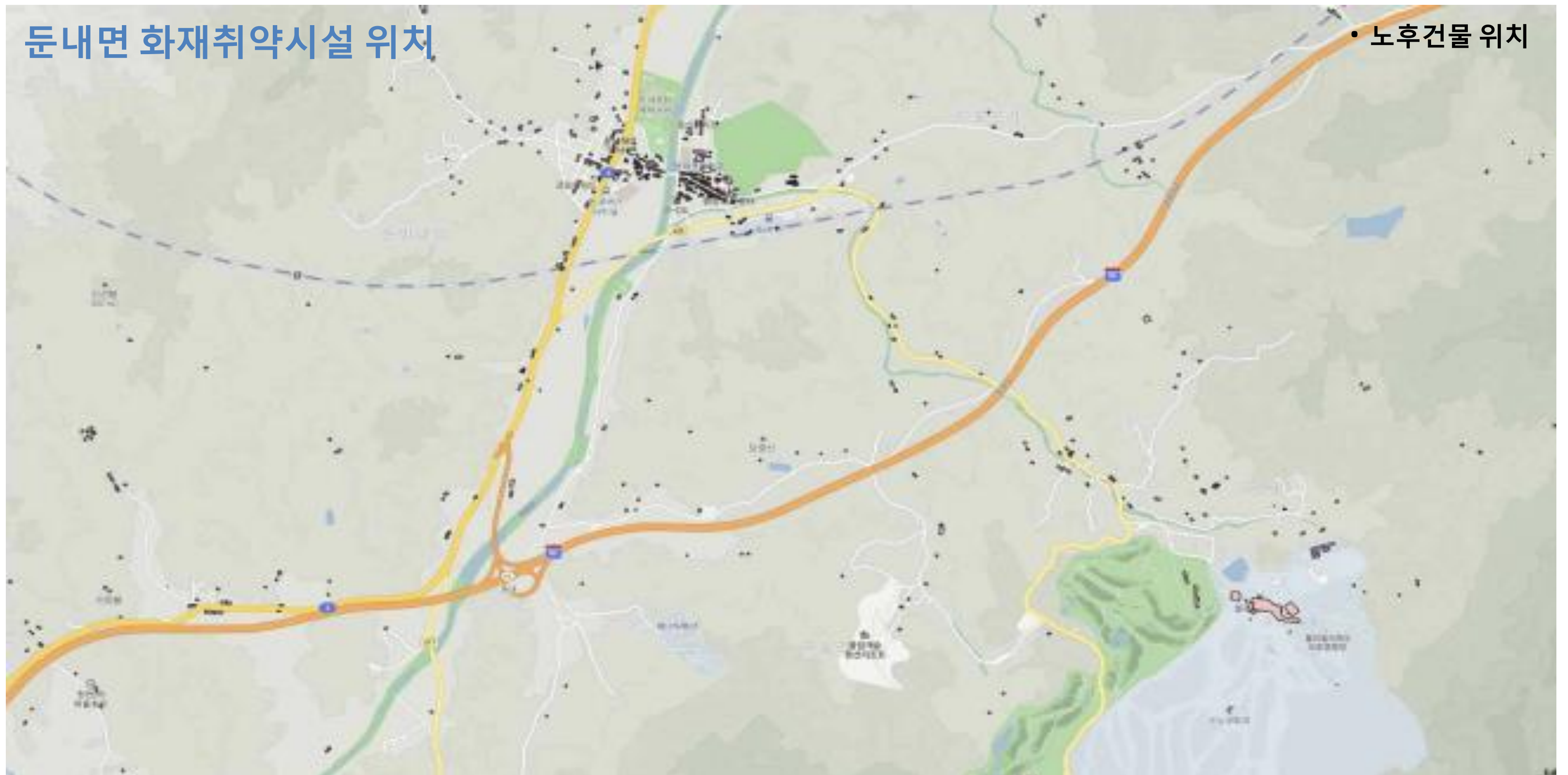
제2조(정의) 이 조례에서 사용하는 용어의 뜻은 다음과 같다.

9. "과소필지"란 토지면적이 90제곱미터 미만인 토지를 말한다.

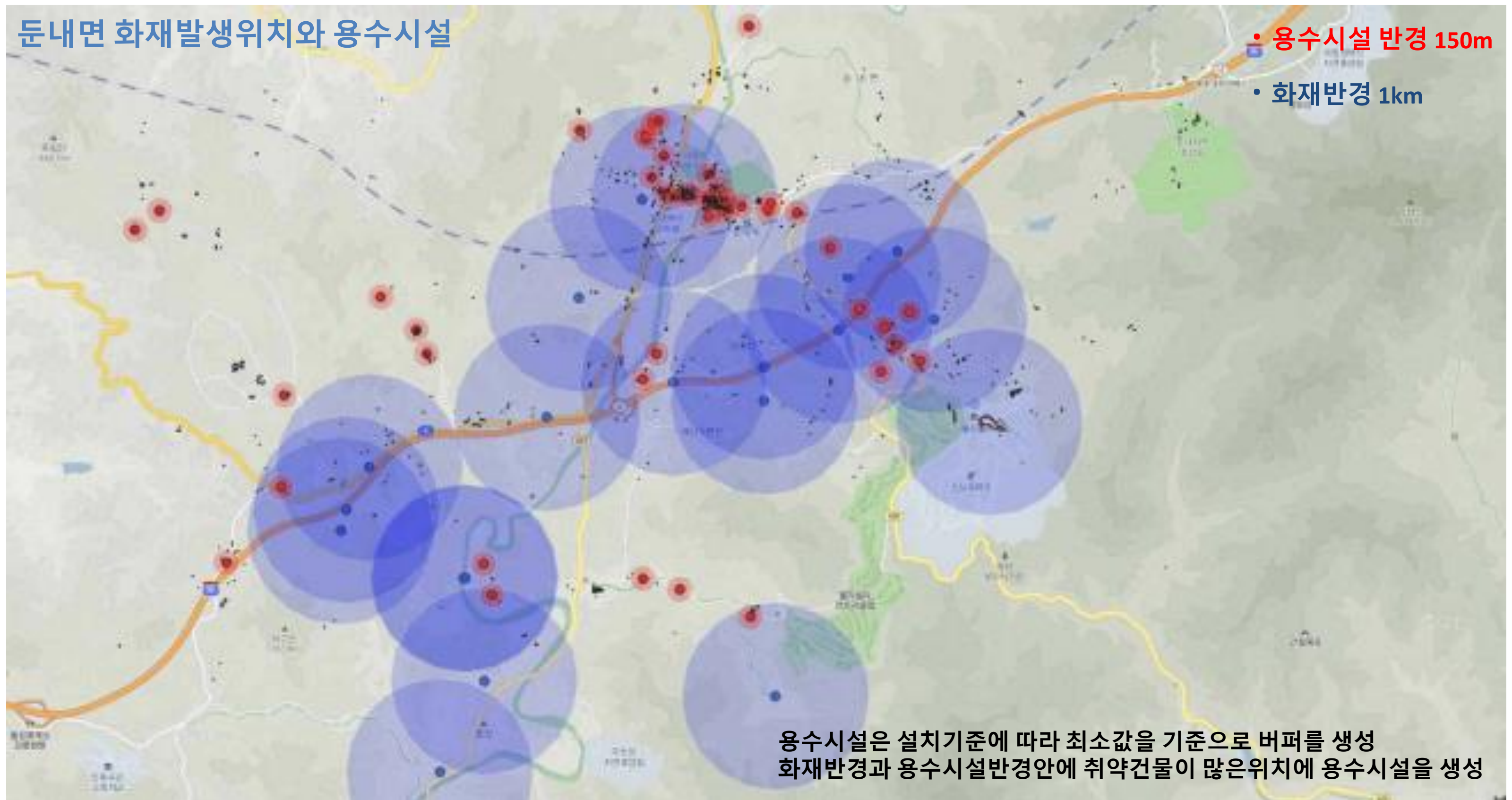
③ 미사용승인건축물의 용도별 분류 및 구조는 건축허가 내용에 따르며, 준공 연도는 재산세 및 수도요금·전기요금 등의 부과가 개시된 날이 속하는 연도로 한다.



노후 불량 건축물 판단 기준
1981 > YEAR = 노후건물



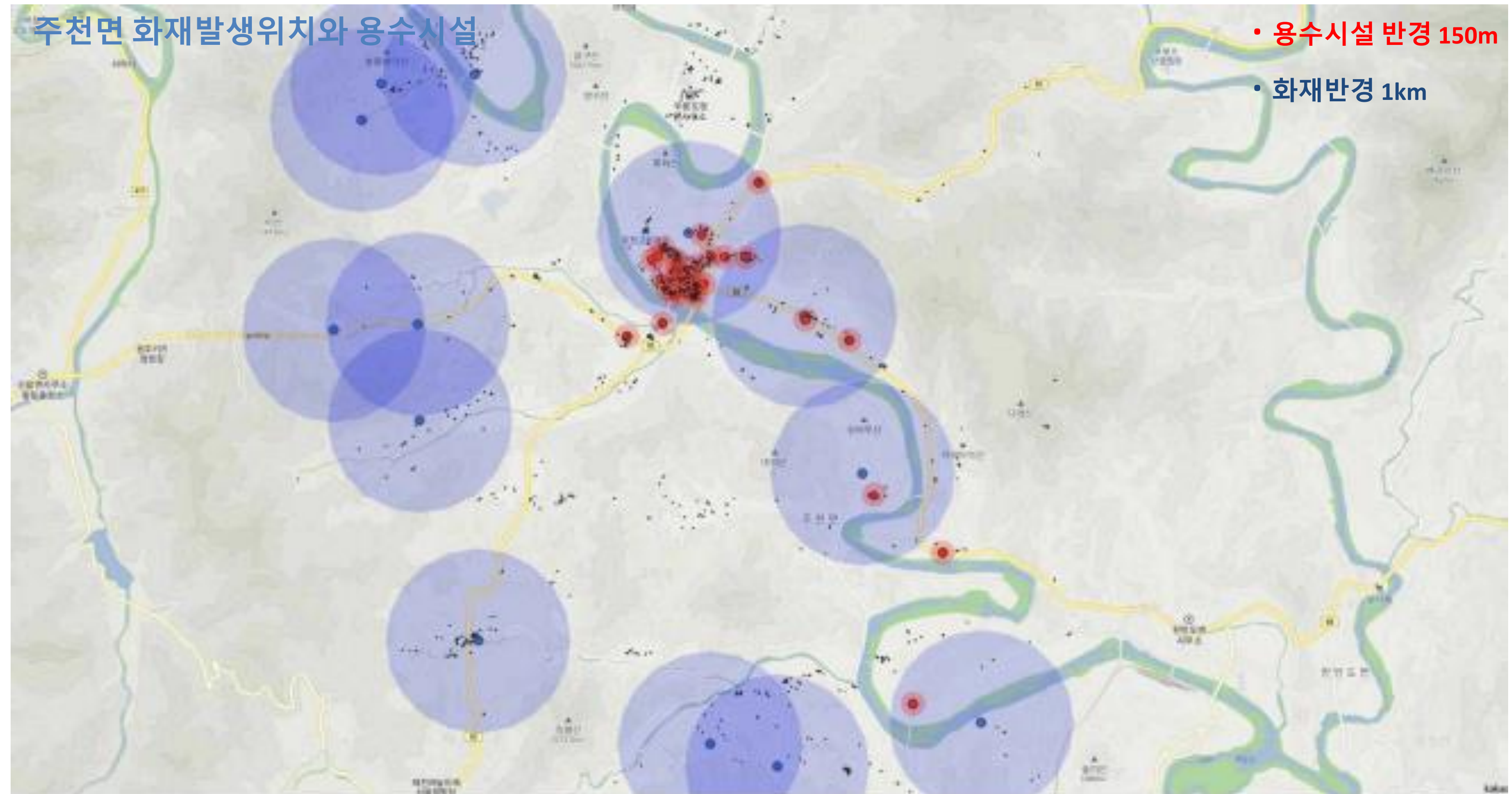
둔내면 화재발생위치와 용수시설



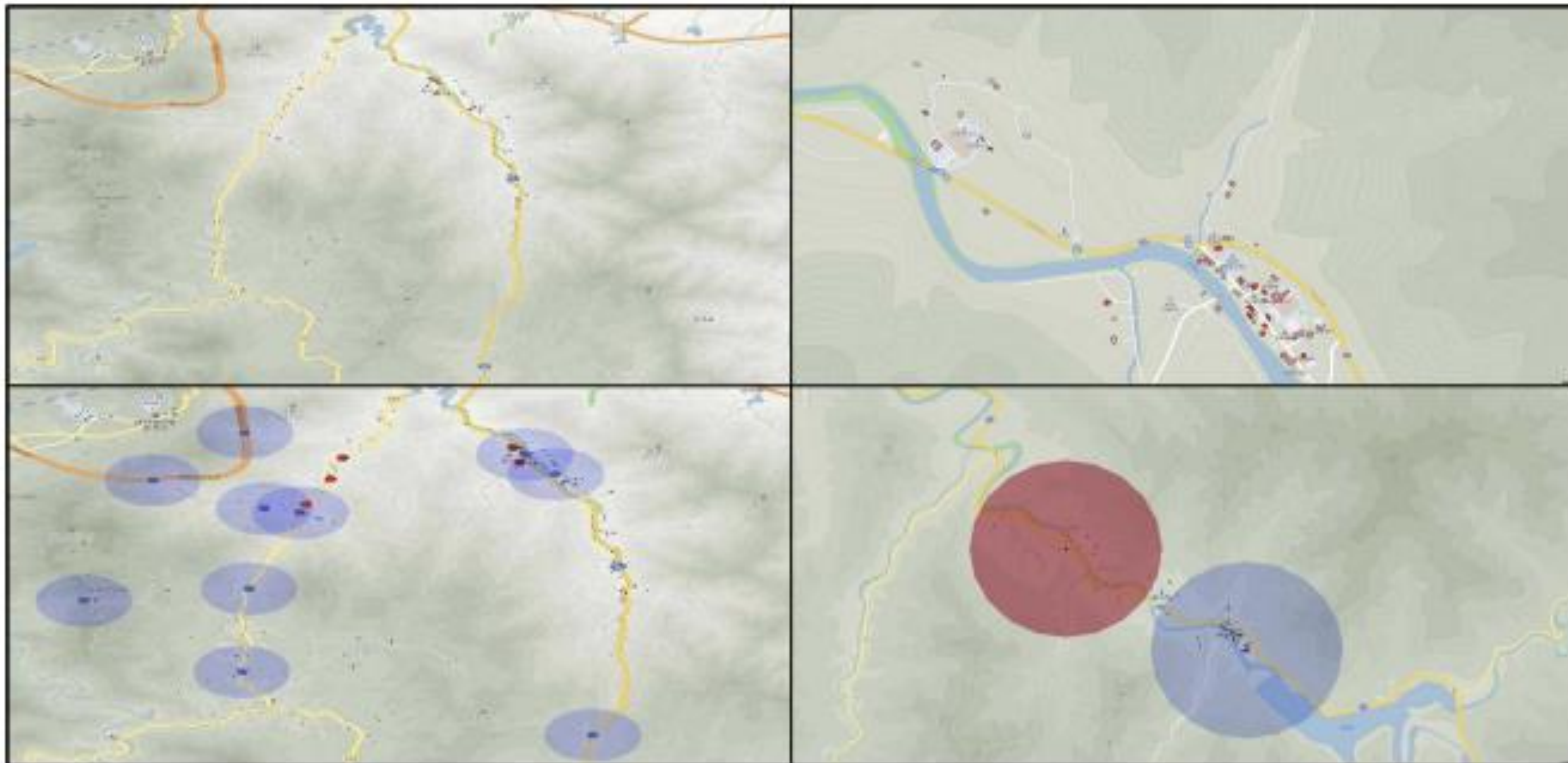
주천면 화재취약시설 위치

• 노후건물 위치





시각화 결과



03

결론

- 결과 설명
- 기대 효과
- 한계점

03 결론

결과 설명

01

변수 설정

화재/용수시설/인구/배관/시간/거리

02

파생 변수 간 합 = Final Score

수치가 낮을수록 화재 피해 ↑

03

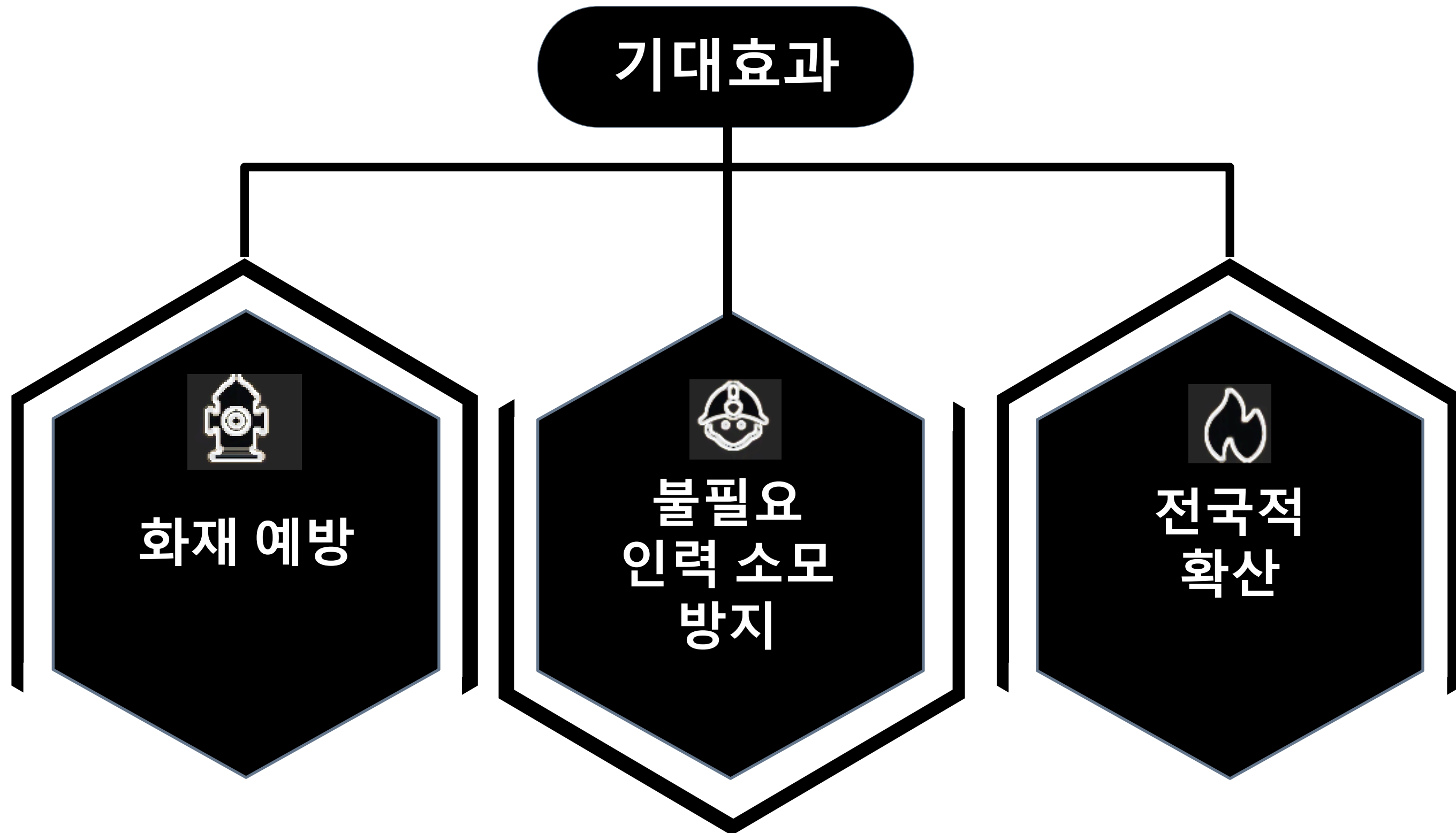
5개 지역 입지 선정

삼척군 하장면/횡성군 둔내면/정선군 남면/
영월군 주천면/강릉시 왕산면



설정된 읍면동
화재 발생 위치,
용수시설 위치 버퍼 생성

중복x, 노후 건물 多
용수시설 설치



1.

화재 데이터 부족(1개년)으로 인한 읍면동 단위의 점수 불균형

2.

데이터별로 기준이 달라, 병합 시 유실 데이터 일부 존재

3.

일부 데이터의 결측치가 있어 평균, 중앙값으로 일부 대체

4.

Final Score 당위성 부족

5.

다양한 화재취약시설 고려 x

6.

소방용수시설 생성 시 해당 위치가 사유지라 만들지 못하는 문제 발생 우려
(분석으로는 대략적인 위치만 파악 가능)

출처 및 분석 툴

소방방재신문 2022.04.20. (<https://fpn119.co.kr/176510>)

이데일리 2020.03.02

(<https://www.edaily.co.kr/news/read?newsId=02945446625699384&mediaCodeNo=257&OutLnkChk=Y>)



감사합니다
