

DS26 ML

사용자 맞춤 여행지 – 추천

머신러닝 프로젝트 평일 저녁 5조
팀원 : 김요은, 김예지, 한혜승

목차

1



개요

2



프로젝트 진행 과정

3



EDA

4



데이터 전처리

5



머신러닝 모델 구현

6



결과 분석 및 결론

1
개요

1. 프로젝트 소개

여행



많은 인터넷 정보

귀찮은 계획 짜기

나한테 딱 맞는 여행

사용자의 취향과 선호도를 바탕으로
맞춤형 여행지를 추천하는 시스템을 개발

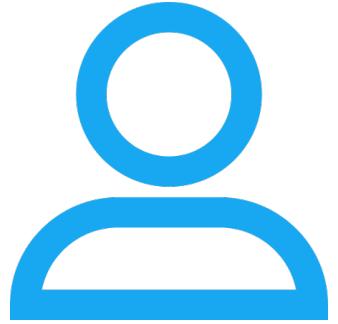
사용자 경험 개선

: 개인화된 추천을 통해 사용자 만족도 증가,
여행 계획 시간 단축

비즈니스 모델 창출 및 기술 확장

: 추천 시스템을 활용한 맞춤형 여행 패키지
제공, 음식 쇼핑 등의 분야로 확장

2. 기대 모델



사용자 정보

사용자가 원하는 여행의
방향을 알려주는 정보



추천 여행지

사용자가 만족할만한
여행지 장소 추천



여행지 명소

추천된 여행지 장소에
가장 적합한 여행 명소 추천

3. 사용 데이터

데이터 소스 : AI-Hub

<https://aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&dataSetSn=282>



<여행객 정보>

- 여행객 ID, 성별, 연령대, 선호시도, 선호시군구, 여행스타일, 거주지, 목적지

<여행 정보>

- 여행객 ID, 여행 ID, 여행목적, 여행 일자

<방문지 정보>

- 여행 ID, 방문지명, 방문일자, 주소, 좌표, 방문지 유형

2

프로젝트 진행과정

1

EDA

- 사용 데이터 확정
- 사용 데이터 분석
- 프로젝트 입출력
데이터 확정
- 전처리 데이터 확정
- 머신러닝 모델 방향성
정의

2

데이터 전처리

- 데이터 가공
(파악 용이)
- NAN 값 처리
- 데이터 병합
- 방문지 정보 주소 확인
- 주소 API 대조 및 처리

3

머신러닝

- 모델 선정
- 여러 모델 적용
- 모델 정확도 비교

4

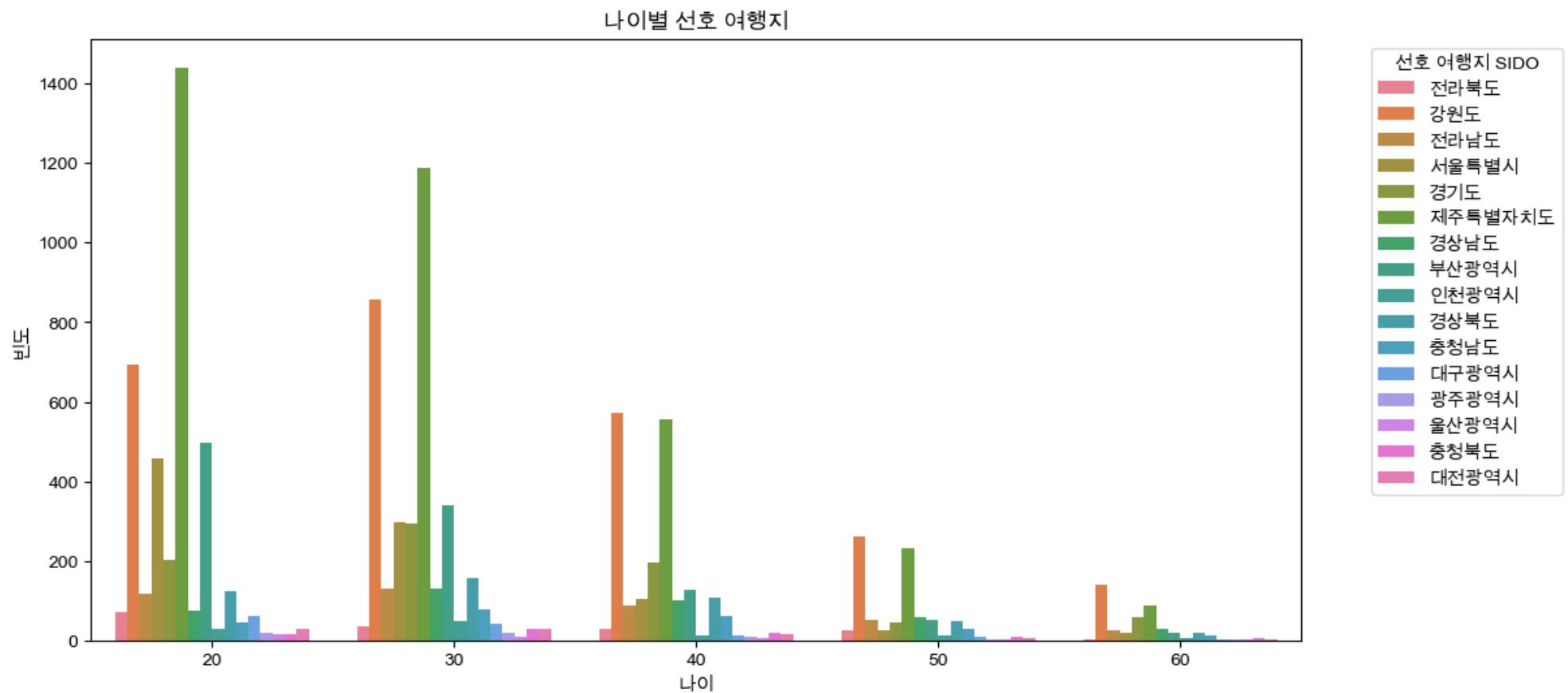
결과 분석

- 결과 출력 값 Folium
- EDA 분석
- 실제 추천명소 사이트
비교

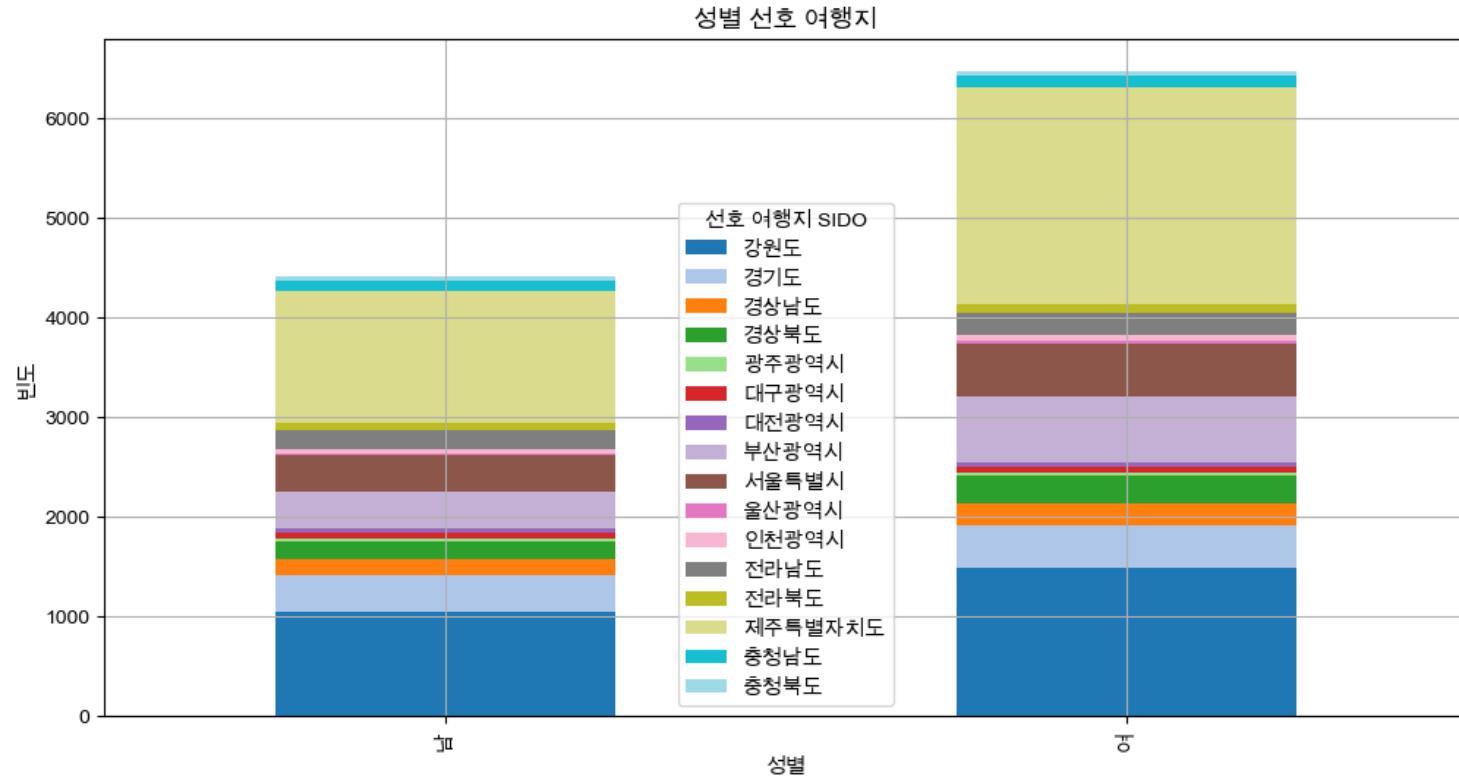
3

EDA

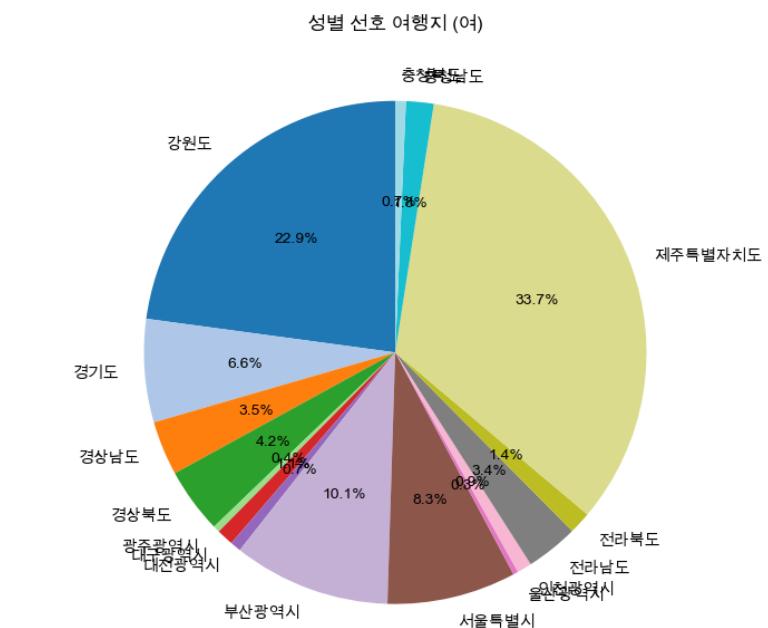
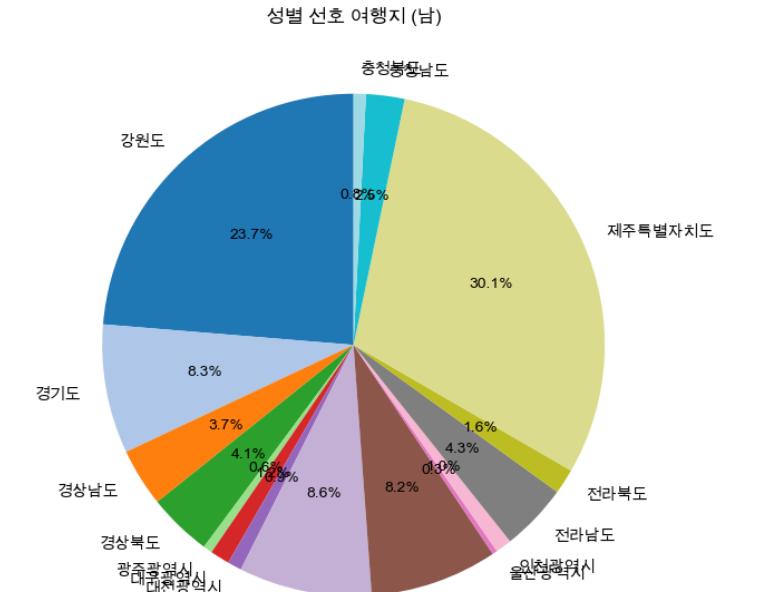
1) 나이별, 성별 선호 여행지(시도)



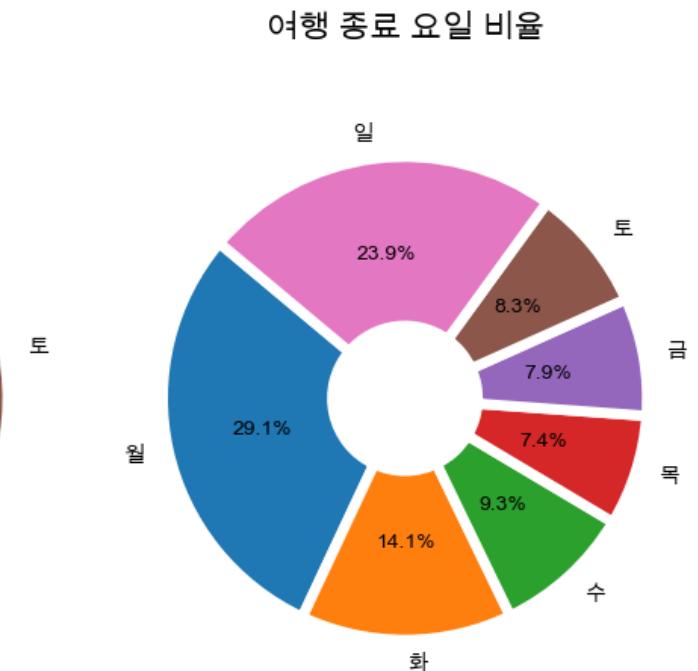
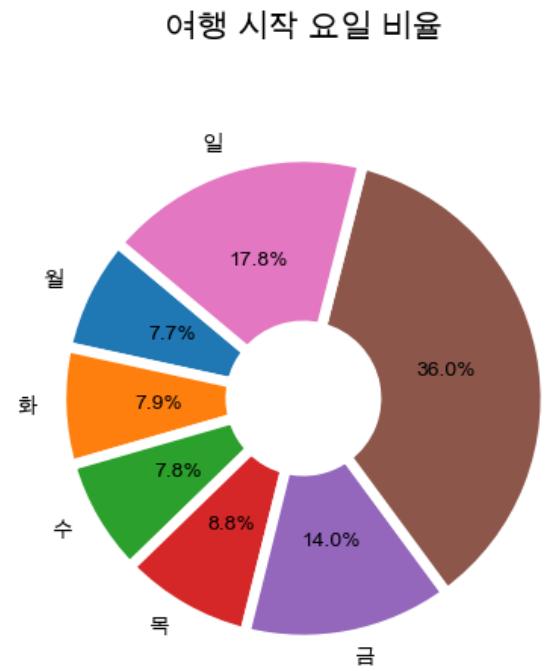
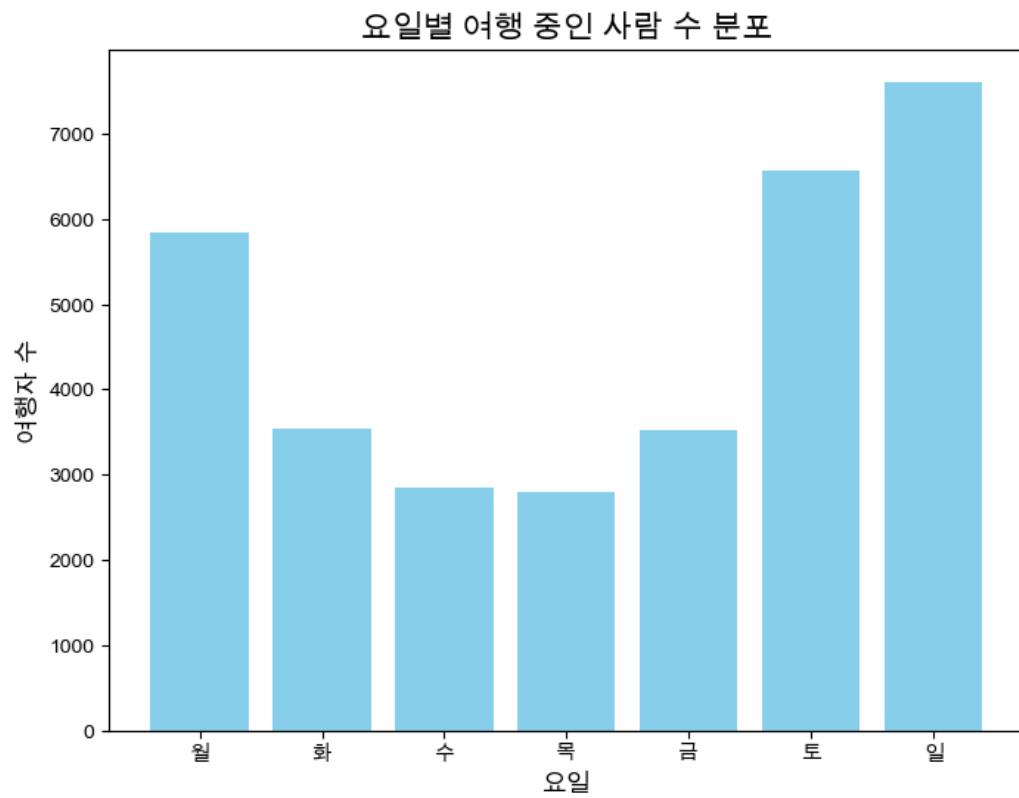
1) 나이별, 성별 선호 여행지(시도)



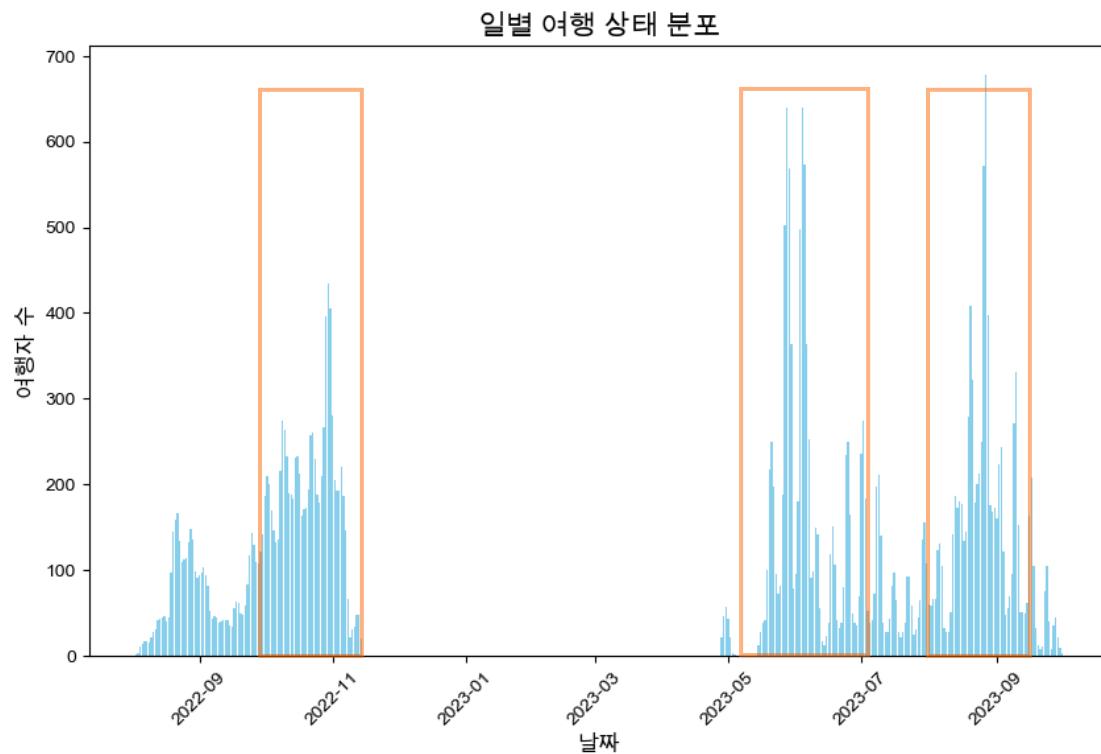
나이별, 성별 선호 여행지의 비율은 큰 차이가 없음.
나이별, 성별 선호 여행지의 응답률은 차이가 있음.



2) 요일별 여행 분포



3) 여행 상태 분포

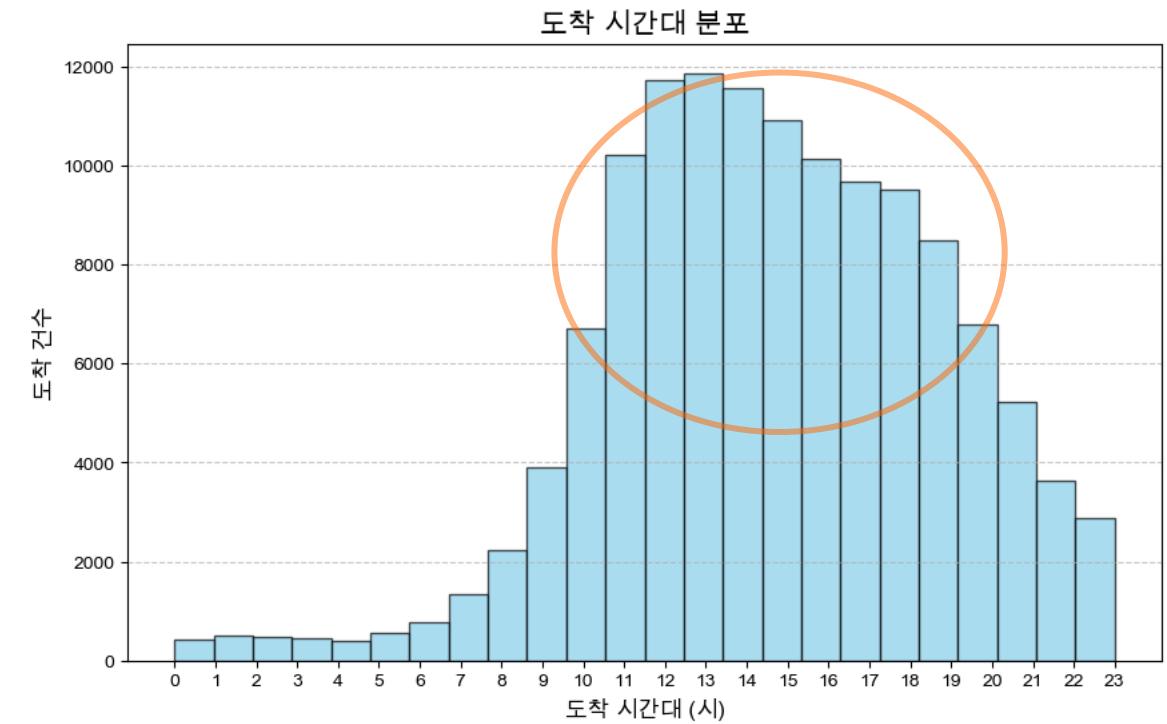
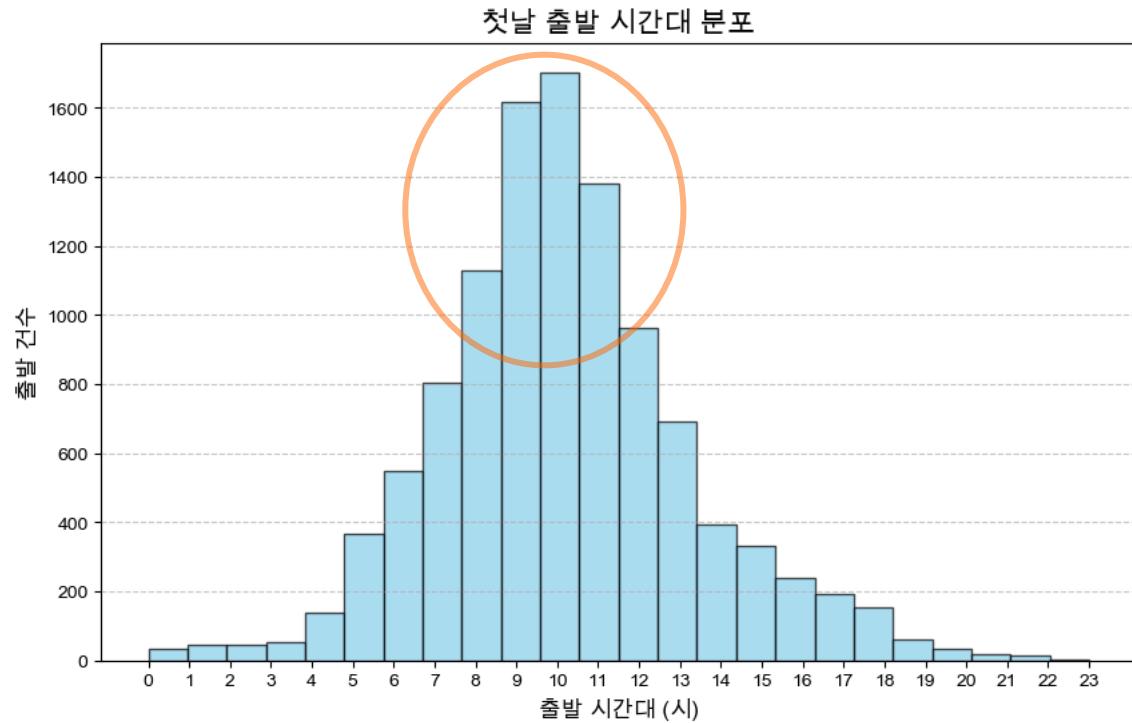


달력		양음력변환					전역일계산		
		2023.05							
일	월	화	수	목	금	토	음력	순없는날	기념일
30	1 근로자의 날	2	3	4 음 3.15 어린이날	5	6 입하			
7	8 어버이날	9	10 바다식물일 유권자의 날	11 동학농민...	12 국제 간호...	13			
14	15 식품안전... 성년의 날	16	17	18 5.18민주... 발명의 날	19	20 세계인의 날			
21	22 소만 부부의 날	23	24	25 방재의 날	26	27 부처님 오...			
28	29 대체 휴일	30	31 세계 금연... 바다의 날	1	달력		양음력변환		

달력		양음력변환					전역일계산		
		2023.06							
일	월	화	수	목	금	토	음력	순없는날	기념일
28	29	30	31	1 의병의 날	2	3 음 4.15			
4	5 환경의 날	6 현충일 망종	7	8	9	10 6.10 민주...			
11	12	13	14 세계 현혈... 노인학대 ...	15	16	17			
21	22 하지 해양조사...	23	24						

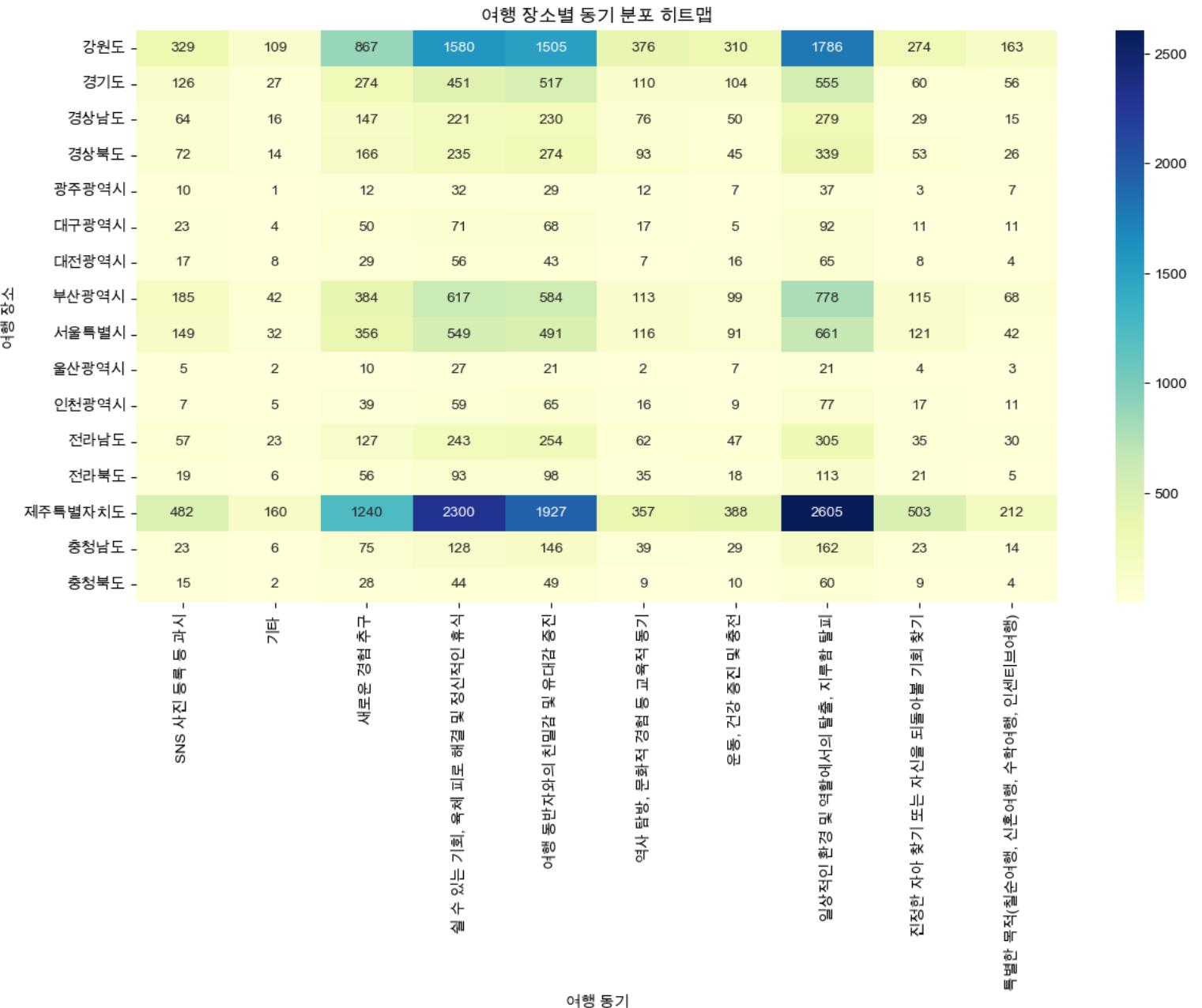
달력		양음력변환					전역일계산		
		2023.08							
일	월	화	수	목	금	토	음력	순없는날	기념일
30	31	1 음 6.15 유두절	2	3	4	5			
6	7	8 입추	9	10 말복	11	12			
13	14	15 광복절	16 음 7.1	17	18	19			
20	21	22 칠석	23 처서	24	25	26			

3) 여행 상태 분포

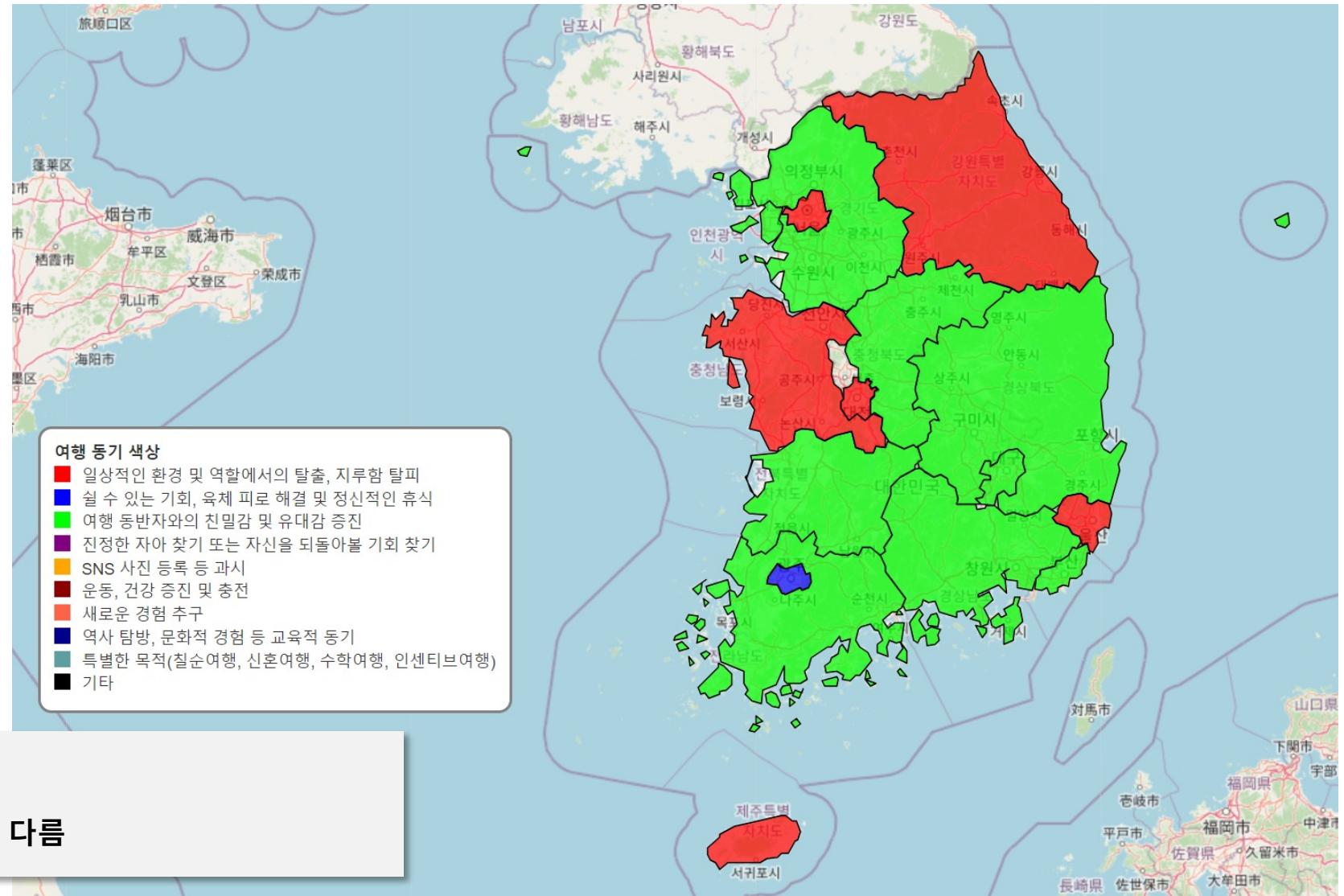


여행지 추천에 요일, 일정, 시간이 유의미한 결과를 내기 어렵다고 판단.
여행 시기와 여행지 접속 방법을 찾을 수 없음.

4) 장소별 여행 동기 분포



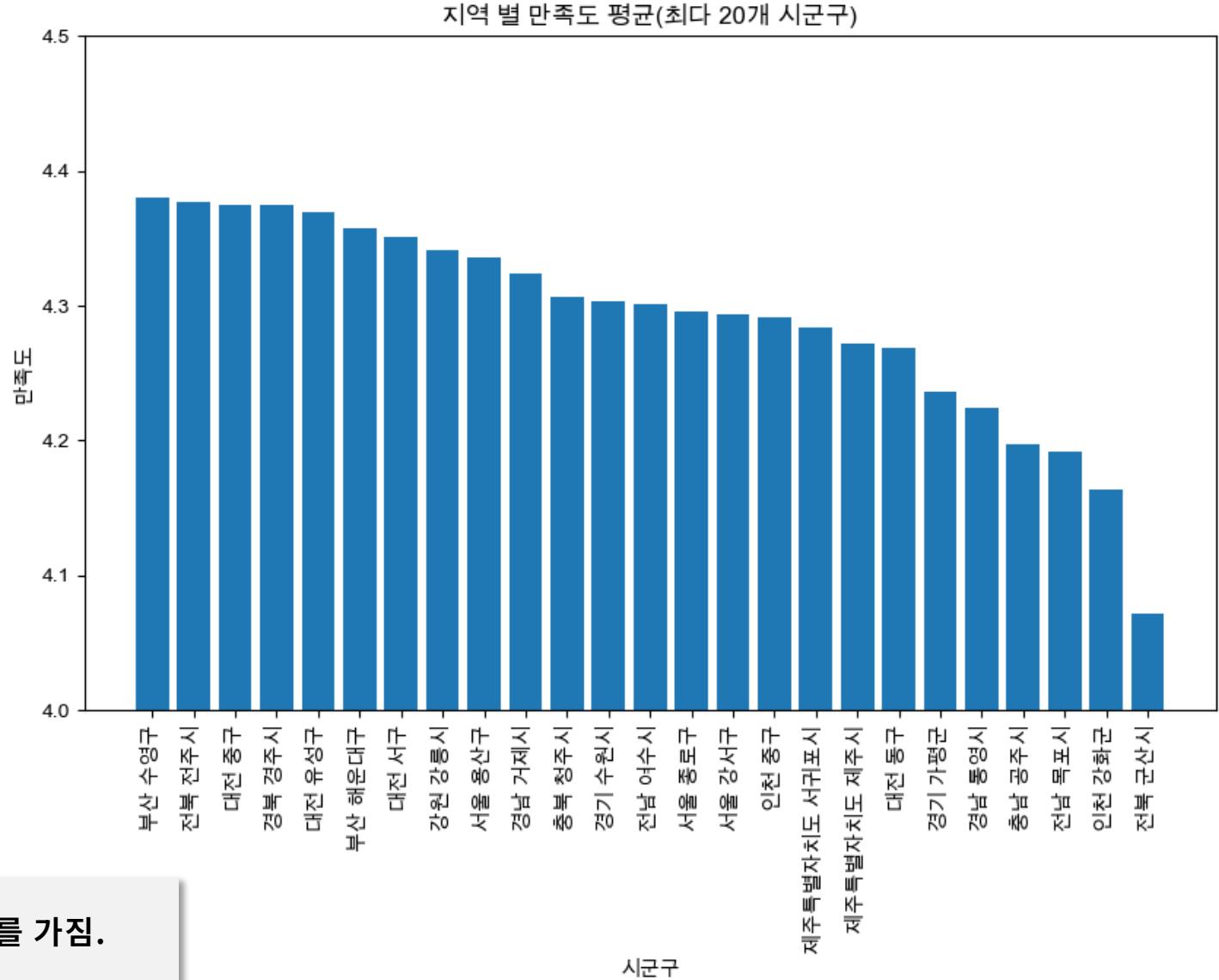
4) 장소별 여행 동기 분포



지역별로 응답된 내용 대부분 비슷

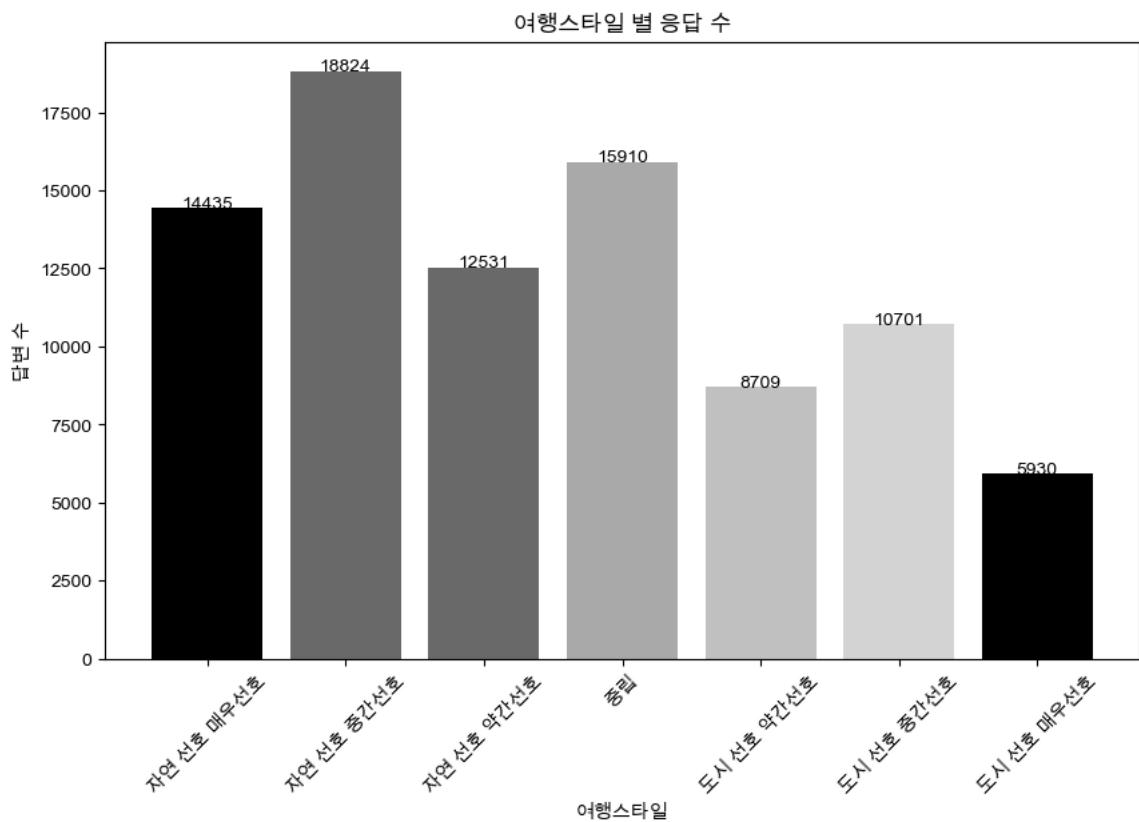
그러나 선호 지역별로 여행 동기가 약간씩 다름

5) 장소별 여행 만족도 (TOP20)

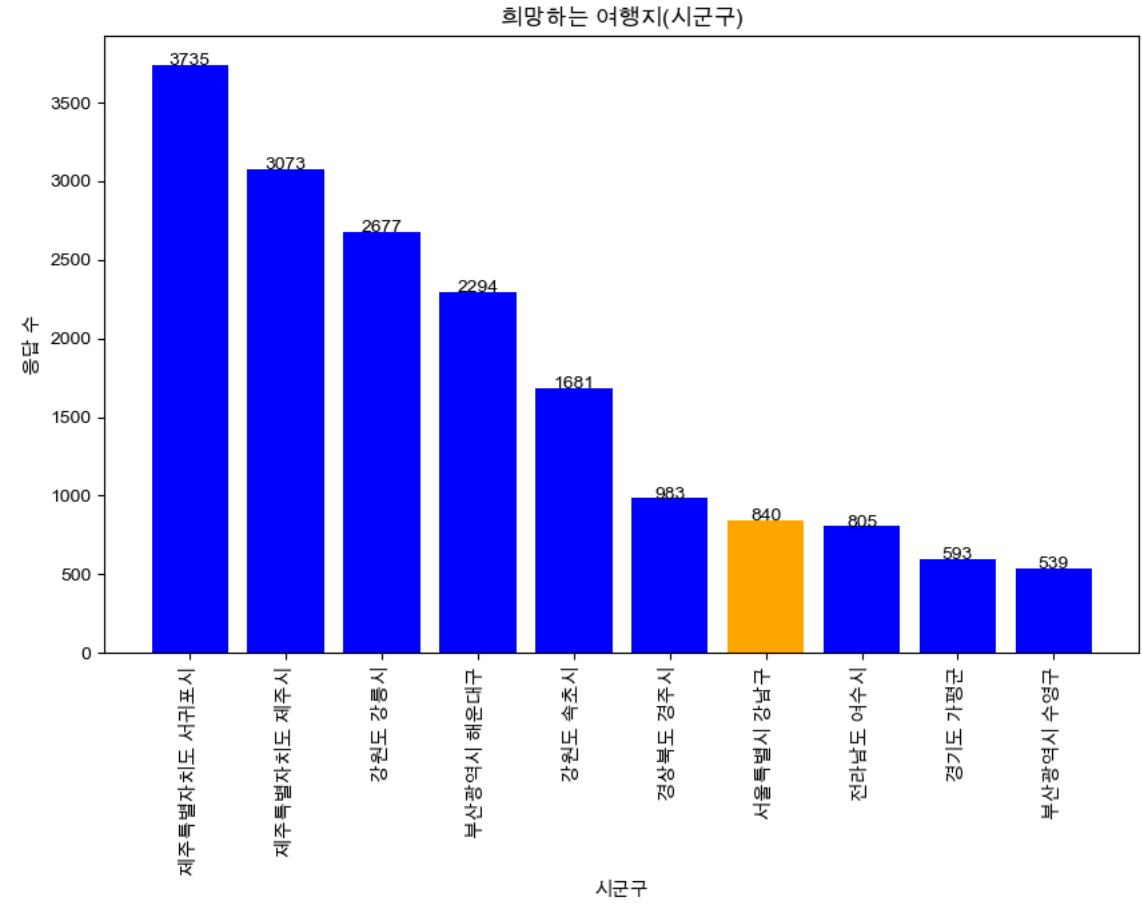


여행 만족도가 지역별로 미세한 차이를 가짐.
만족도가 4점이 넘어가는 지역 다수

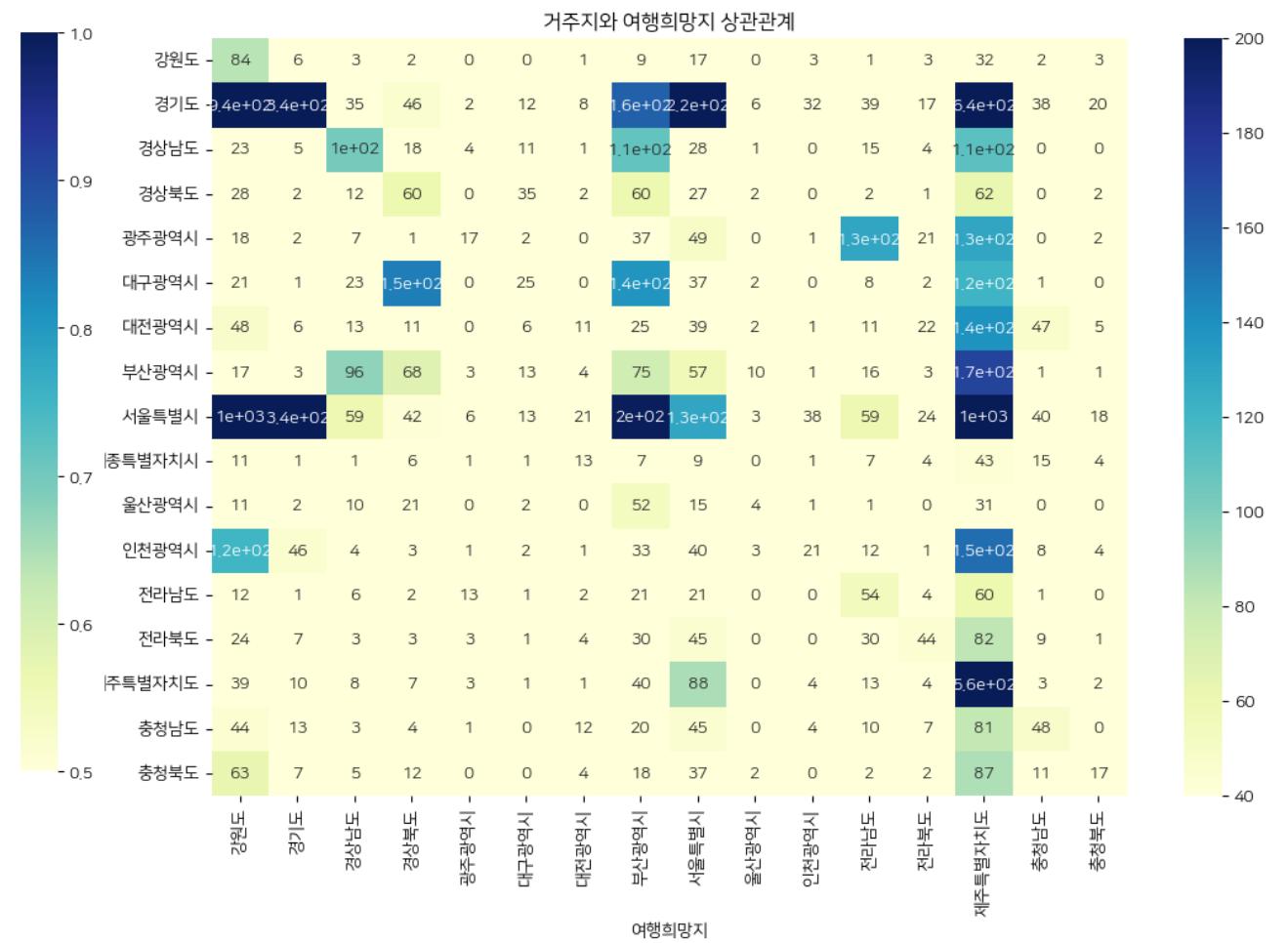
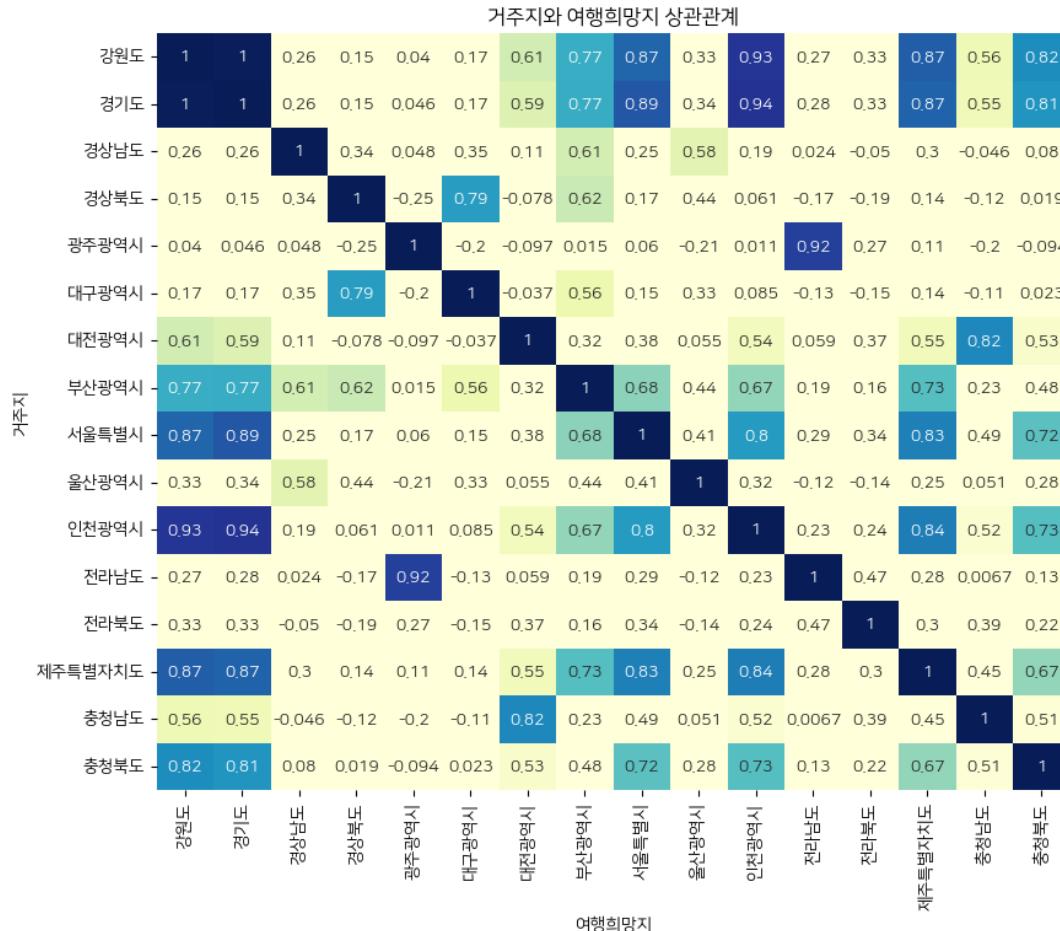
6) 여행 스타일 및 희망하는 여행지



여행 스타일의 응답이 자연 선호가 많지만 다양한 편
희망하는 여행지도 여행 스타일 응답 비율과 비슷한 빈도



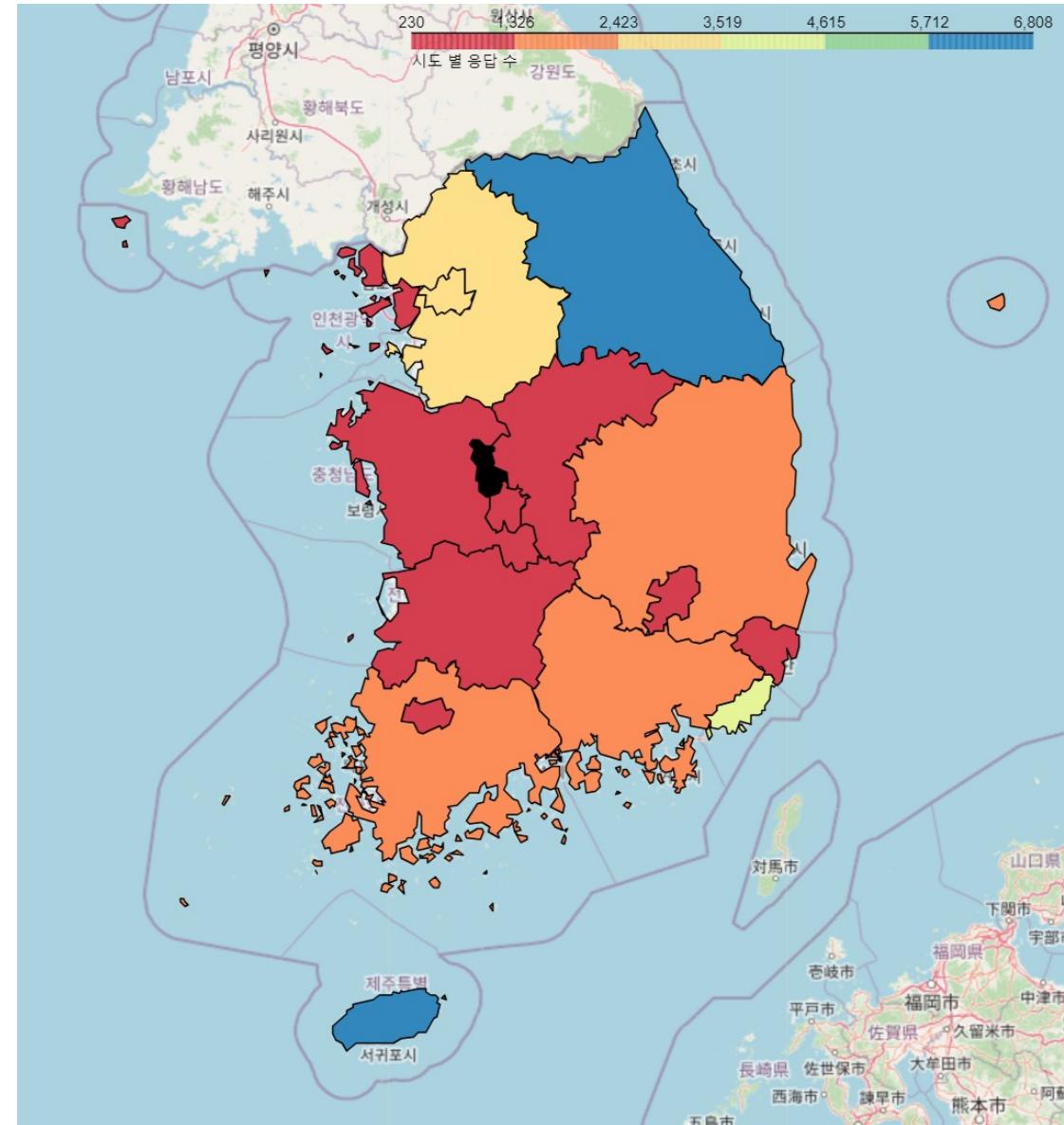
7) 거주지와 희망 여행지 상관관계



희망 여행지가 거주지와 상관없이 제주도가 많음.
상관관계로 볼 때, 근교 혹은 광역시 별로 높음.

7-2) 지역별 희망 여행지 응답 수

희망 여행지의 응답 비율이 강원도, 제주도, 부산, 서울, 경기도가 높은 편.
그 중에서도 제주도, 강원도가 특히 높음.



8) EDA에 따른 데이터 분석 결과

1. 성별, 나이별 여행 분포

: 방문 여행지에 차이점이 있을것이라 판단하여 입력 데이터 선정

2,3. 날짜별 여행 분포, 여행 상태 분포

: 입력 데이터로 선정할 만한 데이터가 아님.
다만, 주말 여행 비율 많은 점은 거주지가 여행지 추천에 유의미한 결과

4. 여행 장소별 여행 동기

: 유의미한 차이 존재라고 생각 / 여행 스타일과 비슷한 응답이라 배제

5. 지역에 따른 여행 만족도

: 지역별 희망여행지 응답수가 다름 / 만족도가 유의미하지 않다고 판단.

6. 여행 스타일에 따른 여행지

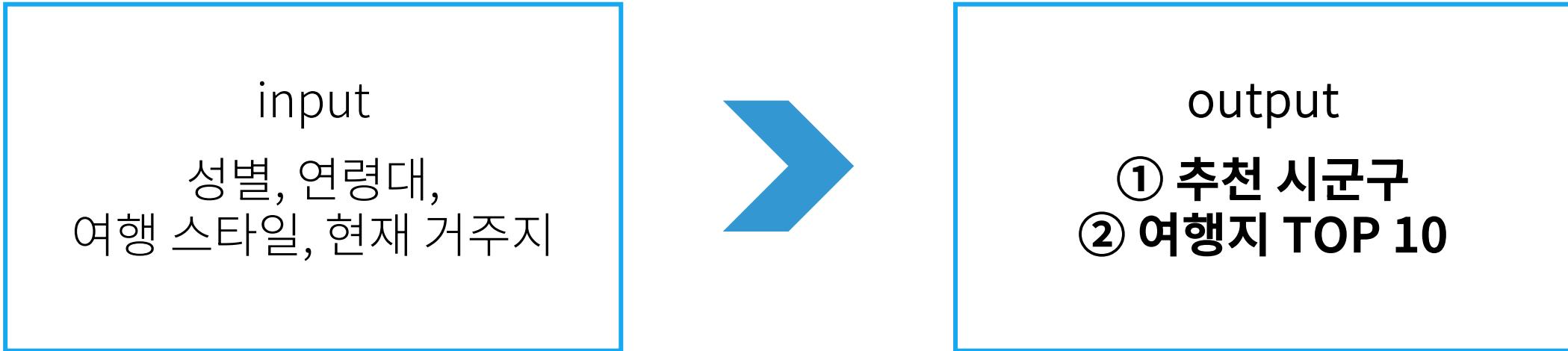
: 다양한 여행 스타일 분포, 선호 여행지 응답과 비슷하다 판단하여 입력 데이터 선정

7. 거주지 구분에 따른 가고싶은 여행지

: 거주지에 따른 선호 여행지 특성이 존재한다는 점에서 입력 데이터 선정

+ 지역별 여행 희망지 응답수가 특정 지역에 편향되어 다양한 여행지를 추천하고자 가중치를 두고자 함.

9) 입출력 데이터



4 전처리

전처리

1) 사용 데이터 현황

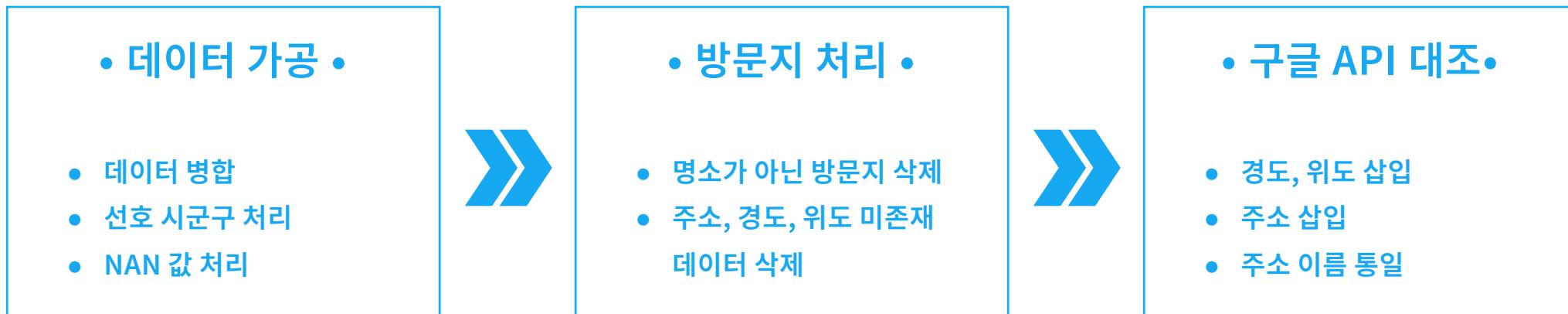
전체 데이터 크기

705.6 MB

주소 등 미정확 데이터 다수

```
23 TRAVEL_STYL_6           float64
24 TRAVEL_STYL_7           float64
25 TRAVEL_STYL_8           float64
26 TRAVEL_STATUS_RESIDENCE object
27 TRAVEL_STATUS_DESTINATION object
28 TRAVEL_STATUS_YMD       object
29 TRAVEL_MOTIVE_1          float64
30 TRAVEL_MOTIVE_2          float64
31 TRAVEL_MOTIVE_3          float64
32 VISIT_AREA_ID            int64
33 VISIT_AREA_NM             object
34 VISIT_START_YMD          object
35 VISIT_END_YMD            object
36 ROAD_NM_ADDR              object
37 X_COORD                   object
38 Y_COORD                   object
39 VISIT_AREA_TYPE_CD        int64
40 REVISIT_YN                 object
41 DGSTFN                    float64
42 REVISIT_INTENTION         float64
43 RCMDTN_INTENTION          float64
44 TRIP_ID                   object
45 START_VISIT_AREA_ID        object
46 END_VISIT_AREA_ID          object
47 START_DT_MIN                object
48 END_DT_MIN                  object
49 MVMN_CD_1                  float64
50 MVMN_CD_2                  float64
dtypes: float64(23), int64(2), object(26)
memory usage: 705.6+ MB
```

2) 전처리 진행 과정



전처리

2) 전처리 진행 과정

• 데이터 가공 •

- 데이터 병합
- 선호 시군구 처리
- NAN 값 처리

• 방문지 처리 •

- 명소가 아닌 방문지 삭제
- 주소, 경도, 위도 미존재
데이터 삭제

• 구글 API 대조•

- 경도, 위도 삽입
- 주소 삽입
- 주소 이름 통일

TRAVEL_LIKE_SGG_1
11500
45110
42210
50110
42210

TRAVEL_LIKE_SGG_1
서울특별시 강서구
전라북도 전주시 완산구
강원도 속초시
제주특별자치도 제주시
강원도 속초시
부산광역시 남구
경기도 가평군
강원도 강릉시
전라남도 화순군
서울특별시 강서구

```
# 21(집), 22(친구/친지집), 23(사무실), 9(역, 터미널, 고속도로 휴게소) 값 제거
visit = visit[~visit['VISIT_AREA_TYPE_CD'].isin([21, 22, 23, 9])] # 24,526개 제거
```

주차장 지우기

```
# "주차장"이 포함된 행 삭제
visit = visit[~visit['VISIT_AREA_NM'].str.contains('주차장', na=False)] # 161개 삭제
```

```
# ROAD_NM_ADDR, X_COORD, Y_COORD가 모두 비어있는 행을 필터링하여 삭제
visit = visit[~(visit['ROAD_NM_ADDR'].isna() & visit['X_COORD'].isna() & visit['Y_COORD'].isna())] # 1063개 삭제
```

전처리

2) 전처리 진행 과정

• 데이터 가공 •

- 데이터 병합
- 선호 시군구 처리
- NAN 값 처리



• 방문지 처리 •

- 명소가 아닌 방문지 삭제
- 주소, 경도, 위도 미존재
데이터 삭제



• 구글 API 대조•

- 경도, 위도 삽입
- 주소 삽입
- 주소 이름 통일

VISIT_AREA_ID	TRAVEL_ID	VISIT_AREA_NM	VISIT_START_YMD	VISIT_END_YMD	ROAD_NM_ADDR	X_COORD	Y_COORD
1	2.304300e+09	e_e000004	창룡문	2023-04-30	2023-04-30	NaN	127.025143 37.287791

1	2.304300e+09	e_e000004	창룡문	2023-04-30	2023-04-30	대한민국 경기도 수원시 팔달구 지동 경수대로 697	127.025143 37.287791
---	--------------	-----------	-----	------------	------------	------------------------------	----------------------

1	2.304300e+09	e_e000004	창룡문	2023-04-30	2023-04-30	경기도 수원시 팔달구 지동 경수대로 697	127.025143 37.287791
---	--------------	-----------	-----	------------	------------	-------------------------	----------------------

5

ML

1) 사용 모델 후보

Cat boost

- CATBoost는 Yandex에서 개발한 그라디언트 부스팅 알고리즘으로, 범주형 데이터를 처리하는 게 강점

<장점>

- 범주형 데이터 처리
 - 고성능
 - 과적합 방지
 - 손쉬운 사용

<단점>

- 복잡성
- 리소스 사용

콘텐츠 기반 필터링

- 추천시스템은 사용자와 항목 간의 상호 작용 데이터를 기반으로 개인화된 추천을 제공하는 시스템

<장점>

- 개인화 추천
 - 단순성
 - 확장성

<단점>

- 데이터 의존성
- 추천의 다양성 부족

Light GBM

- LightGBM은 Microsoft에서 개발한 그라디언트 부스팅 프레임워크로, 대용량 데이터와 빠른 학습 속도

<장점>

- 빠른 학습 속도
- 높은 예측 성능
- 효율적인 메모리 사용
- 자동 특징 중요도 평가

<단점>

- 복잡성
- 과적합 위험
- 특징 엔지니어링 필요

머신러닝

2) 선정 모델

	장점	단점	입출력데이터 구현여부	선정여부	비선정 사유
Cat boost	<ul style="list-style-type: none"> - 범주형 데이터 처리 용이 - 고성능, 손쉬운 사용 - 과적합 방지 	<ul style="list-style-type: none"> - 복잡성 - 리소스 사용. 	O	O	-
콘텐츠 기반 필터링	<ul style="list-style-type: none"> - 개인화 추천에 강함. - 단순성 - 확장성 	<ul style="list-style-type: none"> - 데이터 의존성 - 추천의 다양성 부족 	O	X	데이터 의존도가 높아 원하는 결과 도출 불가
Light GBM	<ul style="list-style-type: none"> - 빠른 학습 속도, 높은 예측 성능 - 효율적인 메모리 사용 - 자동 특징 중요도 평가 	<ul style="list-style-type: none"> - 복잡성, 과적합 위험 - 특징 엔지니어링 필요 	O	X	정성적으로 평가 했을 때, Catboost 모델에 비해 떨어진다고 판단

3) 머신러닝 진행 - CATBOOST

Label Encoder 사용

성별, 거주지, 여행스타일, 여행추천지

가중치 설정

여행 추천지의 빈도수 계산

빈도수에 따른 가중치 설정

모델 훈련

반복횟수 : 100

학습속도 : 0.1

깊이(depth) : 6

```
# 범주형 변수를 숫자로 인코딩
label_encoders = {
    'GENDER': LabelEncoder(),
    'TRAVEL_STATUS_RESIDENCE': LabelEncoder(),
    'STYL1': LabelEncoder(),
    'TRAVEL_LIKE_SGG_1': LabelEncoder()
}

for col, encoder in label_encoders.items():
    use_data[col] = encoder.fit_transform(use_data[col])

# 입력 데이터와 출력 데이터
X = use_data[['AGE_GRP', 'GENDER', 'TRAVEL_STATUS_RESIDENCE', 'STYL1']]
y = use_data['TRAVEL_LIKE_SGG_1']

# 빈도수 계산 및 가중치 설정
class_counts = y.value_counts()
class_weights = {cls: 1.0 / count for cls, count in class_counts.items()}
use_data['weight'] = use_data['TRAVEL_LIKE_SGG_1'].map(class_weights)

# 데이터 분할
X_train, X_test, y_train, y_test, weights_train, weights_test = train_test_split(X, y, test_size=0.2, random_state=42, stratify=y)

# Destination 예측 모델
model = CatBoostClassifier(iterations=100, learning_rate=0.1, depth=6, verbose=10)
model.fit(X_train, y_train, sample_weight=weights_train)
```

머신러닝

3) 머신러닝 진행 - CATBOOST

명소 추천 함수 적용

추천 여행지 방문 빈도 필터링

랜덤 예측 데이터 입력

10개의 데이터 출력

라벨인코딩 적용

```
# 명소 추천 함수 정의
def recommend_attractions(destination, visit_data, top_n=10):
    attractions = visit_data[visit_data['LOCATION'] == destination]
    attraction_counts = attractions['VISIT_AREA_NM'].value_counts().head(top_n)
    top_attractions = attractions[attractions['VISIT_AREA_NM'].isin(attraction_counts.index)]
    return top_attractions.drop_duplicates(subset=['VISIT_AREA_NM'])

# 새로운 데이터 생성 함수
def generate_random_data(n_samples):
    np.random.seed(43)
    genders = np.random.choice(['남', '여'], size=n_samples)
    ages = np.random.choice([20, 30, 40, 50, 60], size=n_samples)
    residences = np.random.choice(sido_all, size=n_samples)
    travel_styles = {
        'STYL1': np.random.choice(['자연 선호 약간선호', '자연 선호 중간선호', '자연 선호 매우선호'],
                                  size=n_samples)
    }
    data = [
        'GENDER': genders,
        'AGE_GRP': ages,
        'TRAVEL_STATUS_RESIDENCE': residences,
        'STYL1': travel_styles['STYL1']
    ]
    new_df = pd.DataFrame(data)
    return new_df
```

6

분석 및 결론

결과 분석

1) 머신러닝 예측 데이터 결과

GENDER: 남, AGE: 50, RESIDENCE: 충청남도, TRAVEL STYLE: 도시 선호 약간선호

추천 여행지: 대전광역시 유성구

추천 명소: 국립 중앙과학관, 엑스포과학공원 한빛탑, 롯데시티 호텔 대전, 대전 신세계 아트앤사이언스, 오씨칼국수 도룡점, 성심당 DCC점, 대전 어린이 회관

=====

GENDER: 남, AGE: 40, RESIDENCE: 대구광역시, TRAVEL STYLE: 자연 선호 약간선호

추천 여행지: 경상북도 영덕군

추천 명소: 고래불해수욕장, 스포트리조트, 영덕해맞이공원, 삼사해상공원, 벌목 메타세쿼이아 숲, 장사상륙작전 전승기념관, 대개 궁, 장사해수욕장, 영덕 해

=====

GENDER: 여, AGE: 20, RESIDENCE: 인천광역시, TRAVEL STYLE: 자연 선호 중간선호

추천 여행지: 인천광역시 남동구

추천 명소: 통 큰 해물 손칼국수 소래포구 본점, 소래포구 종합어시장, 파크 마린 호텔, 인천대공원, 라마다 인천 호텔, 소래습지생태공원, 메가박

=====

GENDER: 여, AGE: 50, RESIDENCE: 광주광역시, TRAVEL STYLE: 자연 선호 매우선호

추천 여행지: 전라남도 영광군

추천 명소: 불갑사, 백수해안도로, 영광 칠산 다워, 야월 교회, 카페 보리, 국제식당, 백제불교 최초 도래지 부용리, 영광 물 무산 행복 숲 질퍽질퍽 맨발

=====

GENDER: 여, AGE: 40, RESIDENCE: 광주광역시, TRAVEL STYLE: 자연 선호 약간선호

추천 여행지: 전라남도 완도군

추천 명소: 방파제, 범바위, 완도타워, 신지명사십리해수욕장, 서편제촬영지, 완도네시아, 지리해수욕장, 카페마르, 신흥해수욕장, 청산도게스트하우스

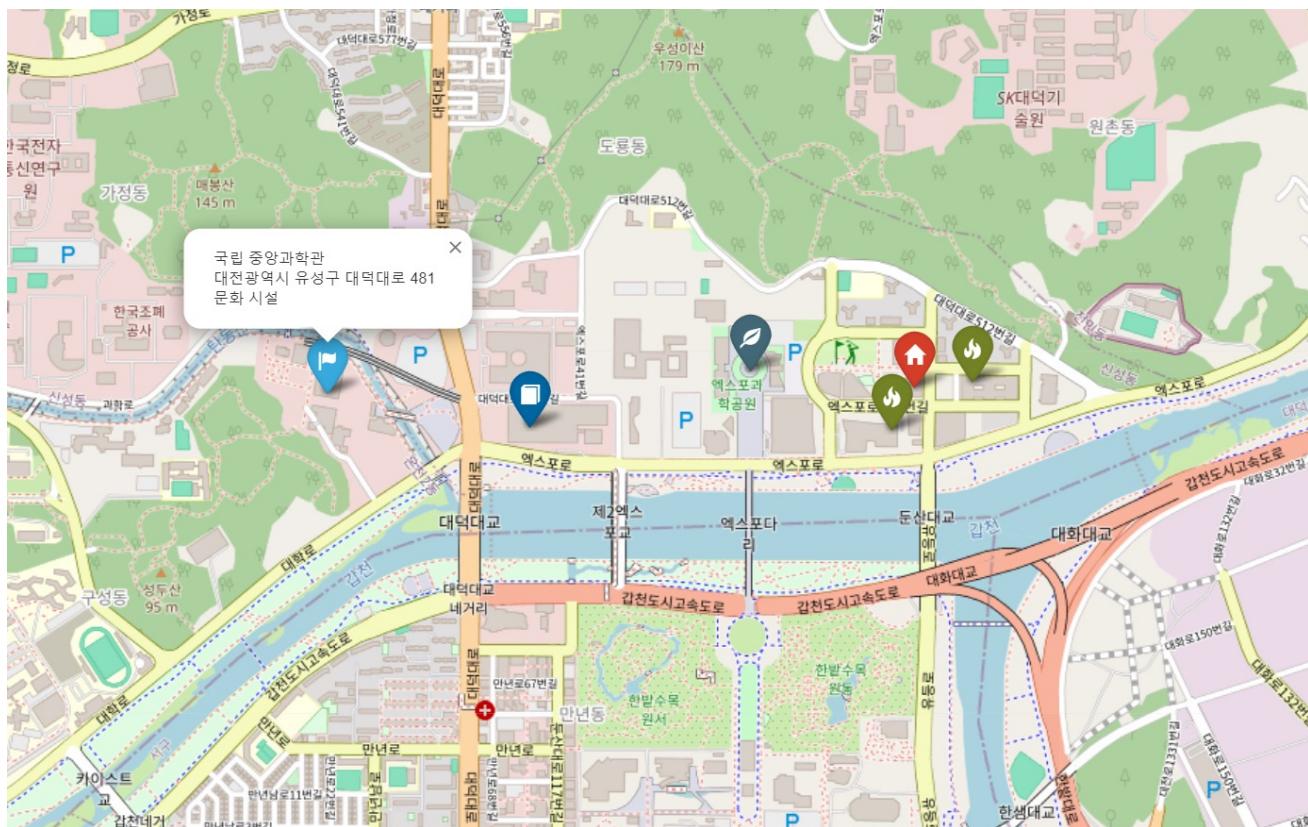
결과 분석

1) 머신러닝 예측 데이터 결과(명소 지도)

GENDER: 남, AGE: 50, RESIDENCE: 충청남도, TRAVEL STYLE: 도시 선호 약간선호

추천 여행지: 대전광역시 유성구

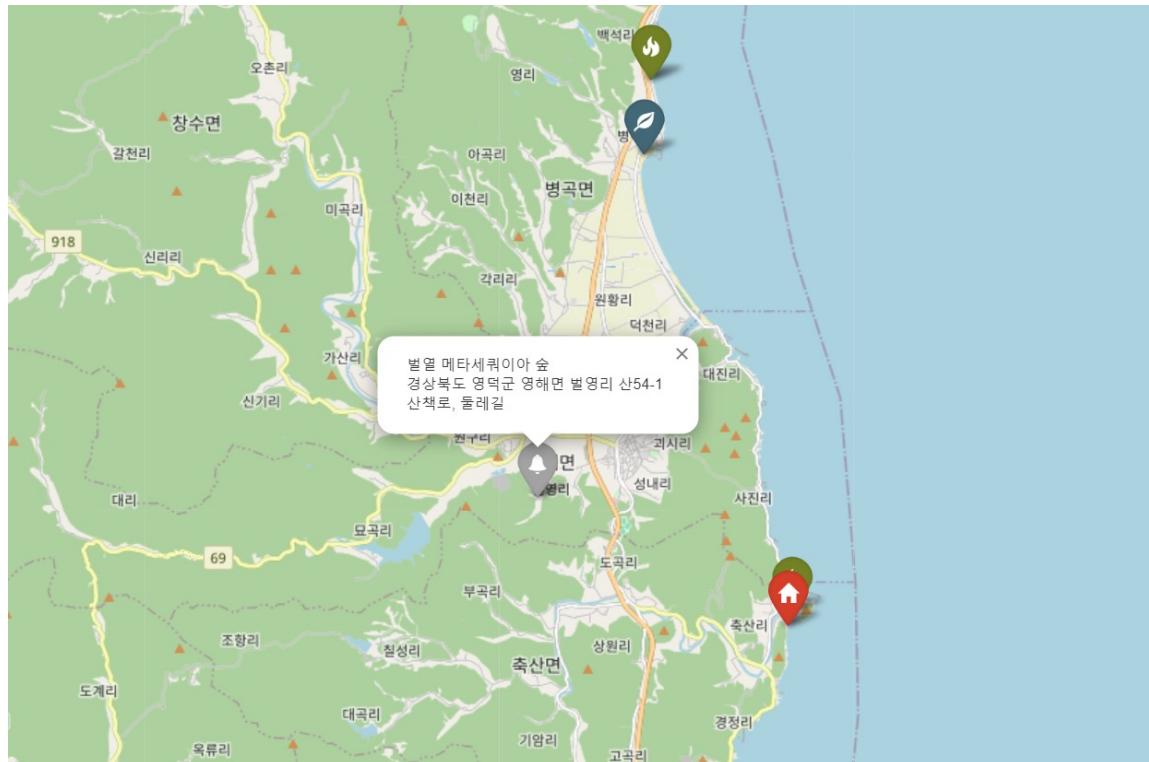
추천 명소: 국립 중앙과학관, 엑스포과학공원 한빛탑, 롯데시티 호텔 대전, 대전 신세계 아트앤사이언스, 오씨칼국수 도룡점, 성심당 DCC점, 대전 어린이 회관



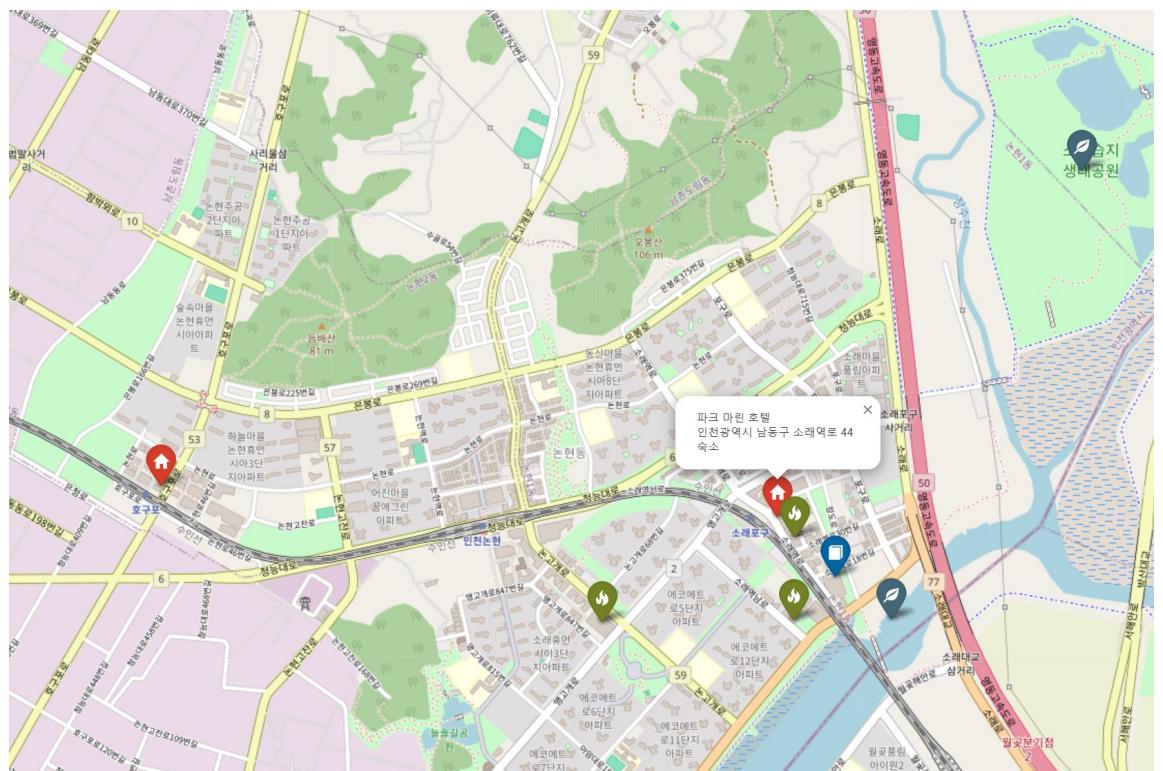
결과 분석

1) 머신러닝 예측 데이터 결과(명소 지도)

GENDER: 남, AGE: 40, RESIDENCE: 대구광역시, TRAVEL STYLE: 자연
추천 여행지: 경상북도 영덕군
추천 명소: 고래불해수욕장, 스포트리조트, 영덕해맞이공원, 삼사해상공원, 벌열



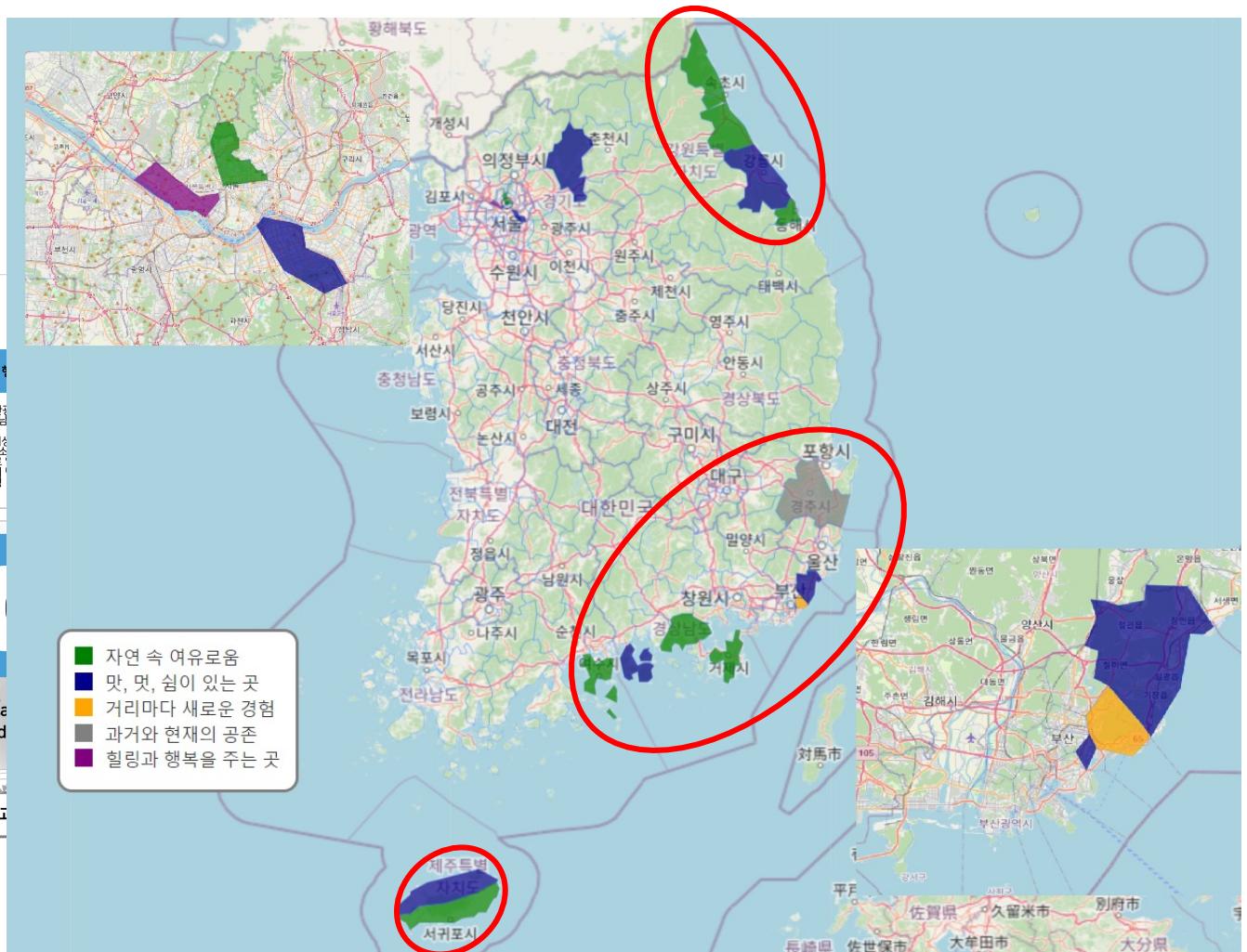
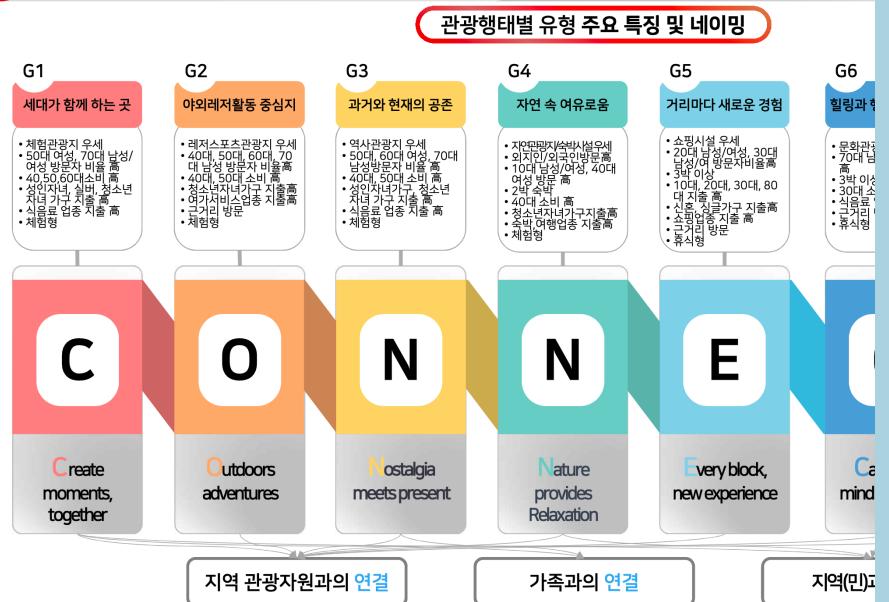
GENDER: 여, AGE: 20, RESIDENCE: 인천광역시, TRAVEL STYLE: 자연
추천 여행지: 인천광역시 남동구
추천 명소: 통 큰 해물 손칼국수 소래포구 본점, 소래포구, 소래포구 종합어시장,



결과 분석

2) EDA 선호 여행지 분석

한국관광데이터랩 - 관광유형 고도화 보고서



결과 분석

3) 추천 명소 비교

한국관광데이터랩

<https://datalab.visitkorea.or.kr/datalab/portal/loc/getTourLocalDataF>

한국관광 데이터랩

지역별 분석

국가별 분석

관광산업 분석

빅데이터

관광통계/실태조사

데이터랩 소개

경기도 수원시 장안구

기간 202307 ~ 202406 조회

전체 다운로드 인쇄 데이터 특성/수정

경기도 수원시 장안구 중심 관광지 총 100 건

이름 검색 조희 0 / 0 <>

순위	스탈필드/수원	경기도 수원시 장안구	유형
1	스탈필드/수원	경기도 수원시 장안구	쇼핑
2	일월수목원	경기도 수원시 장안구	자연관광
3	만석공원	경기도 수원시 장안구	문화관광
4	북수원온천	경기도 수원시 장안구	체험관광
5	수원KT워즈파크	경기도 수원시 장안구	문화관광

스탈필드/수원 연관 관광지 총 50 건

순위	이름	주소	유형
1	AK플라자/수원	경기도 수원시 팔달구	쇼핑
2	갤러리아백화점/광교점	경기도 수원시 영통구	쇼핑
3	롯데몰/수원점	경기도 수원시 권선구	쇼핑
4	화성행궁	경기도 수원시 팔달구	역사관광
5	롯데백화점/수원점	경기도 수원시 권선구	쇼핑

스탈필드/수원 연관 관광지 지도

예측 데이터 추천 여행지	일치 개수
대전광역시 유성구	6개
경상북도 영덕군	8개
인천광역시 남동구	8개
전라남도 영광군	6개
전라남도 완도군	6개
부산광역시 사하구	10개
경상북도 안동시	6개
경기도 군포시	2개
충청북도 충주시	5개

여행
스타일



관광
지역, 명소

‘관광 명소, 너무 유명하지 않은 장소’

감사합니다