**데이터기반 행정으로 국민의 삶의질을 개선하라!**

**데이턴십 해커톤 제 4회**

|  |
| --- |
| **차박 쓰레기 무단투기 해결을 위한, 쓰레기봉투 자판기 최적입지 선정 및 예측** |

분석 결과보고서

|  |  |
| --- | --- |
| 참여조 : | 104조 |
| 참여자 : | 문상수(조장) |
|  | 김나경 |
|  | 김영인 |
|  | 이세은 |
|  | 최윤진 |
|  |  |



**목 차**

[1 차박지 쓰레기 봉투 자판기 최적 입지 분석모델 개요 2](#_Toc80115407)

[**1.1** **분석 배경 및 개요** 2](#_Toc80115408)

[**1.2** **분석목적 및 방향** 8](#_Toc80115409)

[**1.3** **분석 결과 활용 방안** 9](#_Toc80115410)

[2 분석 데이터 9](#_Toc80115411)

[**2.1** **분석데이터 목록** 9](#_Toc80115412)

[**2.2** **데이터 상세 설명** 11](#_Toc80115413)

[**2.3** **데이터 정제 방안** 15](#_Toc80115414)

[3 분석 프로세스 22](#_Toc80115415)

[**3.1** **분석 프로세스** 22](#_Toc80115416)

[**3.2** **분석 내용 및 방법** 23](#_Toc80115417)

[**3.3** **다중선형회귀모델을 이용한, 차박지 수요 예측** 30](#_Toc80115418)

[4 봉투 자판기 최적입지 분석결과 30](#_Toc80115419)

[**4.1** **EDA 및 분석결과 시각화** 30](#_Toc80115420)

[**4.2** **통계 분석 결과(~회귀분석)** 33](#_Toc80115421)

[**4.3** **최적입지분석 결과** 35](#_Toc80115422)

[**4.4** **회귀분석 결과를 바탕으로 잠재적 차박지 수요 예측** 38](#_Toc80115423)

[5 활용 방안 39](#_Toc80115424)

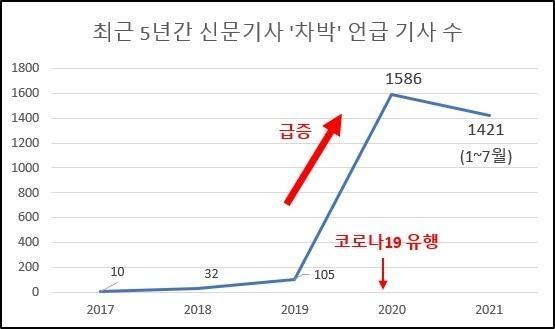
[**5.1** **문제점 개선 방안** 39](#_Toc80115425)

[**5.2** **업무 활용 방안** 39](#_Toc80115426)

[6 참고자료 40](#_Toc80115427)

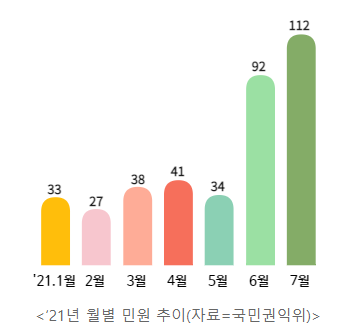
[7 부록 42](#_Toc80115428)

1. **차박지 쓰레기 봉투 자판기 최적 입지 분석모델 개요**
   1. **분석 배경 및 개요**
      1. 최근 차박의 인기 급증에 따른 각종 문제와 민원 발생



[그림 1] 최근 차박 관련 기사 수 증가 그래프

* 코로나 19로 여행지 선택의 폭이 줄어들면서 차에서 숙박하는 이른바 ‘차박’의 유행이 계속되고 있음. 문화체육관광 국민소통실에서 제공한 국민 일상생활과 관련된 SNS분석결과에서의 ‘차박’ 언급량은 코로나 19 이전 대비 223% 증가하였음. 국토 교통부와 통계청에 따르면 캠핑 인구는 약 600만명으로 추산.
* 2010년부터 2021년 7월까지 총 3,181건 기사 중 코로나 19 이후 2020년 1586건으로 급증하였음



[그림 2] 2021년 국민권익위 월별 민원 추이

* 여름 휴가철을 맞아 7-8월 중 국민권익위원회는 ‘휴가철 야외활동’ 관련 민원 예보를 발령함, 국민 권익위 집계에 따르면 최근 3년 6개월간(2018. 1월~ 2021.6월) ‘휴가철 야외활동’과 관련한 민원은 총 2,147건으로, 방학 및 휴가기간인 7, 8월에 다수 발생한 것으로 나타남
* 특히 ‘휴가철 야외활동’과 관련해 나타난 주요 민원은 ▴공원 등에서의 무단 캠핑 및 쓰레기 투기 단속 요청 ▴부당 요금징수 ▴방역수칙 위반행위 단속 등이 다수를 차지함
  + 1. 차박과 관련한 이슈로, 쓰레기 무단투기 문제가 급증

출처 : 제주환경일보(<http://www.newsje.com)> 화이트보드이(가) 표시된 사진

자동 생성된 설명

[그림 3] 차박기사 wordcloud

* ‘차박’단어를 포함하는 신문 Wordcloud 결과, ‘캠핑’,’코로나19’키워드가 가장 많이 언급됨. 그 다음으로 ‘차량’, ‘쓰레기’ 키워드가 언급되어 쓰레기 문제가 발생했다는 것을 알 수 있음
* 차박 커뮤니티에서 약 한달간 차박 캠핑을 즐기는 캠퍼들에게 설문조사를 진행. 차박 캠퍼들은 30.4% 이상이 쓰레기 문제를 매우 심각하다 라고 답했고, 차박에 대한 이미지를 묻는 물음에는 56.4%가 쓰레기를 마구 버리는 행태를 가장 먼저 답함.
* 실제 자신의 쓰레기를 어떻게 처리하느냐는 질문에 전체 응답자중 47.5%가 캠핑 중 발생하는 쓰레기를 봉투에 넣어 다시 가지고 온다고 답했고, 29.4% 현지의 종량제 봉투를 구매해 분리 수거한다고 응답했음.
  + 1. 차박지 폐쇄, 단속 등 지자체가 방안을 내놓고 있지만 한계
* 차박족들이 무단투기한 쓰레기를 치우는 것은 오롯이 지역 주민들의 몫이 되고 있음. 기장군의 경우 차박족들이 버리고 간 쓰레기 양이 워낙 많다 보니 마을 주민의 힘으로 처리가 안돼 자원봉사자들의 도움을 받고 있음

텍스트, 하늘, 실외, 휴지통이(가) 표시된 사진

자동 생성된 설명

[그림 4] -부산 기장군 차박지 무단투기 현장

* 차박 명소에 해당하는 부산 기장군, 강원 속초시, 김천시 등 지차제들은 차박지를 폐쇄하는 행정명령을 발령함. 또, 쓰레기 불법투기 단속에 힘쓰고 있지만 불법 차박에 대한 단속 근거가 미비하고 장소마다 담당부서가 달라서 해결에 난항
* 한 캠핑장을 폐쇄하면 다른 캠핑장으로 몰려 쓰레기 문제를 발생시킴. 폐쇄되는 곳들이 많이 생기자 캠핑 고수들 사이에는 차박 명소를 알리지 않는 분위기까지 형성. 쓰레기 처리가 더 힘들어짐

텍스트, 실외, 하늘, 대지이(가) 표시된 사진

자동 생성된 설명

[그림 5] 기장군 차박지 폐쇄 행정명령 현수막

* + 1. 차박지 폐쇄와 같은 단순 금지보다는 문제 해결에 초점을 두는 것이 필요
* 최근 차박 열풍으로 여러가지 문제들이 부각되고 있지만, 지자체들이 예산을 투입해 합법적인 차박지를 조성하면 지역경제도 살리고 국내 관광 산업도 활성화하는 1석 2조의 효과를 거둘 수 있음
* 충주 수주팔봉, 홍성군 남당항 등 기존 주차장 폐쇄 와 같은 강압적인 단속 대신, 민관 협력을 통해 차박족을 관광산업으로 흡수하여 주민과 상생하는 캠핑문화를 조성중임
  + 1. 차박지 쓰레기 봉투 판매 필요성이 대두됨.
* 차박지가 종량제 봉투 판매처와 멀어 종량제 봉투의 구입이 어렵다는 관광객들의 민원이 발생함. 이로 인해 종량제 봉투가 아닌 일반 봉투에 쓰레기를 담아 무단 투기를 하는 경우가 다수 발생하고 있음
* 차박 시 발생하는 음식물 쓰레기를 공중화장실에 투기하거나 일반쓰레기와 함께 투기하는 문제가 다수 발생함
* 전화를 통한 부산시 기장군 청소자원과 담당자 인터뷰 결과, 쓰레기통이나 쓰레기장을 설치·운영하는 데는 큰 비용이 들기 때문에 모든 지역에 쓰레기 처리시설을 배치하는 것은 현실적으로 불가능한 실정임 따라서 쓰레기를 되가져갈 수 있도록 종량제 봉투를 비롯한 회수용 봉투 필요
* 따라서, 다중회귀분석을 통해 차박 수요에 영향을 미치는 요인을 도출하고, ‘차박 쓰레기 취약 지역’을 선정하여 쓰레기 봉투 자판기 우선/최적입지를 제안하여 차박 명소의 쓰레기 문제 완화 (분석 개요)
  1. **분석목적 및 방향**
     1. 차박 쓰레기 무단 투기 지역 예측
* 회귀분석을 통해 차박지 수요와 관련된 요인들을 도출하고 이를 바탕으로 ‘차박지 수요 예측모델’ 개발
* 예측모델을 활용하여 지자체가 해당 지역의 차박지 수요를 예측하고, 선제적 대응을 할 수 있도록 도움
  + 1. 쓰레기 무단투기 문제의 새로운 해결방안 제시
* 쓰레기 봉투 자판기를 배치함으로써 캠핑족(차박족)의 종량제봉투 사용 혹은 자발적인 쓰레기 회수 유도
* 쓰레기 봉투 자판기의 최적 위치 제시
  + 예측모델의 계수를 바탕으로 가중치 도출
  + 쓰레기 봉투 자판기 우선 배치 순위 선정
* 쓰레기문제 해결과 동시에, 단순히 차박지 폐쇄가 아닌 지역 관광지 활성화 방안 논의
  1. **분석 결과 활용 방안**
     1. 지자체 업무 활용의 측면
* 차박지 쓰레기 무단 투기 문제 개선 및 민원 감소 예상
* 증가하는 차박의 인기에 지자체가 사전 대응하도록 차박지의 수요를 미리 예측하고, 무단투기 예방 조치 가능
  + 1. 행정비용 절감의 측면
* 지자체 별로 쓰레기 처리 인원과 예산이 한정되어 있어   
  수많은 차박 지역에 대한 관리인력 운용이 현실적으로 불가
  + 부산시 기장군 청소자원과 담당자 인터뷰 결과에 따르면, 모든 지역에 분리수거장 등의 쓰레기 처리 시설을 배치하거나, 관리 인력을 배치하는 것은 비용 문제 때문에 현실적으로 불가
* 자판기를 통해 캠핑(차박)족의 자발적인 쓰레기 회수(또는 배출) 유도
  + 지자체의 노지 쓰레기 무단투기 처리비용 절감

1. **분석 데이터**
   1. **분석데이터 목록**

[표 1]

|  |  |  |  |
| --- | --- | --- | --- |
| 구분 | 수집 데이터 | 기간 | 제공 기관 |
| 차박지 쓰레기  관련 데이터 | ‘차박 쓰레기, 언급  뉴스 데이터(제목, 본문) | 2010. 1~ 2021. 8 | 빅카인즈 |
| 네이버  블로그 데이터 | 2020.1 ~ 2021. 8 | 네이버 블로그 |
| SNS 데이터 | 2020.3 ~ 2021. 8 | 인스타그램 |
| 주요 차박지  데이터 | 네이버 카페 데이터 | 2020.1 ~ 2021. 8 | 네이버 카페  (차박캠핑클럽,  차박은 내친구) |
| 네이버  블로그 데이터 |  | 네이버 블로그 |
| SNS 데이터 | ~ 2021. 8 | 인스타그램 |
|  | 전국 야영장 표준데이터 | ~ 2021. 3 | 공공데이터포털 |
| 차박지 주변시설  관련 데이터 | 2020 전국 편의점 주소록 CD | 2020 | 한국콘텐처미디어 |
| 전국 주차장 면적 데이터 | 행정구역별로 다름 | 공공데이터포털 |
| 주차장 정보 데이터 | ~ 2021. 8 | 건축데이터 민간개방 시스템  (eais.go.kr) |
| 해수욕장 해변폭, 해변 연장 정보 | ~ 2021. 8 | 공공데이터포털 |
| 전국 공중화장실 표준데이터 | 행정구역별로 다름 | 공공데이터포털 |
| KAKAO Map 화장실 데이터 | ~ 2021. 8 | 카카오맵 API |
| 차박지 주변 음식점 데이터 | ~ 2021. 8 | 카카오맵 API |
| 차박지역의 면적 데이터 | 시도 행정구역데이터 | 2020 | GEO DEVELOPER |
|  | Geocode 좌표 데이터  (Google 지도로 부터 추출) | 수시 | 실시간 |
|  | 차박 관련 설문조사 | 2021.08.06~ 2021.08~13 | 1회 |

* 1. **데이터 상세 설명**

표 2

|  |  |  |  |
| --- | --- | --- | --- |
| 구분 | 수집 데이터 | 데이터 형식 | 생성 주기 |
| 공공 데이터 | 전국 야영장 표준데이터 | excel | 공공데이터포털 |
| 전국 주차장 면적 데이터 | excel | 월간 |
| 주차장 면적 데이터 | excel | 월간 |
| 해수욕장 해변폭, 해변 연장 정보 | excel | 연간 |
| 전국 공중화장실 표준데이터 | excel | 연간 |
| 시도 행정구역데이터 | shp | - |
| 민간 데이터 | ‘차박 쓰레기, 언급  뉴스 데이터(제목, 본문) | 비정형(텍스트) | 일간 |
| 네이버  블로그 데이터 | 비정형(텍스트) | 수시 |
| SNS 데이터 | 비정형(텍스트) | 수시 |
| 네이버 카페 데이터 | 비정형(텍스트) | 수시 |
| 네이버  블로그 데이터 | 비정형(텍스트) | 수시 |
| SNS 데이터 | 비정형(텍스트) | 수시 |
| 전국 편의점 주소록 CD | excel | 연간 |
| KAKAO Map 화장실 | 비정형(텍스트) | 수시 |
| 차박지 주변 음식점 데이터 | 비정형(텍스트) | 수시 |
| Geocode 좌표 데이터  (Google 지도로 부터 추출) | excel | 실시간 |
| 차박 관련 설문조사 | 비정형(텍스트) | 1회 |

* + 1. 데이터 설명
       1. SNS(인스타그램) 데이터
* 주요 차박지를 도출하고 해당 지역의 언급 수(인기도) 추출에 활용하기 위해 크롤링한 데이터
* ‘차박’ 키워드로 검색했을 때 나오는 모든 게시글 ‘제목’과 ‘본문 정보’를 포함

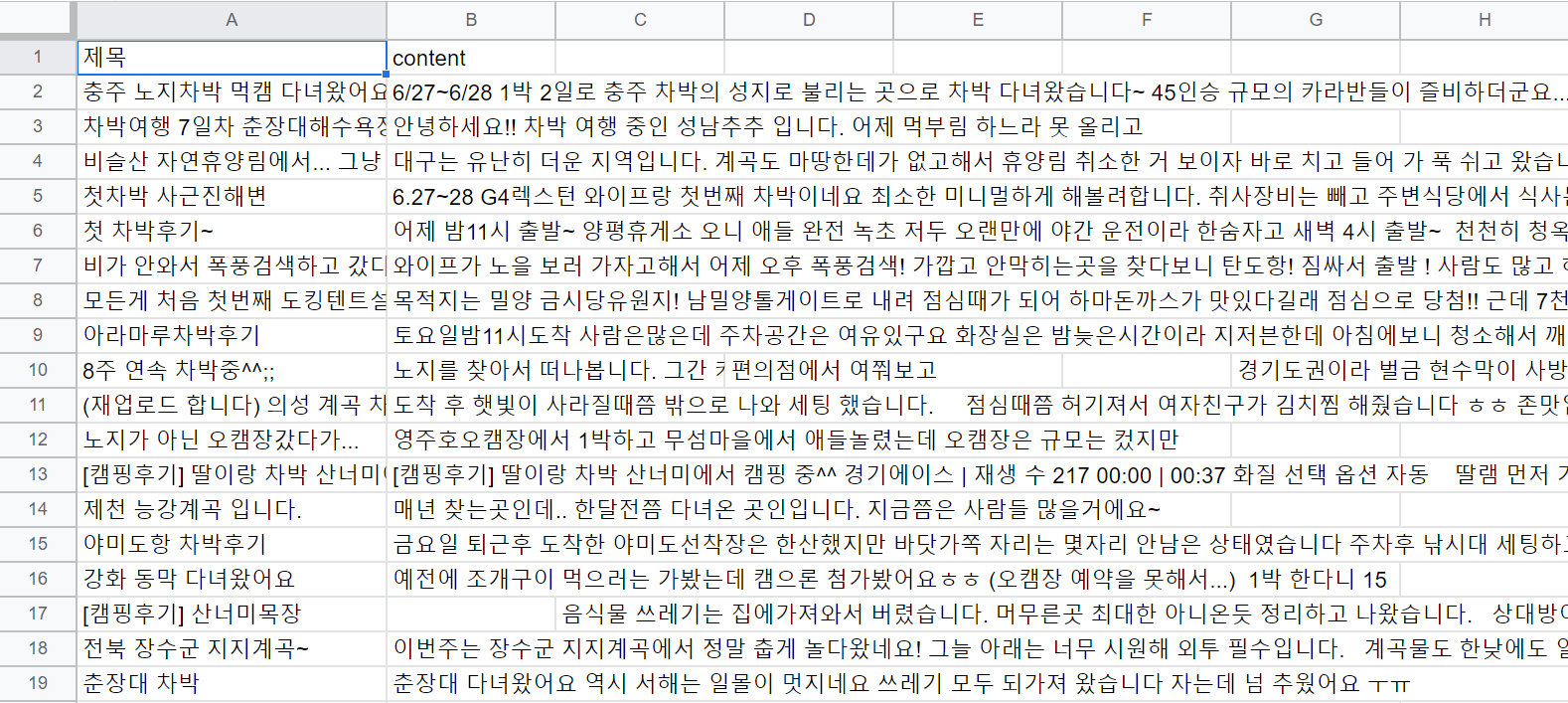


그림 6 인스타그램 차박 관련 게시글

* + - 1. 네이버 블로그 데이터
* 주요 차박지를 도출하고, 해당 지역의 언급 수(인기도) 추출에 활용하기 위해 크롤링한 데이터
* ‘차박’ 키워드로 검색했을 때 나오는 모든 블로그 게시물의 ‘제목’, ‘작성일시’, ‘본문’ 정보를 포함

텍스트이(가) 표시된 사진

자동 생성된 설명

그림 7 네이버 블로그 차박 관련 게시글

* + - 1. 네이버 카페 데이터
* 주요 차박지를 도출하고, 해당 지역의 언급 수(인기도) 추출에 활용하기 위해 크롤링한 데이터
* 차박 후기 게시판 글의 ‘제목’, ‘작성일시’, ‘본문’ 정보를 포함

텍스트이(가) 표시된 사진

자동 생성된 설명

그림 8 네이버 카페 차박 관련 게시글

* + - 1. 주차장 면적 데이터(공공데이터 2종)
* 차박지의 주차가능 면적을 구하기 위해 사용한 데이터
* ‘주소’, ‘해수욕장이름’, ‘길이’, ‘면적’ 등의 정보를 포함

테이블이(가) 표시된 사진

자동 생성된 설명

그림 9 공공데이터포털 주차장 면적 데이터

테이블이(가) 표시된 사진

자동 생성된 설명

그림 10 공공데이터포털 해수욕장 면적 데이터

* + - 1. 화장실, 편의점 데이터(kakaomap API 검색결과 크롤링)
* 검색결과 html을 parsing해서 ‘화장실(편의점)이름’, x좌표(경도), y좌표(위도)를 추출(BeautifulSoup 패키지 사용)

테이블이(가) 표시된 사진

자동 생성된 설명

그림 11 KAKAO MAP API 위도경도 데이터

* 1. **데이터 정제 방안**
     1. 데이터 수집방법

공공데이터로 구하기 부족한 정보는 웹 크롤링, API사용을 통해 구함

* + - 1. 인스타그램 데이터 수집방법
* ‘차박’ 키워드로 검색했을 때 나오는 모든 게시글 크롤링(제목, 본문)



그림 12 인스타그램 차박 언급 게시글 크롤링

* Selenium을 활용하여 인스타그램 접속 후, 차박 검색을 통해 얻은 결과물의 링크를 크롤링하여 csv에 저장.텍스트이(가) 표시된 사진

  자동 생성된 설명

[그림 13] 인스타그램 크롤링 셀레니움 실행 화면

* Csv에 저장된 링크를 통해 해당 페이지 접속 후, BeautifulSoup 패키지를 활용하여 해당 게시글의 날짜, 장소, 본문 내용 크롤링.
  + - 1. 네이버 블로그 데이터 수집방법
* BeautifulSoup이용, 파싱 → 네이버 블로그 url 추출(리스트)
* Selenium 이용, url 리스트의 url을 열어 블로그 제목, 작성일자, 본문 내용을 크롤링하도록 코드 작성

텍스트이(가) 표시된 사진

자동 생성된 설명

그림 14 네이버 블로그 차박 언급 게시글 크롤링

* + - 1. 네이버 카페 데이터 수집방법
* 차박 커뮤니티 카페 ‘차박캠핑클럽’, ‘차박은내친구’의 차박 후기게시판 크롤링(제목, 본문)

텍스트이(가) 표시된 사진

자동 생성된 설명

그림 15 네이버 카페 차박 커뮤니티 크롤링 코드 화면

* 카페 주소로 이동한 후 1000페이지의 게시물의 url받아와서 list에 저장

텍스트이(가) 표시된 사진

자동 생성된 설명

그림 16 네이버 카페 크롤링 코드

* 받아온 url 주소를 이용하여 게시물을 연 후 필요한 데이터(제목, 게시판, 날짜, 내용, 댓글) 크롤링
  + - 1. 카카오맵 데이터 (화장실 & 편의점) 데이터 수집방법
* KakaoMap API를 사용
* ‘추출한 차박지역명’ + ‘화장실(편의점)’의 키워드로 검색한 결과값에서 위경도 좌표를 수집
  + 1. 데이터 정제
       1. SNS 데이터 정제를 통해 주요 차박지 추출 및 빈도수 파악
* 게시글의 제목 및 본문 텍스트에서 특수기호 제거 및 Pykospacing 패키지를 활용하여 띄어쓰기 검사 및 교정.
* 불필요한 문자열 제거를 위해 특정 길이 이하의 문자열 제거.
* 파이썬을 통해 SOYNLP 패키지를 활용하여 단어 토큰화 및 한국 SNS에서 주로 등장하는 ‘ㅋㅋ,ㅎㅎ’ 와 같은 반복되는 문자 정제.
* customized Konlpy 패키지를 활용하여 해수욕장, 캠핑장 등 차박지 추출을 위해 필요한 단어를 담은 사용자 사전을 추가.
* 이를 활용한 형태소 분석을 통해 주요 차박지 600여 곳을 담은 리스트 목록 생성.
* 추출한 주요 차박지 600여곳에 대하여 정제한 SNS 본문 데이터로부터 각 차박지 언급 횟수 카운트하여 csv로 저장.
  + - 1. 주차면적 관련 데이터 정제
* 주차면적은 차박을 주차장에서 하는 경우도 있고, 노지, 야영장 등 다양하기 때문에 차박 가능 지역을 모두 주차면적으로 산정
* 해수욕장, 해변은 해양수산부 해수욕장 정보목록에 있는 해수욕장 면적 데이터를 사용하여 데이터 수집 현실적인 차박 가능 면적을 반영하기 위해 백사장 넓이의 절반을 적용
* 캠핑장, 야영장의 면적은 카카오맵 검색결과를 사용
* 주차장의 면적은 국토교통부 건축행정시스템 세움터, 공공데이터 포털의 주차장 면적 데이터를 사용(공공데이터에 기록되어 있는 주차장의 경우)
* 공공데이터에 없는 주차장의 면적은, Qgis의 $area 면적 함수를 통해 계산

지도이(가) 표시된 사진

자동 생성된 설명

그림 17 QGIS 면적 데이터 화면

* 위 방법으로 구할 수 없는 장소의 면적은 위성지도를 참고하여 면적 어림계산
  + - 1. 가장 가까운 화장실, 편의점까지의 거리 값 도출
* 추출한 차박 지역의 위경도 좌표 추출(KakaoMap API 이용)
* ‘차박 지역명’ + ‘편의점’ 키워드로 KakaoMap API검색 결과 수집
* 검색결과(requests.response)에서 위경도 좌표를 추출(list)

텍스트이(가) 표시된 사진

자동 생성된 설명

그림 18 카카오맵 API 활용 코드

* QGIS에서 차박지역과 편의점의 거리 계산, 지역별로 거리의 최소값만을 추출
* ‘편의점’에 대해서도 동일한 과정으로 반복
  + 1. 음식점 개수 도출
* 추출한 차박 지역의 위경도 좌표 추출(KakaoMap API 이용)
* ‘차박 지역명’ + ‘근처’ + ‘음식점’ 키워드로 KakaoMap API 검색 결과 수집
* 검색결과에서 검색결과 개수 값을 추출
  + 1. 이상치(결측치) 대체(가장 가까운 화장실 거리)
* 캠핑장, 야영장의 경우 KakaoMap API에서 화장실 정보가 검색되지 않음
* 매우 먼(실질적으로 갈 수 없는) 거리의 화장실 거리가 입력됨
* 캠핑장, 야영장의 경우 화장실을 갖추고 있으므로 적당한 값으로 추정 필요
* 다음 수식으로 캠핑장 야영장의 화장실 거리를 추정

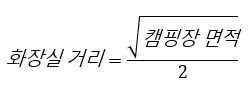
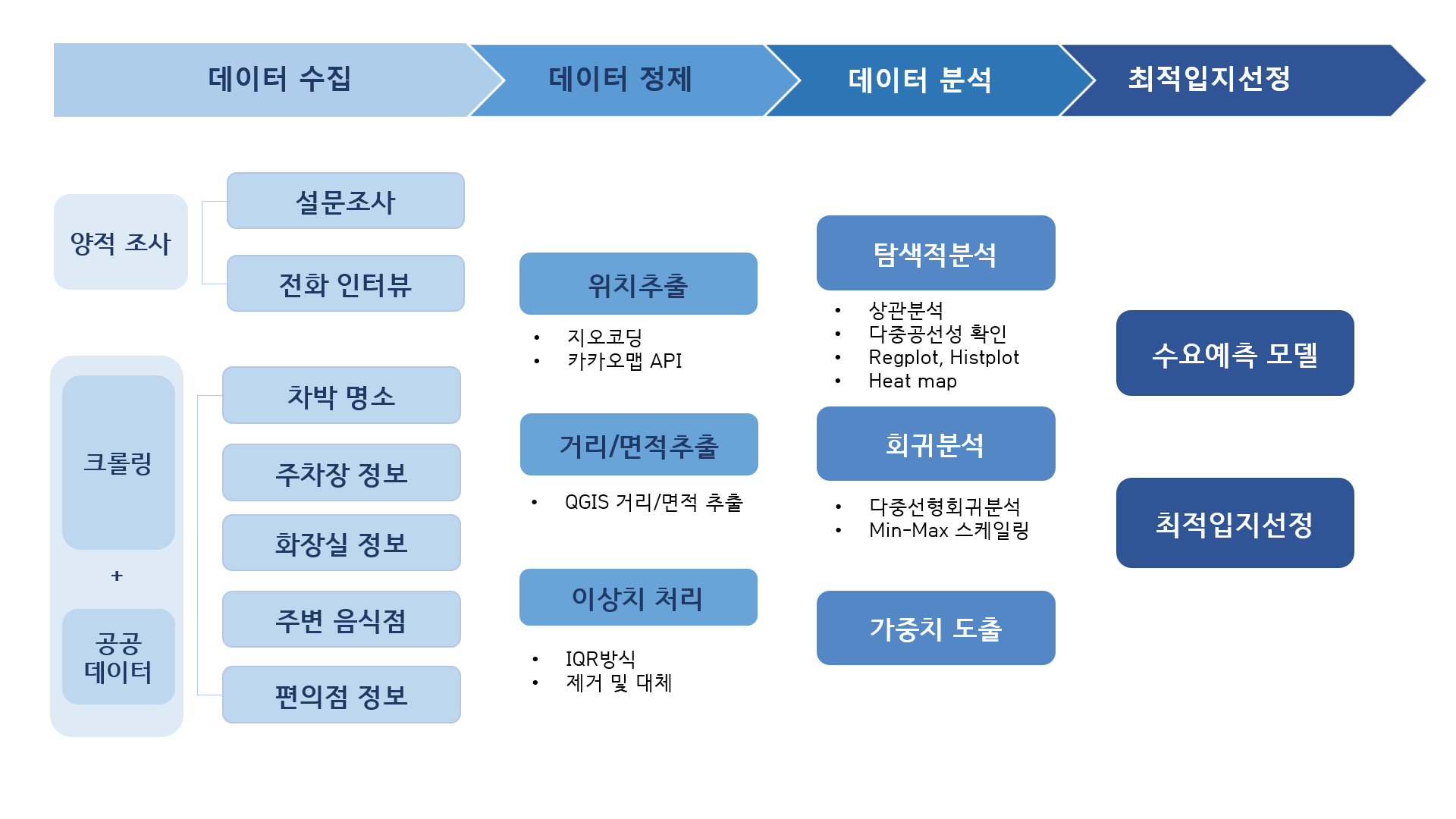


그림 19 화장실 거리 추정

1. **분석 프로세스**
   1. **분석 프로세스**
      1. 봉투 자판기 최적입지 전체 분석 프로세스



[그림 20] 분석 프로세스

* 봉투자판기 최적입지 선정을 위해 데이터 수집, 정제, 분석, 차박지 우선순위 도출 및 최적입지 선정 순으로 분석 실시.
* 차박지 수요(차박 쓰레기 발생량)에 영향을 미치는 변수를 구하기 위해 설문조사 실시.
* 분석에는 Python과 R, Qgis를 주로 이용하였으며, 가중치를 이용한 우선 입지 점수 계산을 위해 엑셀과 구글 스프레드시트 이용
  1. **분석 내용 및 방법**
* 분석을 통해 주요 차박지의 특성이 되는 요인과 차박지의 인기도(해당 지역의 언급 량)의 상관관계를 파악 후, 다중선형회귀모델을 활용하여 각 요인이 언급량에 얼마나 영향을 끼치는지 분석함. 회귀계수를 바탕으로 각 요인 별 가중치를 도출하고, 최종 자판기 입지 선정을 위한 종합 점수를 산출함. 예측모델을 통해 차박지의 수요를 예측하여 쓰레기 발생 가능 정도를 파악
  + 1. 변수 도출
       1. 사용한 변수
* ‘언급 수’를 설명할 수 있는 4가지 요인
  + 차박지의 주차 가능 면적
  + 인근 화장실까지의 최단 거리
  + 인근 편의점까지의 최단 거리
  + 인근 음식점 개수를 선정
    - 1. 변수 선정 이유
* 차박지의 ‘언급수’를 바탕으로, 차박 쓰레기 발생 가능성을 추정
  + 차박지별 쓰레기 발생량 집계 데이터는 존재하지 않기 때문에 ‘차박지 방문자 수가 많을수록 쓰레기 발생량이 많다’는 가정텍스트, 시계이(가) 표시된 사진

    자동 생성된 설명

그림 21 차박 요인 선정 지표

* + SNS, 카페, 블로그에는 차박지를 방문한 경험이 있는 사람들이 작성하는 게시글이 대부분이므로, ‘해당 지역에 대한 게시글이 많을수록 사람들이 많이 방문한다’고 가정
* 설문조사 결과와 카페/블로그 관련글을 바탕으로, 차박 지역의 ‘언급 빈도수’에 영향을 주는 변수 파악
  + 차박 커뮤니티 사용자 대상으로 실시한 설문조사 결과, 차박 경험이 있는 사람 중 차박 지역 선정시 우선적으로 고려하는 요인 중 1위는 61%를 차지한 ‘화장실 여부’ 이며, ‘주차장여부(10.3%)’, ‘인근 마트/편의점 여부(12.7%)’ 가 2, 3위에 해당

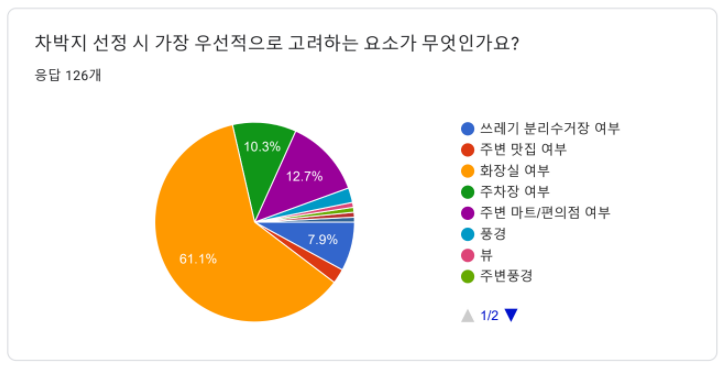


그림 22 구글폼 차박 관련 설문조사

* + 블로그/카페/SNS의 차박 관련 게시글을 수집, 분석하여 명사 추출한 결과, 주차/주차장/주차 공간 등 주차와 관련된 단어가 3262번으로, 추출한 총 16850개의 단어 중 11번째로 가장 많이 언급됨
  + 공중화장실/공공화장실 등 화장실과 관련된 단어는 3057번으로 12번째로 가장 많이 언급됨

테이블이(가) 표시된 사진

자동 생성된 설명

그림 23 차박언급 게시글 명사추출 내림차순 정렬

* 요인 별 가설 설정
  + ‘주차 면적’이 넓을수록, 주차하기 편하고 캠핑용품을 사용할 공간이 넉넉해지기 때문에 차박지로 인기가 많을 것(해당 지역이 차박 명소로 많이 언급될 것)
  + ‘화장실까지의 거리’가 가까울수록, 용변 해결이 용이하기 때문에 차박지로 인기가 많을 것
  + ‘편의점까지의 거리’가 가까울수록, 편의성이 좋기 때문에 차박지로 인기가 많을 것
  + 여행객들은 맛집을 찾아가는 것을 선호하기 때문에, 근처 음식점 수가 많을수록, 차박지로 인기가 많을 것
    1. 데이터 변환
       1. 데이터 분포

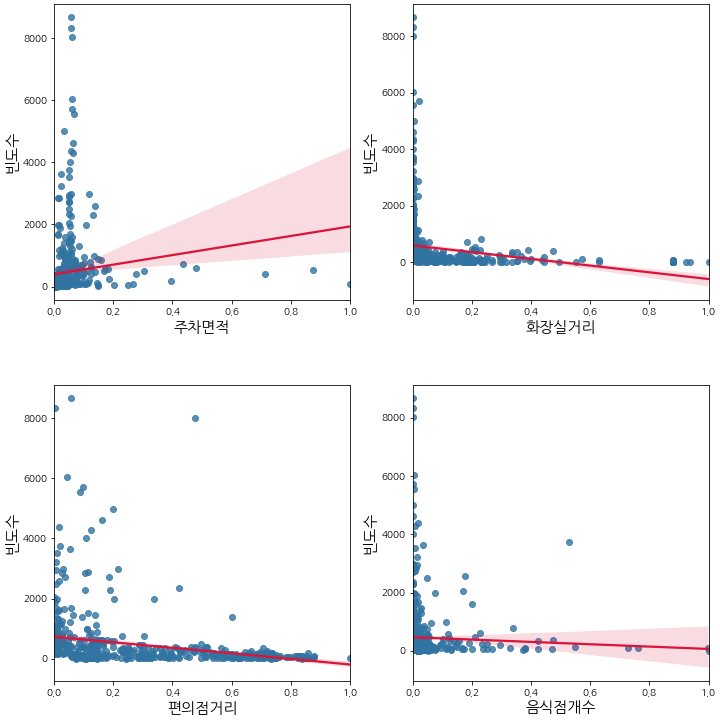


그림 24 전체 데이터 regplot 분포도

* 각 요인들의 분포가 정규분포에 가깝지 않고, 값들의 분포를 알 수 없음
  + - 1. 로그화

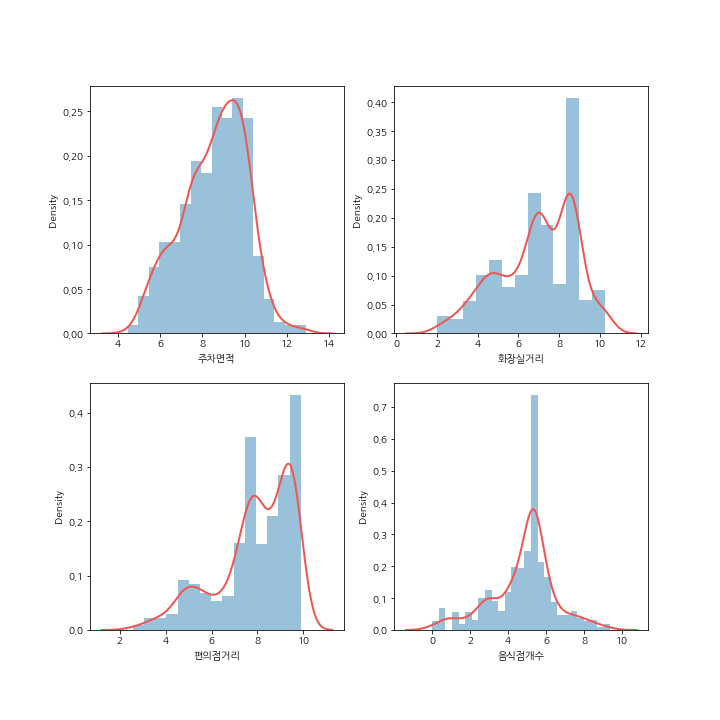


그림 25 전체데이터 log화 이후 분포 histplot

* 정규성을 높이고 회귀분석에서 정확한 값을 얻기 위해 log를 취함. 데이터 간 편차를 줄여 왜도(skewness)와 첨도(Kurtosis)를 줄일 수 있기 때문에 정규분포에 가까운 형태가 됨
  + - 1. 정규화
* 회귀모델에 쓸 설명변수의 분포를 Min-Max scaling을 통해 0에서 1 사이의 값으로 정규화 하여 회귀모델 작성

텍스트, 시계이(가) 표시된 사진

자동 생성된 설명

그림 26 MinMax scaling

* + 1. 회귀분석 (예측모델 도출)
       1. 선정한 요인과 차박지 언급횟수간 상관분석 실시
* 선정한 요인과 차박지 언급횟수 사이의 실제 연관성을 알아보기 위해 피어슨 상관계수분석 실시
* 차박지 언급 횟수와 선형 상관관계를 보이는 요인에 대해 p-value 확인 후, 0.05 이하인 요인들만 다중선형회귀모델의 독립변수로 결정
* Linear Regression을 이용해서 선형회귀분석 실시
  + - 1. 독립변수들의 다중공선성 확인
* 독립변수들 간의 상관정도가 높아 회귀 분석에서의 부정적인 영향을 미치는 현상을 다중공선성이라 함
* 다중공선성이 있을 경우, 부정확한 회귀 결과가 도출되므로 다중회귀모델에서 독립변수간 관계가 있는지 분석하는 요인 VIF(분산 팽창 요인)를 활용

텍스트, 시계이(가) 표시된 사진

자동 생성된 설명

그림 27 다중공선성 수식

* 일반적으로 두 변수의 VIF가 모두 10을 넘으면 다중공선성이 있는 것으로 파악됨.
  + 변수간 VIF 값 확인 후 회귀모델에 사용할 독립변수 결정
    - 1. 다중선형회귀 모델을 사용
* ‘차박지가 언급되는 빈도 수’에 대한 수식 도출

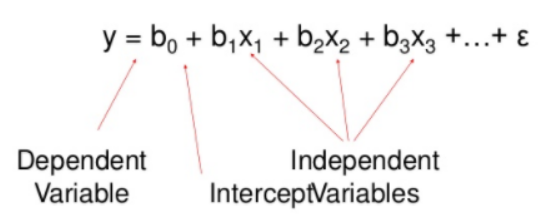


그림 28 다중회귀식

* + 1. 쓰레기 봉투 자판기의 최적입지 선정 모델 개발
       1. 다중선형회귀 분석을 통해 산출한 회귀 계수를 바탕으로 각 요인 별 가중치 부여

텍스트이(가) 표시된 사진

자동 생성된 설명

그림 29 개별 가중치 계산

* + - 1. 각 차박지 별 쓰레기 관련 언급 수를 요인으로 추가
* 뉴스 및 언론, SNS 데이터를 통해 차박지 별 ‘쓰레기 문제 언급 수’를 파악하여 요인으로 추가
* 해당 지역과 ‘쓰레기 문제’ 가 함께 언급되었다면, 개선 조치가 시급 할 것으로 판단하고 가중치 부여
* 가중치 값은 다른 요인들의 가중치들과 비슷한 수준(평균)의 2배로 결정
  + 뉴스에 해당 지역의 쓰레기 문제가 언급되었다면, 쓰레기봉투 자판기 설치가 시급한 것으로 반영

텍스트이(가) 표시된 사진

자동 생성된 설명

그림 30 쓰레기 문제 가중치 결정

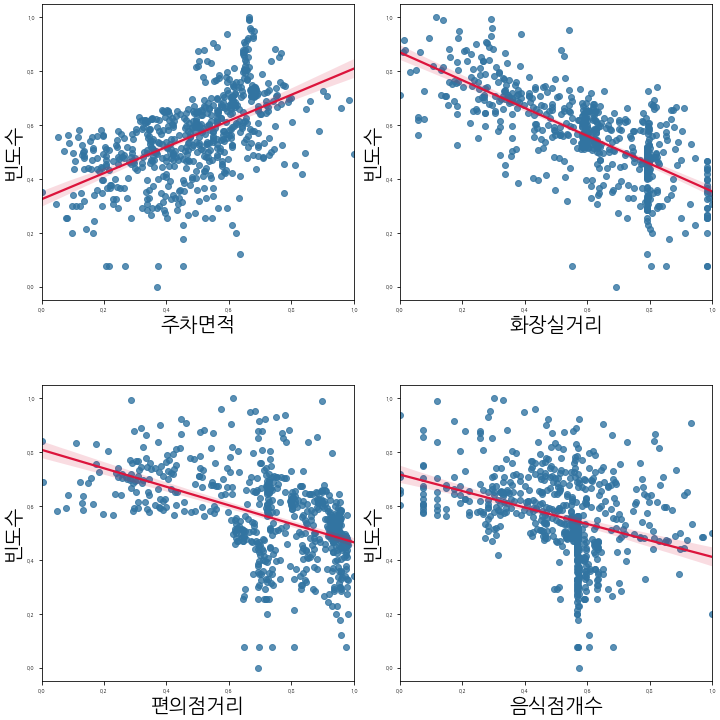
* + - 1. 가중치를 바탕으로 입지 우선순위 점수를 계산
* 요인별 MinMax정규화한 값 \* 해당 가중치 값을 합산하여 입지 우선순위를 결정하는 점수로 사용



그림 31 총 점 계산식

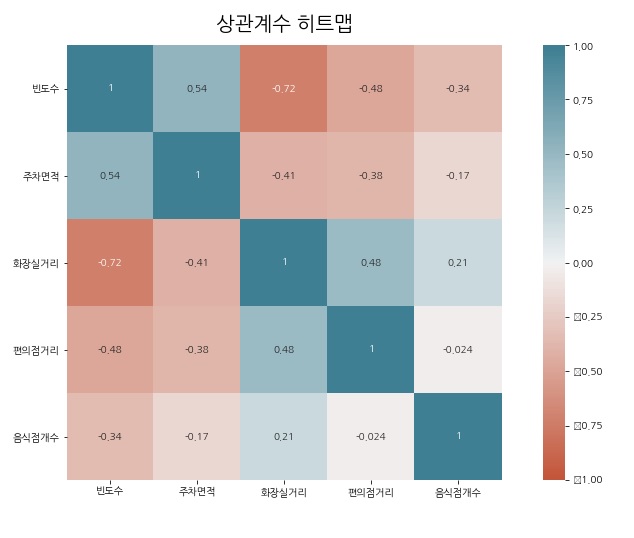
* + - 1. 산출한 점수를 바탕으로, 상위 50위 지역도출
      2. 상위 50위 지역 중, 사설 캠핑장 제외, 편의점거리가 먼 순서로 정렬
* 사설 캠핑장의 경우, 유료로 운영되어 이미 쓰레기 처리 시설이 존재하고, 쓰레기 처리용 봉투를 제공하는 경우가 대부분이기 때문에 쓰레기봉투 자판기 입지 우선순위에서 제외
* 편의점거리가 가까운 지역의 경우, 종량제봉투 혹은 분리수거용 봉투를 쉽게 구매할 수 있을 것이므로 설치 우선순위에서 제외
  + - 1. 최종적인 자판기 입지 우선순위 결정
  1. **다중선형회귀모델을 이용한, 차박지 수요 예측**
* 차박 명소의 후보지에 대해 조사한 값(주차가능면적, 화장실까지의 거리, 편의점까지의 거리, 인근 음식점 수)을 선형모델에 대입하여, 잠재적 차박지 수요를 예측
* 지자체에서 해당 차박지의 SNS 언급 수를 예측하여, 쓰레기 문제에 선제적으로 대응 가능

1. **봉투 자판기 최적입지 분석결과**
   1. **EDA 및 분석결과 시각화**
      1. 로그화 후 산포도 확인(regplot)



[그림 32] log후 regplot 그래프

* 데이터 분석을 위한 log변환과 scaling 이후 분포도를 보면 선형관계로 변환된 것을 볼 수 있음
* 주차 면적’ 변수는 X 값이 증가할수록 Y값이 증가하는 경향을 보이며 양의 상관관계를 띔
* ‘화장실거리’, ‘편의점 거리’, ‘음식점 개수’와 ‘빈도수’간의 그래프의 추세선를 통해, X값이 증가할수록 Y값이 감소하는 경향을 보이며 음의 상관관계를 띄는 것을 알 수 있음
* ‘음식점 개수’의 경우 회귀 곡선의 경향성으로부터 벗어난 점들이 비교적 많음.
  + 1. 요인 별 상관 분석
       1. Heatmap을 통한 전체적인 상관관계 파악



[그림 33] 상관계수 히트맵

* 화장실거리는 -0.72의 값으로 높은 상관관계를 보임
* 차박지 언급 빈도수와 주차면적은 다소 높은 상관관계(0.54)를 나타내는 것으로 파악
* 차박지 언급 빈도수와 화장실거리, 편의점 거리, 인근 음식점 개수는 음의 상관관계를 보임
* 편의점거리의 경우 -0.48의 다소 높은 상관관계를 보임
* 음식점 개수의 상관계수는 -0.34 로 비교적 약한 상관관계를 보임
* Pearson 상관계수와 p-value 도출 결과 모든 경우 p-value < 0.05로, 4개의 요인 모두 회귀모델의 설명변수로 사용하기로 결정함
  + - 1. VIF(분산 팽창 요인)을 통해 변수간 다중 공선성 확인

테이블이(가) 표시된 사진

자동 생성된 설명

[그림 34] VIF 다중공선성 확인

* 모든 VIF가 10미만으로 다중 공선성 문제가 없는 것으로 확인되어, 모든 변수를 요인으로 선정 가능
  1. **통계 분석 결과(~회귀분석)**
     1. 분석 결과
        1. 회귀분석 결과

텍스트이(가) 표시된 사진

자동 생성된 설명

[그림 35] 회귀분석 결정계수 결과

* 회귀 분석 결과 결정계수는 0.5471 로 도출됨
* 이는 약 54%의 정도의 설명력을 가짐

텍스트이(가) 표시된 사진

자동 생성된 설명

[그림 36] 모델 결정계수 , P-value 값 결과

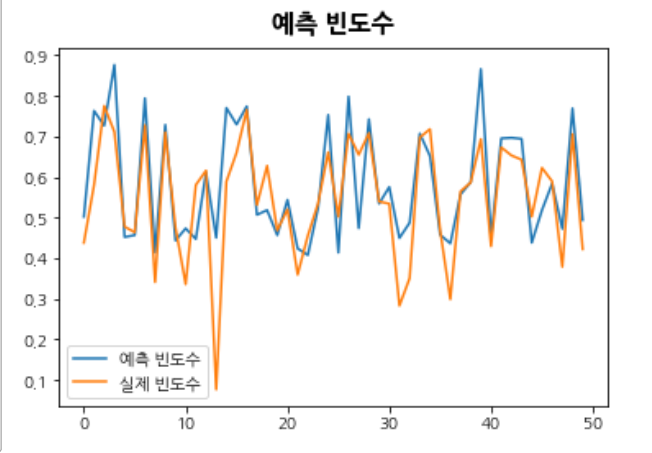
* p-value 값은 유의수준 모든 변수 모두 유의함

텍스트이(가) 표시된 사진

자동 생성된 설명

[그림 37] coef 회귀계수 결과

* 각 변수의 회귀 계수는 0.25, -0.39, -0.12, -0.16 값을 도출함
  + - 1. 예측 결과



[그림 38] 예측 결과 그래프

* 예측 모델은 총 624개의 데이터로 train 80% , test 20%로 진행
* 테스트셋의 x 독립변수 데이터 50개를 이용하여 빈도 수 예측
  + 1. 가중치 선정 결과

텍스트이(가) 표시된 사진

자동 생성된 설명

[그림 39] 변수별 가중치 결과

* 각 요인 별 가중치를 산출한 결과, 음식점 개수는 약 -0.19, 화장실거리는 -0.41, 편의점거리는 -0.12, 주차면적은 약 0.3, 쓰레기 언급은 0.5의 가중치를 부여
* 위 가중치를 활용하여 각 차박지 별 합산 점수 도출
  1. **최적입지분석 결과**
     1. 쓰레기 봉투 자판기 최적 입지 상위 50개 산출

합산 결과를 바탕으로 쓰레기 봉투 자판기 최적 입지 상위 50개 산출

텍스트이(가) 표시된 사진

자동 생성된 설명

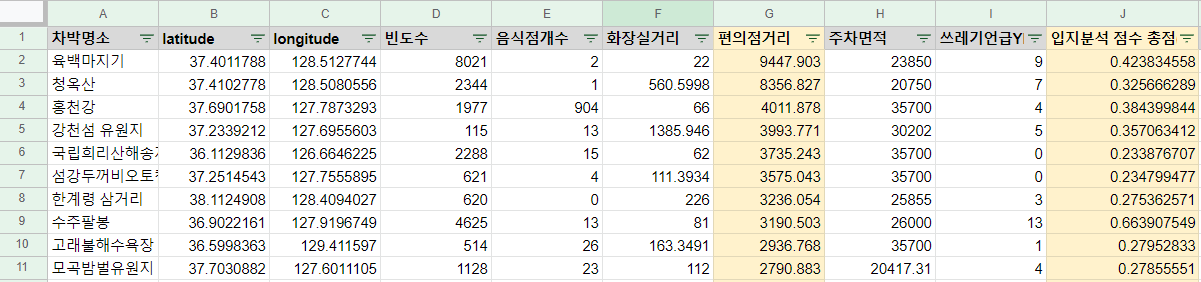
[그림 40] 최적입지분석 결과 상위 10위

텍스트, 지도, 나무이(가) 표시된 사진

자동 생성된 설명

[그림 41] 최적입지 결과 QGIS 지도 시각화

* 빨간색: 자판기 최적 입지 상위 50위
* 초록색: 전체 데이터 624개 차박지
  + 1. 편의점 거리가 먼 순으로 20개 장소 선정
* 선출된 50개 차박지 중 사설 캠핑장 제거 후 편의점 거리가 먼 순으로 내림차순 정렬



[그림 42] 최적입지 순위 사설 캠핑장 제거 후 편의점 내림차순

지도이(가) 표시된 사진

자동 생성된 설명

[그림 43] 편의점 거리 기준 상위 20위

* 쓰레기 봉투 자판기 설치 우선 입지 분석 결과
  + 1위: 육백마지기

2위: 청옥산

3위: 홍천강

* + 1. 지자체별 쓰레기봉투 자판기 설치 입지 제안(예시)
* 인구대비 면적이 넓고, 차박지가 많은 강원도, 충청도의 경우를 확인

지도이(가) 표시된 사진

자동 생성된 설명

그림 44 충청도 자판기 설치 제안 위치

* 충주 비내섬, 목계솔밭, 수주팔봉, 희리산 해송 자연휴양림에 쓰레기봉투 자판기를 우선 도입하도록 제안

지도이(가) 표시된 사진

자동 생성된 설명

그림 45 강원도 자판기 설치 제안 위치

* 청옥산, 안반데기, 섬강두꺼비오토캠핑장, 모곡밤벌유원지, 삼거리, 기타 해변에 쓰레기봉투 자판기를 우선 도입하도록 제안
  1. **회귀분석 결과를 바탕으로 잠재적 차박지 수요 예측**
     1. 회귀모델을 사용하여, 새로운 차박지 수요 예측

텍스트이(가) 표시된 사진

자동 생성된 설명

그림 46 빈도수 예측 결과

* 예측모델에 임의로 선정한 차박지에 대한 데이터를 넣어 빈도수 예측 (가평 백팔 유원지, 청령포, 고려산 농원, 진남 교반)
* 예측 빈도수를 바탕으로 쓰레기 문제 발생 가능성 예측 및 대비

1. **활용 방안**
   1. **문제점 개선 방안**
      1. 문제점

* 코로나로 인해 차박이라는 단어가 급부상한지 얼마되지 않아 차박 관련 데이터 부족, 이로 인해 크롤링한 데이터를 바탕으로 분석
* 예측 모델 학습 시 충분하지 못한 데이터(624개)로 분석하여 정확도 (54.71%)가 높지 않음
* 노지의 경우, 정확한 주차공간면적 파악 및 자판기 설치 위치 선정의 어려움 존재
* 쓰레기 봉투 자판기 설치 시 예산의 한계(설치 비용)와 자판기 관리 문제 발생
  + 1. 개선 방안
* 빈도수는 유동인구데이터로 대체하고 주차장 정보, 화장실 정보, 차박지별 쓰레기 배출량 등 정확한 데이터를 각 지자체에서 받아 분석한다면, 더욱 신뢰성이 높은 분석 가능
* 더 많은 데이터를 수집한다면 모델의 정확도를 조금 더 높일 수 있을 것으로 예상
* 노지의 경우, 자판기 설치 위치 선정이 어려우므로 근처 공중화장실에 쓰레기 봉투 자판기 설치
* 쓰레기 봉투 자판기를 설치함으로써 줄인 쓰레기 처리 인력과 비용을 자판기 설치/관리에 투입
  1. **업무 활용 방안**
     1. 기대효과
* 현재 차박지 중 쓰레기 불법투기 문제가 심각한 곳에 쓰레기봉투 자판기를 우선 설치함으로써 쓰레기 무단투기 문제 개선 기대
* 쓰레기 무단투기 문제로 인한 하천 오염, 해양 오염 등 환경오염 문제 개선
* 쓰레기 봉투 자판기 도입을 통해 쓰레기 무단투기를 줄임으로써 효율적인 쓰레기 처리/단속 인력 배치 및 비용 절감 가능
* 차박지 수요를 예측하여 쓰레기 문제에 선제적으로 대응 가능
* 차박지 폐쇄가 아닌 쓰레기 문제 개선을 통해 지역 관광 활성화 및 지역경제 활성화 도모
  + 1. 활용방안
* 차박지 수요 예측 모델을 활용하여 지역 관광지 개발 시 사용 가능
* 지역별 쓰레기 무단 투기 문제 해결을 위한 대책 마련 시 제언 가능
* 수요 예측 모델을 활용하여 쓰레기 무단투기 증가에 대한 사전 대응 가능
* 차박지 외에도 쓰레기 문제가 심각한 곳(대학가, 원룸 촌)에 쓰레기 봉투 자판기 활용 가능

1. **참고자료**

[1] 이삭, “차박 금지’ 대신 ‘총량제’ 선택한 충주시 ”, 경향신문, 2021년 07월 29일

[2] 최지호, “차박이 뜨자 바다가 운다” , MBC 뉴스데스크 , 2021년 05월 19일

[3] 박명기, “양양군내 불법 차박 뿌리 뽑는다.” , 전국매일신문 , 2021년 04월 08일

[4] 김동이 , “마을회관, 항포구에도 차박...도 넘은 캠핑족 어쩌나 “ , 오마이뉴스, 2021년 03월 12일자

[5] 김원경, “울산, 캠핑족 몰리는 강동해변 ‘쓰레기 전쟁’” , 울산제일일보 , 2021년 05월 30일

[6] MBN 디지털 뉴스부, “무의도, '차박족'에 몸살…"쓰레기, 해도 해도 너무해"” , MBN , 2021년 05월 02일

[7] 경남이슈투데이, “일운면, 국토대청소·쓰레기 되가져가기 캠페인” , 2021년 07월 01일

[8] 박민주 , “캠핑이 발견한 자동차의 부캐...‘차박’ 검색 2,300% 급증”, 서울경제, 2020년 09월 08일

[9] MBC 뉴스데스크 , ”차박·캠핑 환영하지만…쓰레기는 갖고 가세요“ , 2021년 07월 17일

[10] 성연제, “[길따라 멋따라] '차박 성지'의 수난…쓰레기는 어찌합니까” , 연합뉴스 , 2020년 07월 04일

[11] 신하림, “‘머문 자리 아름답게’ 캠핑 에티켓 직접 알리는 주민들” , 강원일보, 2021년 08월 03일

[12] 이현경 , “코로나19 이후 SNS에 '차박' 언급량 2배 이상 증가” , 뉴스핌, 2020년 11월 13일

[13] 강민선, “야영객·낚시꾼, 쓰레기 투척에...방파제 인근 '아수라장' , 세계일보, 2021년 08월 01일

[14] 최은지, “ 코로나19 여파에 캠핑객 늘어난 인천 섬…쓰레기에 몸살”, 연합뉴스, 2021년 04월 25일

[15] 정태겸, “부릉부릉캠프, 차박캠퍼 대상 쓰레기 처리 설문 결과 공개”, 캠프파이브,2021년08월09일

[16] 정동원, “화장실서 샤워하고 쓰레기 버리고…오죽하면 폐쇄”, SBS뉴스, 2021년08월10일

[17] 김영준, ” 강변 유원지 차박 유행…쓰레기 투기 몸살”, KBS뉴스, 2021년07월18일

[18]박재천, ”충주 수주팔봉 '차박 총량제' 도입…하루 120대만 허용”, 연합뉴스, 2021년07월23일

[19] 새올전자민원창구 , gwanakgu.eminwon.seoul.kr

[20] 서울시 민원정보 시스템 , democracy.seoul.go.kr

[21] 네이버 카페 차박은 내친구, https://cafe.naver.com/gpsf

[22] 네이버 카페 차박캠핑클럽 https://cafe.naver.com/chcamping

[23] 한국언론재단, 빅카인즈 https://www.bigkinds.or.kr/

[24] 한국어텍스트 전처리 관련자료 딥러닝을 이용한 자연어 처리 입문 (wikidocs.net)

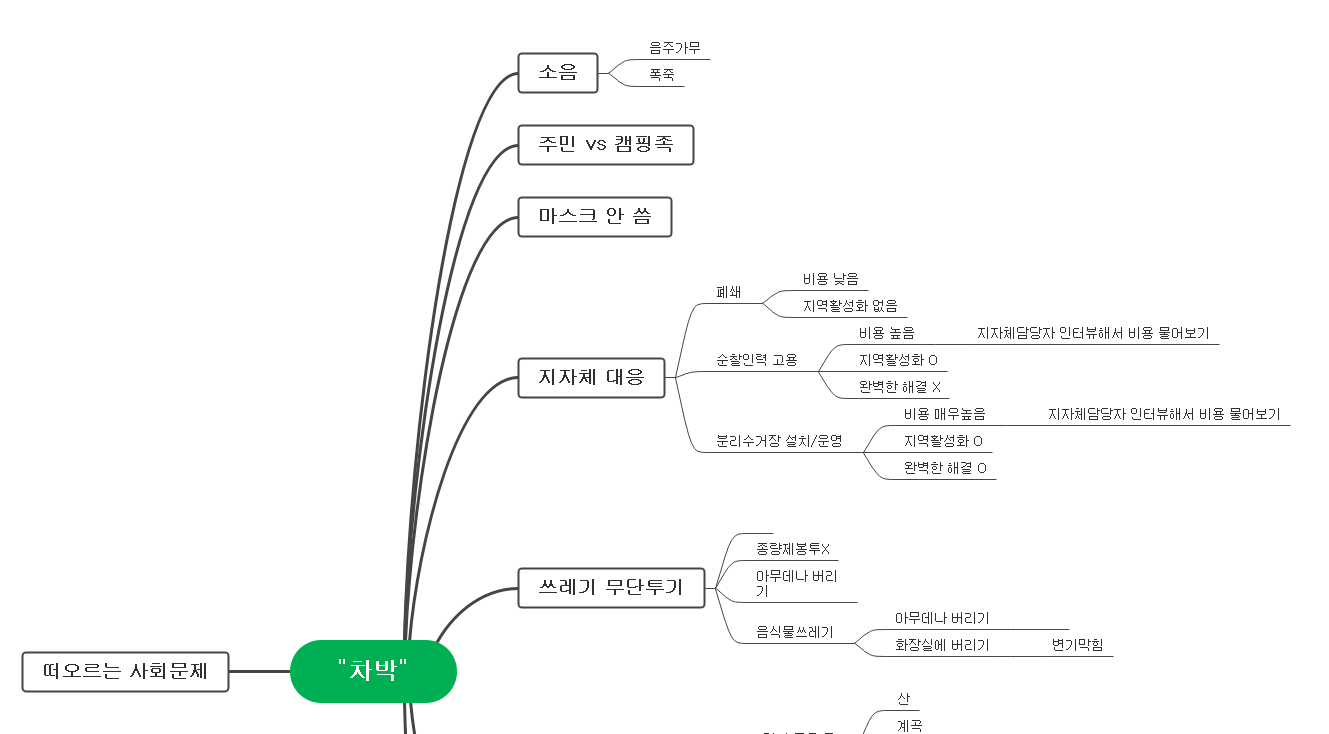
[25] 선형회귀 및 다중공선성 관련 참고자료 Regression(02) - 다중선형회귀 및 다중공선성 | DataLatte's IT Blog (heung-bae-lee.github.io)

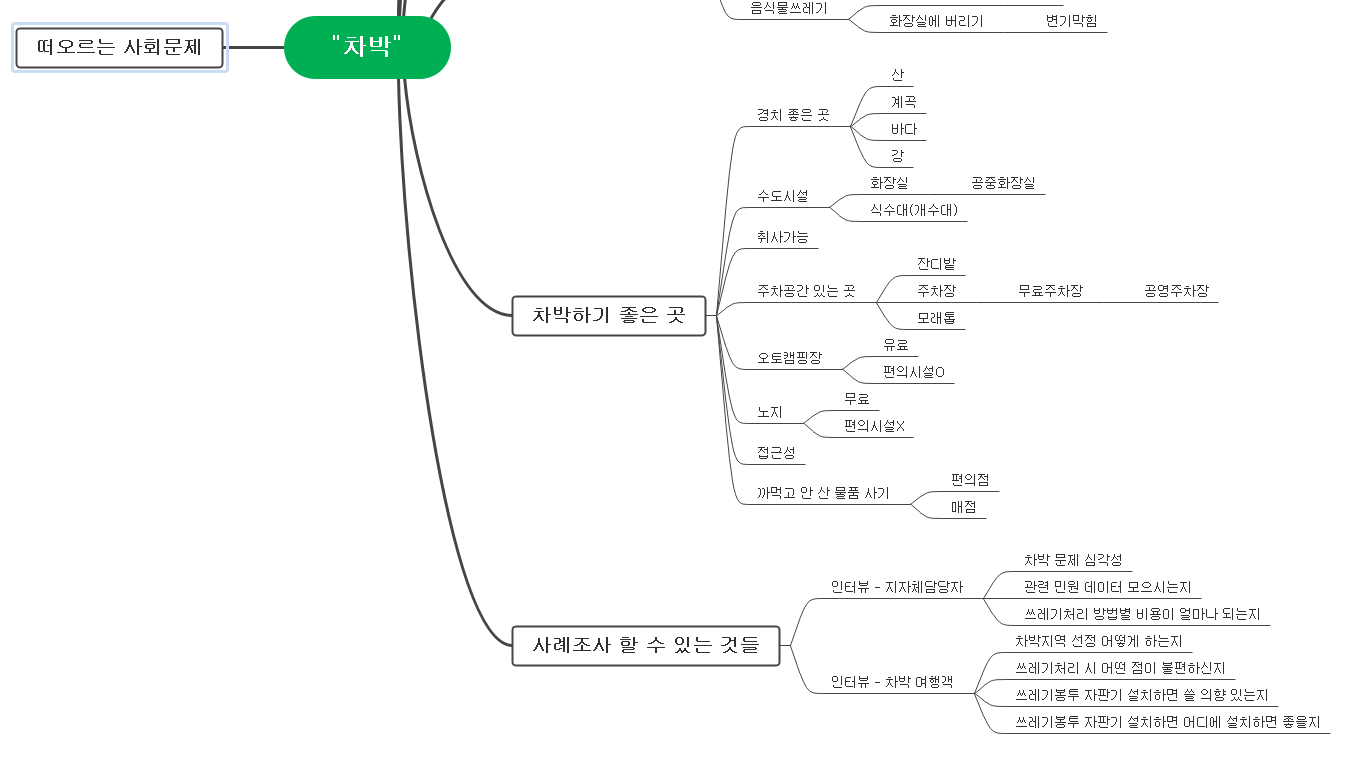
[26] 다중공선성(Multicollinearity)과 VIF(Variance Inflation Factors) 참조 (tistory.com)

[27] 데이터 로그화 https://evening-ds.tistory.com/31

[28] 회귀분석 로그변환/ 회귀분석을 할 때 로그 변환을 하는 이유 (tistory.com)

1. **부록**
   1. **마인드맵**





* 1. **코드 첨부**
     1. **텍스트 크롤링 데이터 정제 후, 명사 추출 코드(일부)**
* SNS, 블로그, 카페에서 ‘제목’, ‘본문’ 등 텍스트 형태의 비정형 코드를 크롤링 후, 명사를 분리하여 차박 명소의 목록을 추출하는 데 사용

from konlpy.tag import Okt  
  
okt=Okt()  
  
nouns\_list=[]  
 for i in range(len(sns)):  
 try:  
 nouns\_list.extend(list(okt.nouns(sns.loc[i]['content'])))  
 except:  
 continue  
  
len(nouns\_list)  
  
#명사 리스트에 '주차장' 있는지 확인  
'주차장' in nouns\_list  
  
from collections import Counter  
count=dict(Counter(nouns\_list).most\_common())  
  
len(count)  
  
count.keys()  
  
#한글자 제거  
key\_list=list(count.keys())  
for key in key\_list:   
 if len(key)<=1:  
 count.pop(key)  
len(count)  
  
df=pd.DataFrame(columns=['단어','count'])  
df['단어']=list(count.keys())  
df['count']=list(count.values())  
  
df=df[df['count']>1]  
len(df)  
  
df.to\_csv('/content/drive/MyDrive/빅데이터\_팀플/본문에서명사추출.csv',encoding='euc-kr')

**7.2.3 카카오맵 API 크롤링 코드(일부)**

* 화장실, 편의점의 좌표데이터를 검색 후 결과를 크롤링하는 코드.

|  |
| --- |
| # 차박 지역별로 ‘지역명’+’키워드’를 검색한 결과의 x, y(경도,위도) 좌표 추출.  url = 'https://dapi.kakao.com/v2/local/search/keyword.json'  for i in range(len(location)):  print(location\_keyword[i], '에 대해 검색 시작.')  # 좌표  x = float(location[i][1])  y = float(location[i][2])  headers = {'Authorization' : "카카오 Developers 사용자 키값"}  page\_num = 1    params = {'query':location\_keyword[i], # 검색어  'page':page\_num, # 페이지번호(필요없을거같지만 일단 남겨둠)  'rect': f'{start\_x},{start\_y},{end\_x},{end\_y}'} # 검색좌표 범위    resp = requests.get(url, params=params, headers=headers) # 카카오api 에서 검색결과 받아오기  search\_count = resp.json()['meta']['total\_count'] # 검색결과 개수  if search\_count == 0:  print(' 검색결과 수가 0입니다.')  for j in range(len(resp.json()['documents'])): # 한 검색결과 안에서 갯수만큼 반복  results.append([location\_keyword[i], # 장소+키워드  resp.json()['documents'][0]['place\_name'], # 검색결과명  resp.json()['documents'][j]['y'], # y좌표(위도)  resp.json()['documents'][j]['x']] # x좌표(경도)  # 결과 저장하기.  data = pd.DataFrame(results, columns=['장소명+키워드명', '이름', 'y좌표', 'x좌표'])  data.to\_csv(savefile\_name, index=False, encoding='utf-8-sig')  print('저장까지 완료.') # PermissionDenied 에러 뜨면, 해당 파일을 열어놨는지 확인.  data.head(10) |

**7.2.4인스타그램 크롤링 코드(일부)**

1. 7 def get\_content(url\_list):
2. location\_infos = []
3. location\_hrefs = []
4. date\_titles = []
5. main\_texts = []
6. cnt=0
7. for url in url\_list:
8. time.sleep(5)
9. driver.get(url)
10. #날짜
11. try:
12. date\_object = driver.find\_element\_by\_css\_selector(date\_object\_css)
13. date\_title = date\_object.get\_attribute("title")
14. except:
15. date\_title = None
16. date\_titles.append(date\_title)
18. #본문내용
19. try:
20. main\_text\_object = driver.find\_element\_by\_css\_selector(main\_text\_object\_css)
21. main\_text = main\_text\_object.text
22. main\_text=re.sub(r'[^ ㄱ-ㅣ가-힣A-Za-z]', '',main\_text)
23. except:
24. main\_text = None
25. main\_texts.append(main\_text)
27. #위치
28. try:
29. location\_object = driver.find\_element\_by\_css\_selector(location\_object\_css)
30. location\_info = location\_object.text
31. location\_href = location\_object.get\_attribute("href")
32. location\_info=re.sub(r'[^ ㄱ-ㅣ가-힣A-Za-z]', '',location\_info)
33. location\_href=re.sub(r'[^ ㄱ-ㅣ가-힣A-Za-z]', '',location\_href)
34. except:
35. location\_info = None
36. location\_href = None
37. location\_infos.append(location\_info)
38. location\_hrefs.append(location\_href)
40. #파일에 쓰기
41. f = open('insta\_write3.csv','a', newline='')
42. wr = csv.writer(f)
43. wr.writerow([date\_title,main\_text, location\_info,location\_href])
44. f.close()
45. cnt=cnt+1
47. if(cnt%100 ==0):
48. print(cnt,end=' ')
49. print('save success!')

52. #혹시몰라 df로 마지막에 저장
53. df=pd.DataFrame(columns=['날짜','본문','위치정보','위치url'])
54. df['날짜']=date\_titles
55. df['본문']=main\_texts
56. df['위치정보']=location\_infos
57. df['위치url']=location\_hrefs
59. return df
60. # #### temp에 넣을 링크리스트의 index 설정해야함!
61. # In[11]:
62. temp=link\_list[1000:2313]
63. len(temp)
64. # In[77]:
65. text\_df=get\_content(temp)
66. # In[ ]:
67. text\_df
68. # In[ ]:
69. text\_df.to\_csv('insta\_write2\_again.csv',index=False)
70. # In[ ]:
71. # In[ ]:
72. f=open('write.csv','w',newline='')
73. wr = csv.writer(f)
74. wr.writerow([3,'뇸', '뇸뇸'])
75. f.close()
76. # In[ ]:
77. f = open('insta\_write.csv','a', newline='')
78. wr = csv.writer(f)

**7.2.5‘쓰레기’ 언급량 관련 정제 코드**

1. content\_df=pd.read\_csv('/content/drive/MyDrive/빅데이터\_팀플/블로그,카페,인스타/sns\_content\_data최종.csv')  
   content\_df.head(5)  
     
   lst=['쓰레기','쓰 레기',' 쓰레기','쓰레기 ']  
     
   # 쓰레기있는 행 추출  
   content\_list=[]  
   for i in range(len(content\_df)):  
    for word in lst:  
    if word in str(content\_df.iloc[i][1]):  
    content\_list.append(content\_df.iloc[i][1])  
    break  
   len(content\_list)  
     
   #차박명소 불러오기  
   명소=pd.read\_csv('/content/drive/MyDrive/빅데이터\_팀플/차박명소\_\_빈도수word(New).csv')  
   명소list=list(명소['camp'])  
     
   len(명소list)  
     
   import kss  
   total=[]  
   for i in range(len(content\_list)):  
    temp=[]  
    for sent in kss.split\_sentences(content\_list[i]):  
    for word in lst:  
    if word in sent and sent not in temp:  
    temp.append(sent)  
    for loc in 명소list:  
    if loc in sent:  
    if sent not in temp:  
    temp.append(sent)  
    total.append(temp)  
     
   total  
     
   for i in range(0,len(total),1):  
    print(i)  
    line=''  
    for sen in total[i]:  
    line=line+' '+str(sen)  
    line.strip()  
    flag=0 #명소가 없으면 flag=0유지  
    for 명소 in 명소list:  
    if 명소 in line:  
    flag=1  
    break  
    if flag==0:#명소가 없다면  
    total.remove(total[i])  
     
   len(total)  
     
   word=[]  
   for i in range(0,len(total),1):  
    print(i,end=' ')  
    line=''  
    for sen in total[i]:  
    line=line+' '+str(sen)  
    line.strip()  
    flag=0 #명소가 없으면 flag=0유지  
    for 명소 in 명소list:  
    if 명소 in line:  
    if 명소 not in word:  
    word.append(명소)  
    flag=1  
    break  
    # if flag==0:#명소가 없다면  
    # total.remove(total[i])  
     
   len(total)  
     
   from collections import Counter  
   쓰레기언급장소=list(Counter(word).keys())  
     
   쓰레기언급장소  
     
   명소df=pd.read\_csv('/content/drive/MyDrive/빅데이터팀플/차박명소빈도수word(New).csv')  
   명소df  
     
   명소df.insert(1,'trash',['0']\*len(명소df))  
   명소df  
     
   for i in range(len(명소df)):  
    if 명소df.iloc[i][0] in 쓰레기언급장소:  
    print(명소df.iloc[i][0])  
    명소df.loc[i]['trash']='1'  
     
   명소df.to\_csv('쓰레기장소표시.csv')  
     
   명소df['trash'].unique()  
     
   f=open('/content/drive/MyDrive/빅데이터팀플/차박지\_쓰레기관련.txt',mode='w')  
   f.writelines(str(total))  
   f.close()  
     
   line=''  
   for sen in total[1]:  
    line=line+' '+str(sen)  
   line.strip()  
   print(line)  
   flag=0 #명소가 없으면 flag=0유지  
   for 명소 in 명소list:  
    if 명소 in line:  
    flag=1  
    print(명소)  
    break  
   if flag==0:#명소가 없다면  
    print('없다!')  
     
   for i in range(50):  
    for el in total[i]:  
    print(el)  
    print('-------------------------------------------------------------------------------------------')

**7.2.6 카페 크롤링 코드**

|  |
| --- |
| "driver.get('https://cafe.naver.com/chcamping/ArticleList.nhn?search.clubid=27087488&amp;search.menuid=12&amp;search.boardtype=L&amp;search.totalCount=11101&amp;search.cafeId=27087488&amp;search.page=733')\n",  "import pandas as pd\n",  "import json, re\n",  "post\_list = []\n",  "page=2\n",  "for j in range(1,1000):\n",  " try:\n",  " driver.switch\_to\_frame(\"cafe\_main\")\n",  " elem = driver.find\_elements(By.CLASS\_NAME, \"article\")\n",  " for e in elem:\n",  " post\_list.append(e.get\_attribute(\"href\"))\n",  " driver.find\_element\_by\_xpath(f\"\"\"//\*[@id=\"main-area\"]/div[6]/a[{page}]\"\"\").click()\n",  " if j == 10:\n",  " page = 3\n",  " \n",  " if page == 12:\n",  " page = 3\n",  " \n",  " else :\n",  " page += 1\n",  " driver.switch\_to\_default\_content()\n",  " except:\n",  " print(\"error\")\n",  " pass\n"  "start = 0\n",  "\n",  "f = open(\"naver\_cafe\_week.txt\", \"w\", encoding=\"utf-8\")\n",  "\n",  "for url in post\_list[start:end]:\n",  " print(str(start) + \"/\" + str(len(post\_list)), end=\"\\r\")\n",  " start += 1\n",  " try:\n",  " driver.get(url)\n",  " time.sleep(2)\n",  " driver.switch\_to\_frame(driver.find\_element(By.NAME, \"cafe\_main\"))\n",  " \n",  " title = driver.find\_element(By.CLASS\_NAME, \"title\_text\").text.strip()\n",  " board\_type = driver.find\_element(By.CLASS\_NAME, \"link\_board\").text.strip()\n",  " datetime = driver.find\_element(By.CLASS\_NAME, \"date\").text\n",  " contents = driver.find\_element(By.CLASS\_NAME, \"article\_viewer\").text.replace(\"\\n\", \" \")\n",  " f.write(\"POSTING\" + \"\\t\" + datetime + \\\n",  " \"\\t\" + board\_type + \"\\t\" + title + \"\\t\" + contents + \"\\n\") \n",  "\n",  " reply\_list = driver.find\_elements(By.CLASS\_NAME, \"comment\_box\")\n",  " for reply in reply\_list:\n",  " comment = reply.find\_element(By.CLASS\_NAME, \"text\_comment\").text.replace(\"\\n\", \" \")\n",  " user\_nik = reply.find\_element(By.CLASS\_NAME, \"comment\_nickname\").text\n",  " reply\_date = reply.find\_element(By.CLASS\_NAME, \"comment\_info\_date\").text\n",  " f.write(\"COMMENT\" + \"\\t\" + user\_nik + \"\\t\" + reply\_date + \\\n",  " \"\\t\" + board\_type + \"\\t\" + \"None\" + \"\\t\" + comment + \"\\n\")\n",  " driver.switch\_to\_default\_content()\n",  " time.sleep(2)\n",  " except:\n",  " driver.switch\_to\_default\_content()\n",  " time.sleep(2)\n",  " continue\n",  "f.close()" |

**7.6현황파악 – 참조기사**

**텍스트이(가) 표시된 사진

자동 생성된 설명**