네카라쿠배 데이터 사이언스 1기

야 너두! 카페 창업 할 수 있어

서울시 **카페 상권분석** 결과 예측

머신러닝 4조

김수민

장현영

정원준

목차
INDEX

STEP 1
프로젝트 배경

카페 창업 앗 뜨거워.. 코로나 와중에 15% 늘어

0lhyewon님 외 **410명**이 좋아합니다

c.won 🤍 카페 오픈했다 🤍
놀러와 얘드라 😦👍

STEP 1
프로젝트 주제 및 목표

카페 창업 입지 선정, 커피베이 상권 분석 시스템 주목

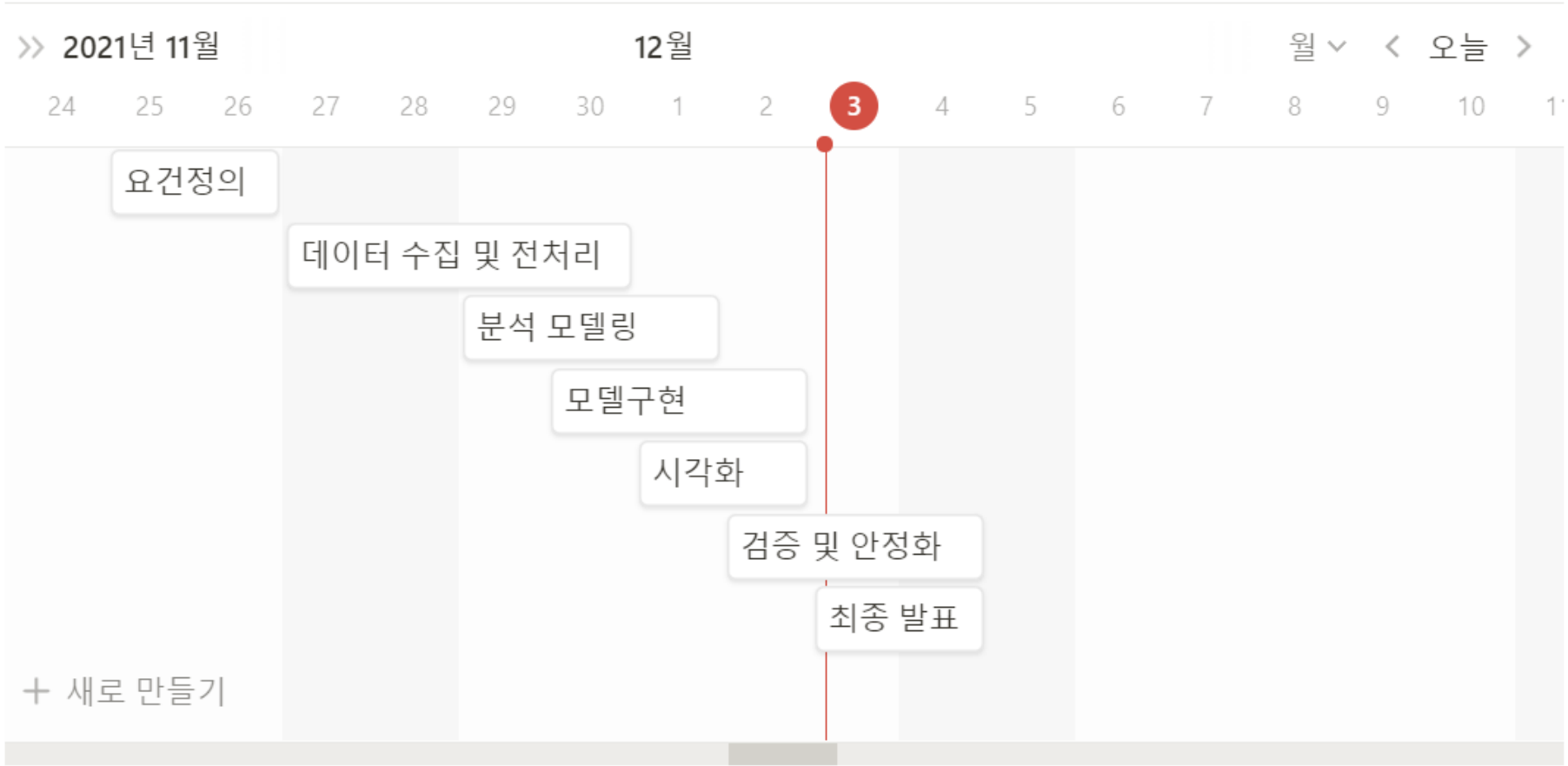
단기간에 분석이 어려운 상권 특성 파악 및 교통상황과 경쟁업체, 주거 특성 등
매출에 영향을 줄 수 있는 요소들을 종합적으로 고려해 최적의 입지를 선정

카페 예비 창업자에게 창업 시 도움이 될 수 있는 insight 제공

현재, 대형 프렌차이즈에서 **상권 분석 시스템**을 제공하고 있으나, 일반 개인 카페를 창업하
고자 하는 예비 창업자들은 상권 입지에 대한 정보를 얻기 쉽지 않습니다.

실제 창업을 위해 현장조사 시 중요하게 참고해야 할 사항들 제시함으로써 **개인 카페 예비
창업자**들을 또한 자금력에 맞는 최적의 점포를 찾을 수 있도록 도움을 제공합니다.

STEP 1
일정 계획


2021년 11월
12월
월
오늘
24 25 26 27 28 29 30 1 2 3 4 5 6 7 8 9 10
요건정의
데이터 수집 및 전처리
분석 모델링
모델구현
시각화
검증 및 안정화
최종 발표
새로 만들기

STEP 2
데이터 수집 및 출처

활용 데이터

주변 상권 변수

주변 음식점 수
주변 프렌차이즈 카페 수
별식/퓨전 요리 업종 매출액
양식 업종 점포 수
일식 업종 점포 수
패스트푸드점 점포 수

서울시 열린 데이터 광장
공공데이터 포털

지역 특성 변수

임대시세
공시지가
월 평균 매출액
유동인구, 직장인구, 주거인구
가구 수, 소득분위
주차장 수, 지하철 수
집객시설(은행, 병원, 학교, 버스 정류장 등)

스마트 치안 빅데이터 플랫폼

인스타그램 태그

#서울커피맛집
#서울카페추천
#서울감성카페
#카페스타그램
.
.
.

인스타그램 크롤링

STEP 2
Data Cleaning

1. 카페 데이터만 추출

```
sale = []
for i in tqdm([2014, 2015, 2016, 2017, 2018, 2019, 2020]):
    tmp = pd.read_csv('./data/서울시 우리마을가게 상권분석서비스(상권-추정매출)_{}.csv'.format(i), encoding='cp949')
    sale.append(tmp)
sale = pd.concat(sale)
```

```
100%|██████████| 7/7 [00:07<00:00,  1.08s/it]
```

```
sale.shape
```

```
(1091003, 80)
```

```
# 서비스 업종에서 '커피-음료'만 추출
coffee_sale = sale[sale['서비스_업종_코드_명']=='커피-음료']
```

```
len(coffee_sale['상권_코드_명'].unique())
```

```
1329
```

```
print(coffee_sale.shape)
coffee_sale.head(2)
```

```
(34201, 80)
```

STEP 2
Data Cleaning

2. 모든 데이터 '**행정동**' 단위로 **맵핑**

	상권_구분_코드	상권_구분_코드_명	상권_코드	상권_코드_명	x좌표 값	y좌표 값	시군구_코드	행정동_코드	기준_년월_코드	geometry
0	R	전통시장	1001453	낙성대시장	196121	442084	11620	11620585	201810	POLYGON ((196213.760 442152.080, 196186.890 44...
1	R	전통시장	1001454	봉천제일종합시장	195147	442413	11620	11620595	201810	POLYGON ((195242.520 442426.730, 195236.250 44...

```
code = pd.read_csv('./data/행정동코드.csv', index_col=0)
code.head(2)
```

통계청행정동코드	행자부행정동코드	시도명	시군구명	행정동명
1101053	11110530	서울	종로구	사직동
1101054	11110540	서울	종로구	삼청동

```
coffee_sale = pd.merge(coffee_sale, area[['상권_코드', '상권_구분_코드_명', '상권_코드_명', '행정동_코드']], how='left', on=['상권_코드
```

```
coffee_sale[coffee_sale['행정동_코드'].isnull()]['상권_코드'].unique()
```

```
array([1001252], dtype=int64)
```

```
coffee_sale[coffee_sale['행정동_코드'].isnull()]['상권_코드_명'].unique()
```

```
array(['서울 성북구 미아삼거리역_2'], dtype=object)
```

```
area[area['상권_코드'] == 1001252]
```

	상권_구분_코드	상권_구분_코드_명	상권_코드	상권_코드_명	x좌표 값	y좌표 값	시군구_코드	행정동_코드	기준_년월_코드	geometry
1227	D	발달상권	1001252	서울 강북구 미아삼거리역_2	202544	457073	11290	11290685	201810	MULTIPOLYGON (((202667.740 456954.570, 202675....

◀ 카페 관련 정보를 가지고 있는 데이터는
행정동 단위가 아닌 상권 단위로 나뉘어 있기 때문에
'상권_코드 ' 에 해당하는 '**행정동** ' 으로 맵핑 작업

두 데이터를 합친 결과에서 행정동코드가 null값인 지역은 상권 코드명이 일치하지 않기 때문에 null값으로 처리됨

'상권_코드'는 '1001252'로 일치하고 미아삼거리역을 검색했을 때 강북구로 나오기 때문에 coffee_sale의 '상권_코드_명'을 성북구에서 강북구로 변경해주고 행정동 코드(11290685)를 추가해줌

STEP 2
Data Cleaning

3. 결측치는 최대한 찾아서 채워 줌

```
cafe_all[cafe_all['행정동명'].isnull()]
```

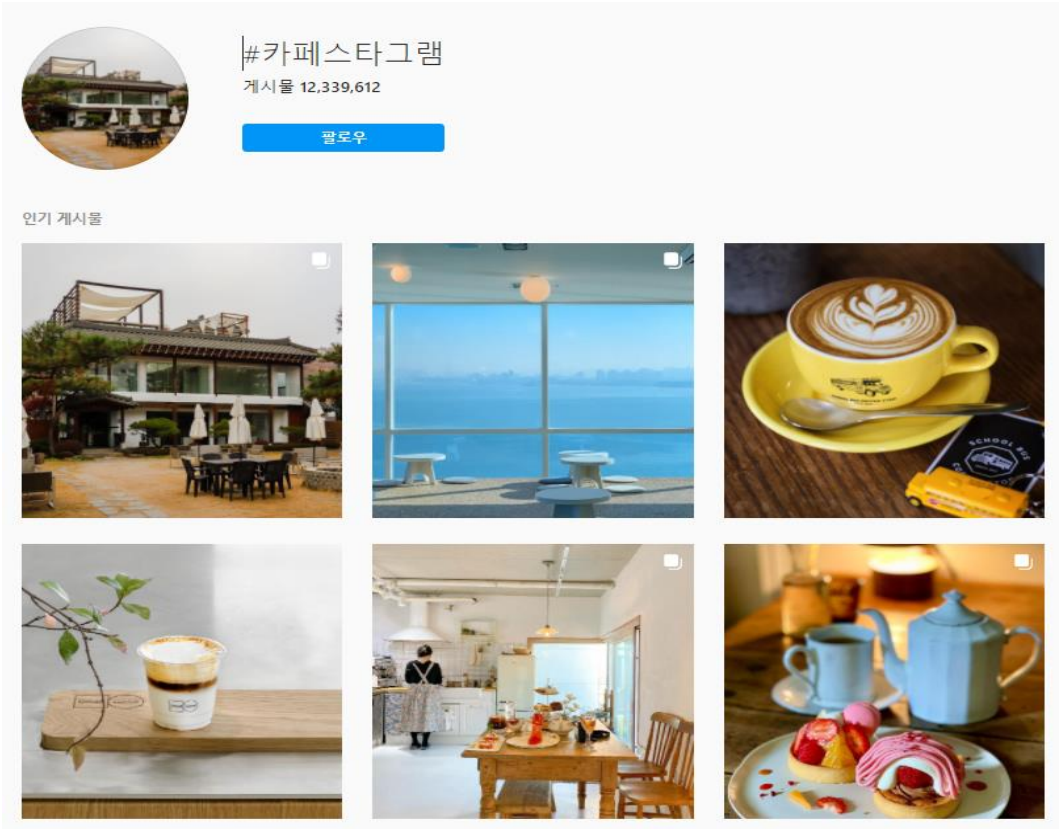
	년도	업종명	허가신고일	업소명	소재지도로명	소재지지번	영업장면적(㎡)	행정동명	폐업일자	업태명
11605	2016	휴게음식점	20160201	후아나	서울특별시 노원구 석계로1길 18, 301호 (월계동)	NaN	NaN	NaN	NaN	커피숍
11592	2016	휴게음식점	20161107	커피온리(only)	서울특별시 노원구 공릉로 221, 1층 (공릉동)	NaN	NaN	NaN	NaN	커피숍

▲

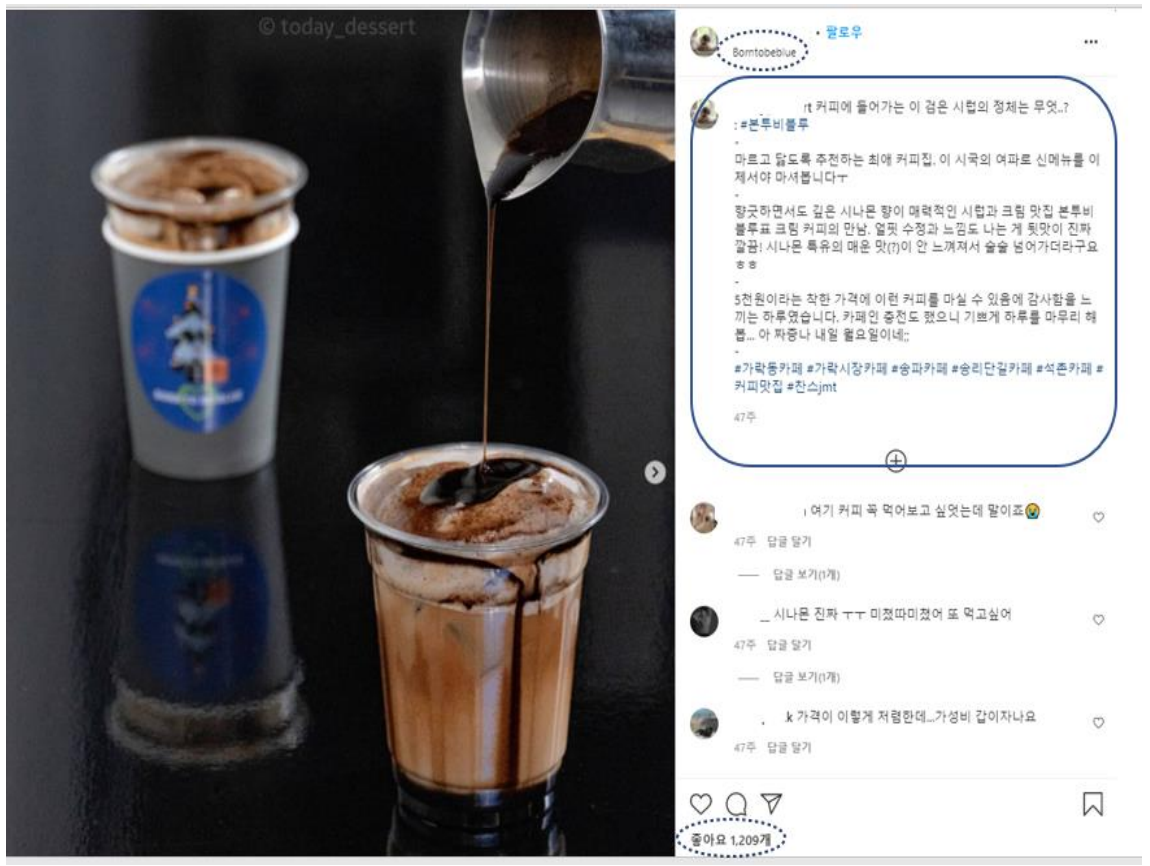
결측치에 대해서 최대한 채울 수 있는 부분은 직접 찾아서 채워 줌

STEP 2
Data Integration

STEP 3
인스타그램 크롤링



위치, 게시글, #태그, 좋아요, 게시일



서울시 카페 크롤링을 위해 12개 키워드 해시 태그로 검색
이 중 위치가 포함되어 있는 정보글 200개씩 수집
(ex 카페스타그램, 서울커피맛집,서울핫플카페 등)

서울시 내의 카페 관련 키워드 분석을 위해 게시글, 좋아요 개수,
해시 태그 전체 크롤링 진행

STEP 3
크롤링 데이터 수집

	place	like	content	tags	date
0	오츠커피 마포 - oats coffee Mapo	147	여기 짱...2분컷...✅	[]	2021-05-23
1	태양커피	203	태양커피는 외부디저트 반입되니깐 좋다니 ...	['#아인슈페너맛집', '#태양커피', '#유튜래...	2020-05-14
2	리사르커피	178	요기 느무 핫하다~ 호로록 정신이 번쩍 오네로소 꿀맛..❤️#에스프레소바 #리사르커...	['#에스프레소바', '#리사르커피', '#유튜랭_약수']	2020-04-24
3	프릳츠 커피 컴퍼니 - Fritz Coffee Company	170	설레는입구	[]	2020-06-04
4	리사르커피	161	그라니따가 짱 맛있는 날씨😈#에스프레소바 #리사르커피 #유튜랭 약수	['#에스프레소바', '#리사르커피', '#유튜랭_약수']	2020-05-21

```
<class 'pandas.core.frame.Da
Int64Index: 2221 entries, 0
Data columns (total 5 columns
 #   Column   Non-Null Count
---  ------   --------------
 0   place    1632 non-null
 1   like     2221 non-null
 2   content  2221 non-null
 3   tags     2221 non-null
 4   date     2221 non-null
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1045 entries, 0 to 1390
Data columns (total 7 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   place    1045 non-null   object
 1   like     1045 non-null   int64
 2   content  1045 non-null   object
 3   tags     1045 non-null   object
 4   date     1045 non-null   object
 5   위도       1045 non-null   float64
 6   경도       1045 non-null   float64
```

```python
def get_contents(driver):
    html = driver.page_source
    soup = BeautifulSoup(html, 'html.parser')
    # 위치(없으면 None)
    time.sleep(5)
    try:
        place = soup.select('div.M90cS')[0].text
        print('place :', place)
    except:
        return None

    # 좋아요
    try:
        like = soup.select('a.zV_Ni')[0].text[4:-1]
        if int(like) <= 10:
            return None # 10개 이하로 광고로 판단된 경우 None (100하 None--광고 가능성이 큼)
    except:
        return None # 좋아요가 없거나 좋아요 수를 숨긴 경우 None
    time.sleep(3)
    # 본문 content
    try:
        content = soup.select('div.C4VMK > span')[0].text
    except:
        content = ' '
    # 해시 태그
    time.sleep(5)
    tags = re.findall(r'#[^\s#,\w]+', content)

    # 작성 일자
    date = soup.select('time._1o9PC.Nzb55')[0]['datetime'][:10]

    data = [place, like, content, tags, date]
    time.sleep(5)
    return data
```

```python
driver.get(insta_url('서울감성카페'))
time.sleep(3)
select_first(driver)
cnt = 0
error_cnt = 0
for i in tqdm(range(500)):
    data = get_contents(driver)
    try:
        if data is None:
            first_move(driver) if i == 0 else move_next(driver)
            continue
        res.append(data)
        first_move(driver) if i == 0 else move_next(driver)
        cnt += 1
        print(f'{cnt}개')
        if cnt == 200:
            break
    except:
        error_cnt += 1
        if error_cnt > 5:
            print('last', i)
            break
        time.sleep(3)
        first_move(driver) if i == 0 else move_next(driver)
```

```
  0%|                                                  | 0/500 [00:00<?, ?it/s]
place :
  0%|▏                                                 | 1/500 [00:10<1:30:46, 10.91s/it]
place : ilikemountain
  0%|▎                                                 | 2/500 [00:21<1:29:57, 10.84s/it]
```

총 수집데이터 2400개 중 place 결측 치, 중복 제거, 광고로 의심되는 태그(#광고, 좋아요 10개 이하 포함), 위치태그가 불분명한 경우 (google 검색 시 나오지 않는 데이터) 제외 후 1045개

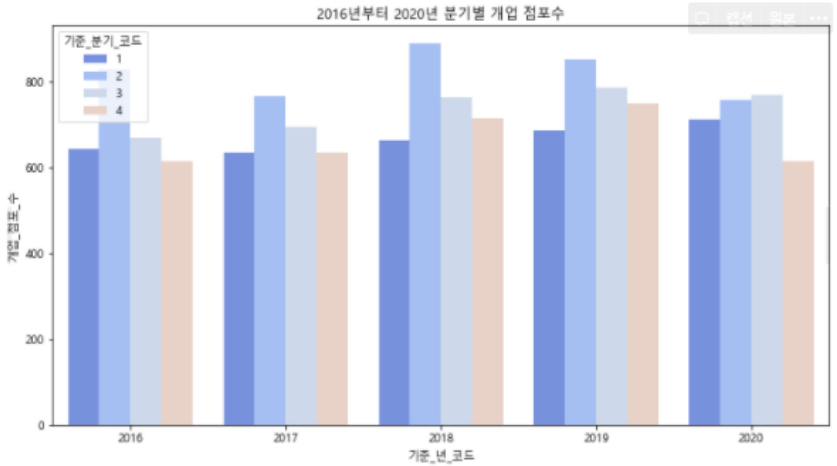
서울시 내의 카페 관련 키워드 분석을 위해 게시글, 좋아요 개수, 해시 태그 전체 크롤링 진행

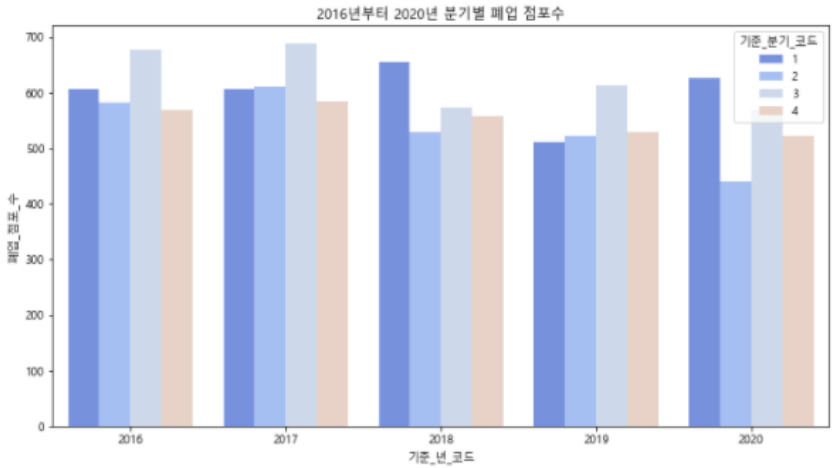
STEP 3
워드클라우드(wordcloud)

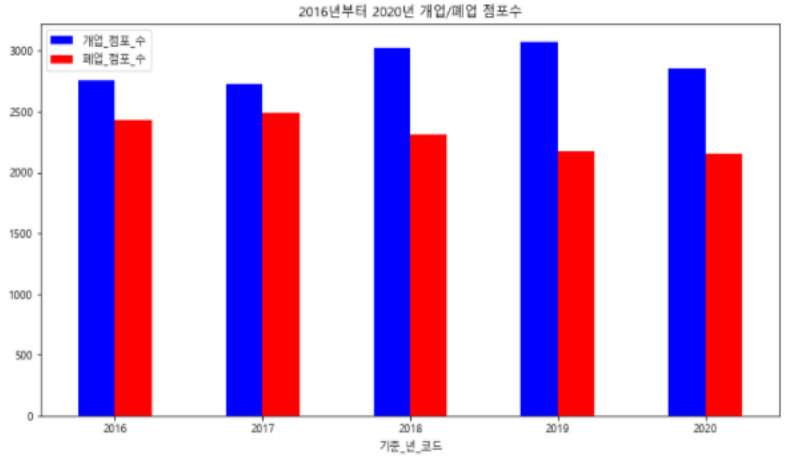
Konlpy 게시글 content 내용 명사 추출, 언급단어 상위 200개
- 커피, 추천, 맛집, 크림, 공간, 분위기, 등 카페 평가 관련 단어위주

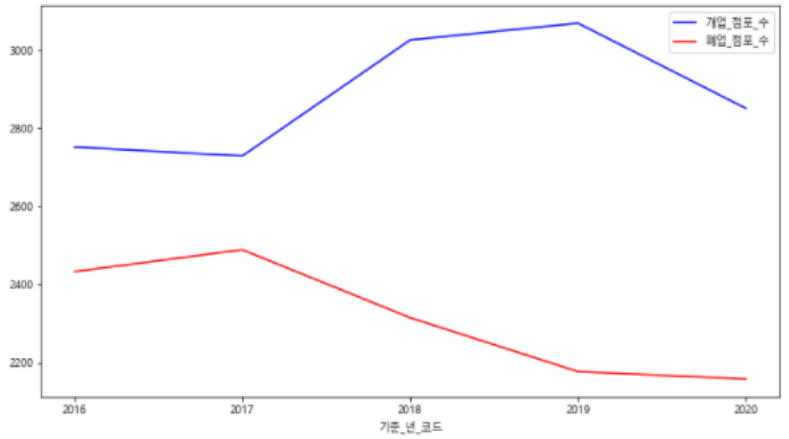
전체 태그 내용 중 가장 언급단어 상위 200개
- 카페 관련 태그 중 맞팔, f4f, 좋반 등 정보 공유 계정의 tag들이 많음

STEP 3
EDA









- 개업시기는 대체적으로 2분기(봄-여름)
- 폐업시기는 매우 고른 분포를 나타냄

- 최근 3년 사이에 개업 점포 수 증가 후 감소세
- 코로나로 인해 작년에 감소폭을 보였으나, 꾸준히 폐업 대비 개업 점포 수가 늘어나고 있다는 것을 확인

STEP 3

Folium 지도 시각화 및 타겟 변수 설명

카페 창업에 적합한 지역 설정을 표현하는 항목 – 이익, 폐업률

이익 : 매출액 – 평당 임대료 * 지역별 평균 매장면적
폐업률: 폐업 점포 수 / (점포 수 + 폐업 점포 수) * 100

임대료를 제외한 비용은 서울 지역에 상관없이 비슷할 것으로 판단하여 이익지표를 매출, 임대료,면적으로 계산

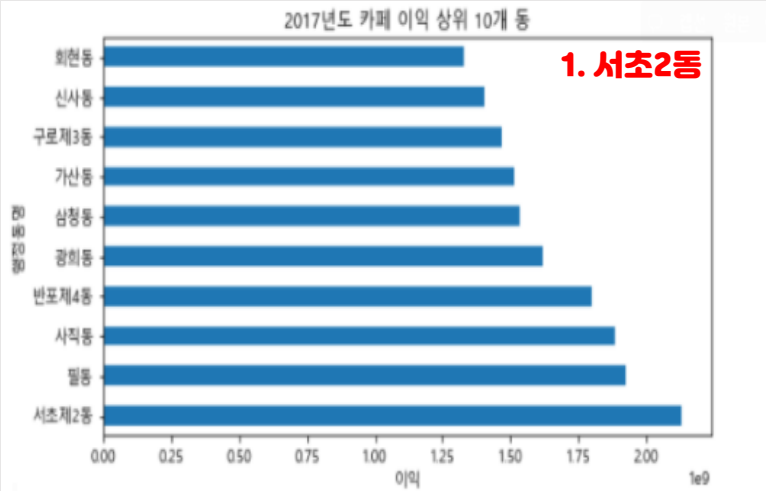
행정동별 매출액(2020)을 folium으로 시각화한 것

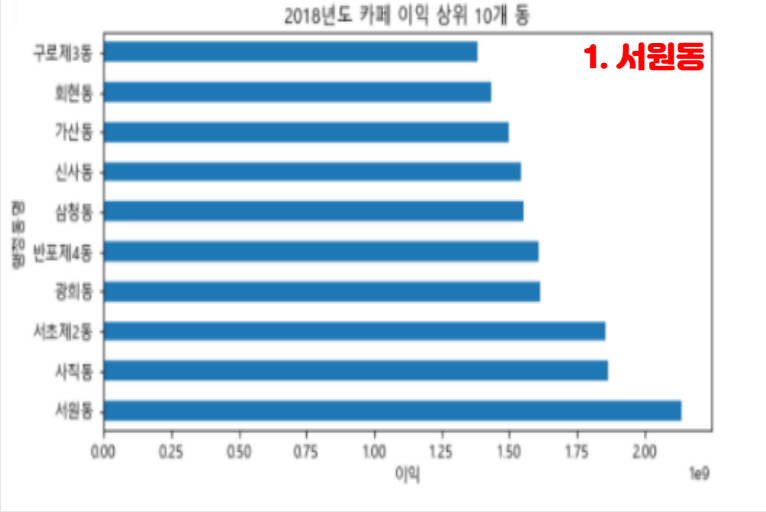
폐업률은 해당 지역의 경쟁 정도, 이익 등의 정보가 포함된 지표로 고려

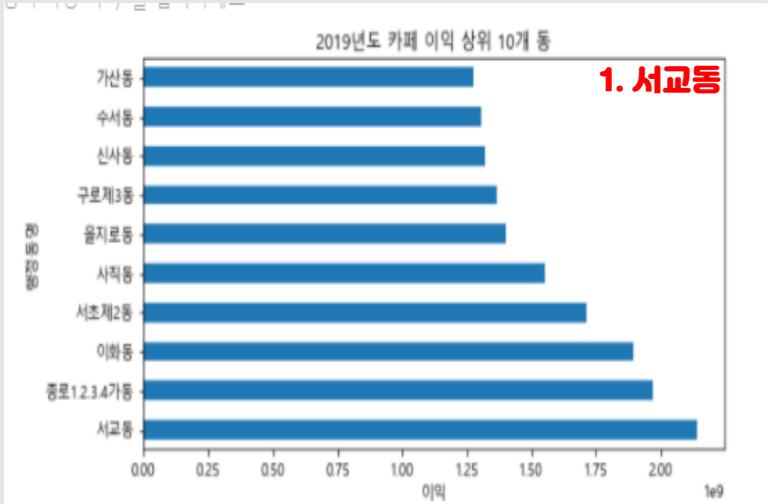
행정동별 폐업률(2020)을 folium으로 시각화한 것

STEP 3

EDA – 이익 상위 10개동







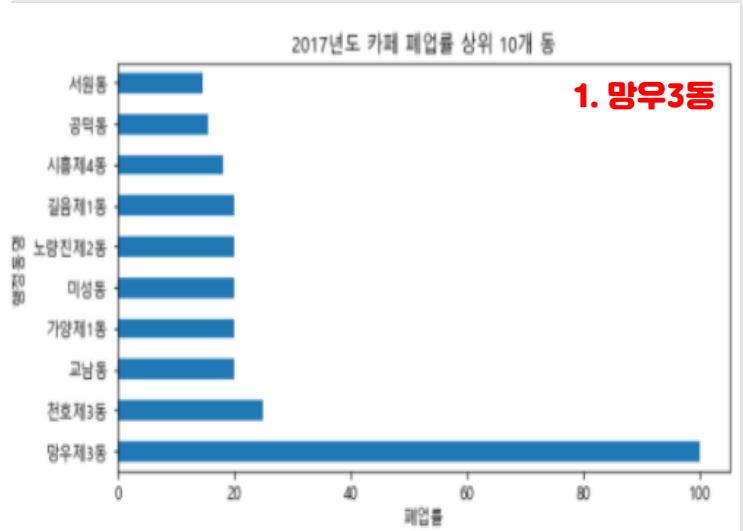
- **연도별 이익**

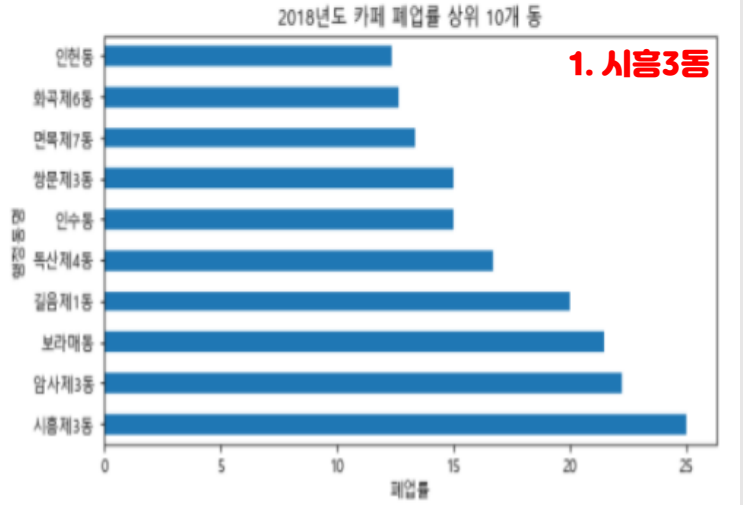
2017년 서초2동, 필동, 사직동 등 직장, 기업이 밀집한 지역이 이익이 가장 크게 나타났는데, 2019년 이후 서교동 등 비교적 젊은 세대가 많이 이용하는 지역의 이익이 크게 발생

절대적인 트렌드는 아니나 직장 상업지구 뿐만 아나라 SNS광고나 카페를 찾는 연령대도 더욱 다양해지는 만큼 트렌드 분석이 필요함

STEP 3

EDA – 폐업률 상위 10개동





- **연도별 폐업률**

2017년 망우3동, 천호3동 등 주거시설 밀집 지역 중심으로 폐업률이 높은 것으로 파악

이후 높은 폐업률을 보이는 시흥3동, 고덕2동 화곡4동, 보라매동 역시 직장과 연결된 배후시설이 낮은 편이고 주거시설이 밀집된 지역이 많음

STEP 3

타겟 변수에 영향이 있는 feature selection

총 수집한 독립변수 특성이 될 수 있는 column의 개수(174개)가 많았기 때문에 특정 데이터의 종류가 많은 경우 가장 중요하다고 생각되는 feature를 선택해서 모델링을 진행

1. SelectKBest

Target 변수와 그 외 변수 사이의 상관관계를 계산해 가장 상관관계가 높은 변수 k개를 사용자가 선정하는 모듈
F -regressor를 이용해 score를 선정 (1차 선정 20-30개)

```
from sklearn.feature_selection import f_regression, SelectKBest
## selotor 정의,
selector = SelectKBest(score_func=f_regression, k=20)
## 학습데이터에 fit_transform
X_train_selected = selector.fit_transform(X_train, y_train)
## 테스트 데이터는 transform
X_test_selected = selector.transform(X_test)

all_names = X_train.columns
## selector.get_support()
selected_mask = selector.get_support()
## 선택된 특성(변수)들
selected_names = all_names[selected_mask]
## 선택되지 않은 특성(변수)들
unselected_names = all_names[~selected_mask]
print('Selected names: ', selected_names)
print('Unselected names: ', unselected_names)
```

```
Selected names:  Index(['치킨전문점_개수', '여성_생활인구_수', '연령대_10_생활인구_
수', '연령대_40_생활인구_수',
       '연령대_50_생활인구_수', '연령대_60_이상_생활인구_수', '월요일_생활인구_수',
'화요일_생활인구_수',
       '월요일_매출_비율', '화요일_매출_비율', '수요일_매출_비율', '목요일_매출_비
```

폐업률	이익
['치킨전문점_개수', '총_생활인구_수', '남성_생활인구_수', '여성_생활인구_수', '연령대_40_생활인구_수', ...(중략)...'연령대_60_이상_매출_비율', '관공서_수', '초등학교_수', '고등학교_수', '대학교_수', '극장_수', '숙박_시설_수'] feature 30개 선정	['점포_수', '유사_업종_점포_수', '프랜차이즈_점포_수', '분기당_매출_금액', '분기당_매출_건수', '주중_매출_금액',...(중략)... ' 남성연령대_50_직장_인구_수 ' , ' 여성연령대_20_직장_인구_수 ' , ' 여성연령대_30_직장_인구_수] feature 30개 선정

STEP 3

타겟 변수에 영향이 있는 feature selection

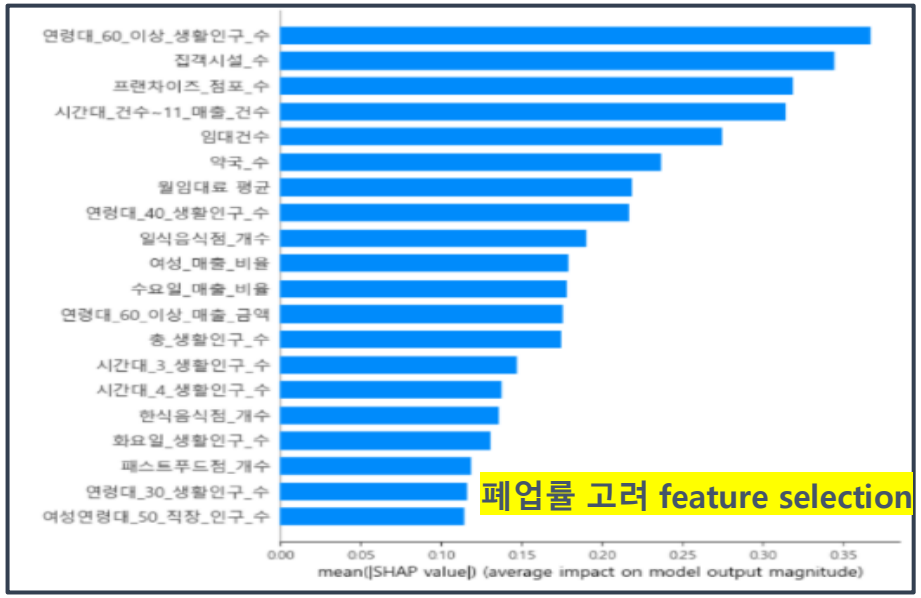
총 수집한 독립변수 특성이 될 수 있는 column의 개수(174개)가 많았기 때문에 특정 데이터의 종류가 많은 경우 가장 중요하다고 생각되는 feature를 선택해서 모델링을 진행

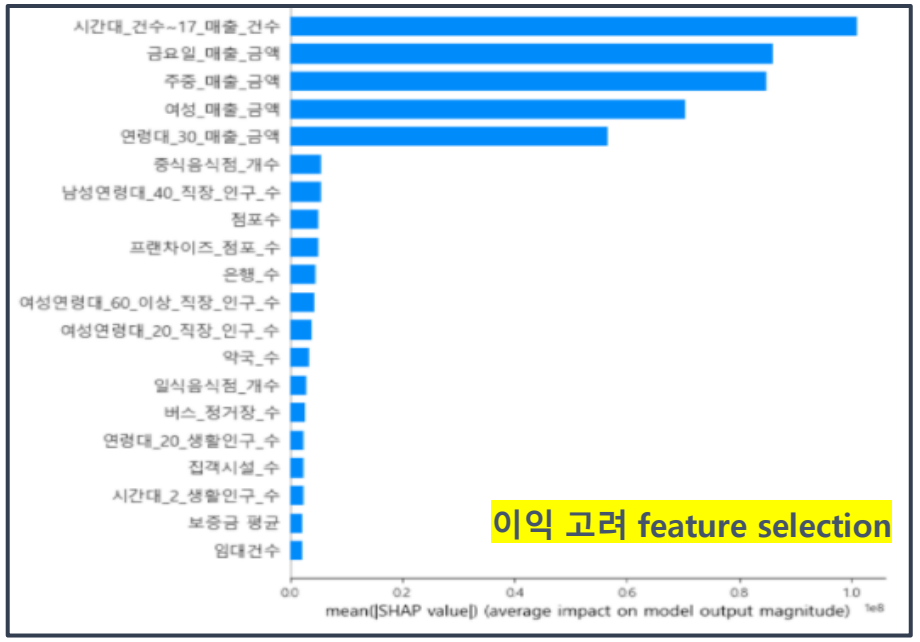
2. Shap Value

Sampling된 다른 feature들의 값에 따라 특정 feature의 margin contribution이 달라지고, 이를 반복 수행하여 weight 산출함

Weight으로 해당 feature의 중요도 판단

폐업률	이익
[연령대 60이상 생활인구 수, 집객 시설 수, 프랜차이즈 점포 수, 시간대 건수 ~11 매출건수...(중략)... 패스트푸드점 개수 연령대 30 생활인구 수 여성 연령대 50 직장인구 수] feature 20개 선정	[시간대건수~17_매출 건수, 금요일 매출 금액, 주중 매출 금액, 여성 매출 금액, 연령대 30대 매출 금액', ...(중략)..., 보증금 평균, 남성연령대 40 직장인구 수, 임대건수'] feature 20개 선정

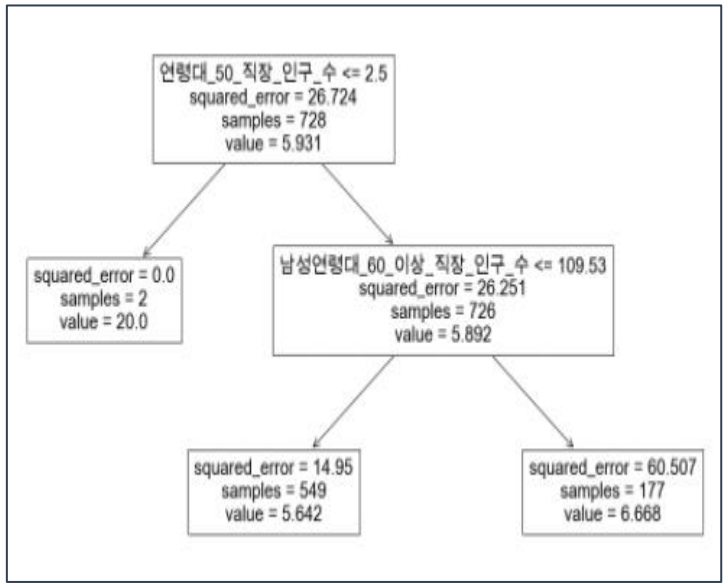




STEP 3

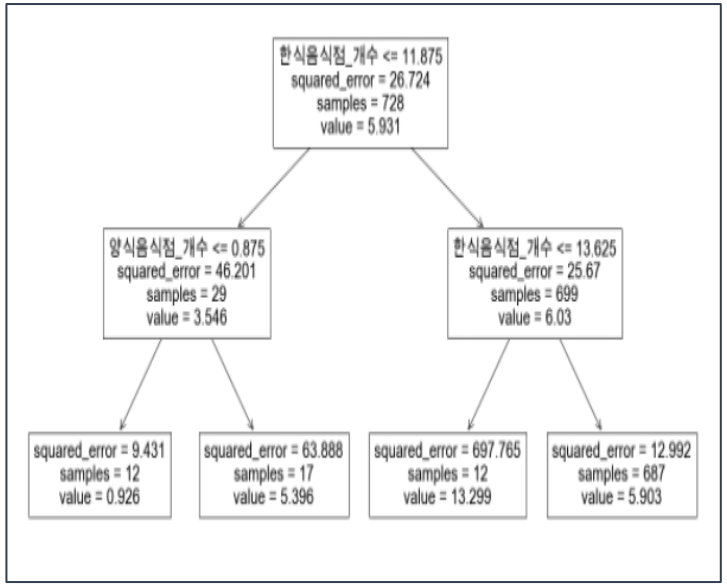
타겟 변수에 영향이 있는 feature selection

총 수집한 독립변수 특성이 될 수 있는 column의 개수(174개)가 많았기 때문에 특정 데이터의 종류가 많은 경우 가장 중요하다고 생각되는 feature를 선택해서 모델링을 진행



3. Decision Tree

엔트로피를 가장 낮추는 feature를 선정하여 구분 진행 decision tree에서 가장 먼저 선택되는 변수 선택



폐업률	이익
['남성연령대_60_이상_직장_인구_수', '목요일_생활인구_수', '버스_정거장_수', '시간대_06~11_매출_비율', '약국_수...{중략)..., '일식음식점_개수', '집객시설_수', '토요일_매출_비율', '한식음식점_개수'] feature 17개 선정	['양식음식점_개수', '여성연령대_30_직장_인구_수', '여성연령대_40_직장_인구_수', '연령대_10_생활인구_수...(중략)... ' 연령대_30_매출_비율 ' , ' 연령대_50_직장_인구_수 ' , ' 은행_수 '] feature 17개 선정

STEP 3

타겟 변수에 영향이 있는 feature selection

Selectkbest, shap value, decision tree로 선정한 feature 중 최종 Feature 7개 선택

4. VIF

- **Feature들 간 설명력을 분석, 높을 수록 다른 변수들로 설명이 가능한 feature이므로 우선적으로 제거**
- **3가지 방법으로 선택된 후보에 대해 VIF를 진행하여 VIF가 10보다 작은 feature들을 선택**
- **다중공선성을 제거, 변수간의 상관관계가 높은 것을 방지**

	VIF Factor	features
0	1.40000	버스_정거장_수
1	1.10000	보증금 평균
2	3.20000	시간대_2_생활인구_수
3	1.70000	약국_수
4	1.60000	연령대_60_이상_직장_인구_수
5	2.10000	임대건수
6	3.60000	중식음식점_개수

STEP 4
모델링

모델링

1. 두개의 Target변수(이익지수, 폐업률)에 대해 5등분하여 5개 모델을 사용하여 Classification 진행

이익지수의 모델링 결과

```
AdaBoostClassifier의 test 데이터 accuracy score :  0.3516483516483517
GradientBoostingClassifier의 test 데이터 accuracy score :  0.5879120879120879
RandomForestClassifier의 test 데이터 accuracy score :  0.64285714285714291
DecisionTreeClassifier의 test 데이터 accuracy score :  0.4945054945054945
LGBMClassifier의 test 데이터 accuracy score :  0.6098901098901099
```

폐업률의 모델링 결과

```
AdaBoostClassifier의 test 데이터 accuracy score :  0.23626373626373626
GradientBoostingClassifier의 test 데이터 accuracy score :  0.25274725274725274
RandomForestClassifier의 test 데이터 accuracy score :  0.25824175824175827
DecisionTreeClassifier의 test 데이터 accuracy score :  0.24175824175824176
LGBMClassifier의 test 데이터 accuracy score :  0.25824175824175827
```

STEP 4
모델링

모델링

2. So What??

이익지수의 모델링 결과
- 해당 **feature**들의 설명력이 좋다는 것을 알 수 있지만 후보군 선정에 활용하기는 어려움
- 그래서 **feature**들로 다음 해의 이익지수를 예측하는 작업을 진행
  - **Ex) 2017**년 자료로 **2018**년 이익지수를 예측
  - 시계열적 특성을 피하기 위해 **2017**년 이익지수는 **feature**에서 제외
- 후보군 추천을 위해 **regression**도 진행

폐업률의 모델링 결과
- 성능이 좋지 않아 사용하지 않기로 함

STEP 4
모델링

모델링

2. 추가 모델링 결과

이익지수의 모델링 결과(Classification)

```
AdaBoostClassifier CV Score 평균 = 0.41998, 표준편차 = 0.02874
GradientBoostingClassifier CV Score 평균 = 0.55808, 표준편차 = 0.03314
RandomForestClassifier CV Score 평균 = 0.60511, 표준편차 = 0.04827
DecisionTreeClassifier CV Score 평균 = 0.48608, 표준편차 = 0.03099
LGBMClassifier CV Score 평균 = 0.59922, 표준편차 = 0.04265
```

이익지수의 모델링 결과(Regression)

```
RandomForestRegressor의 test 데이터 rmse :  264,378,123
DecisionTreeRegressor의 test 데이터 rmse :  341,695,643
LGBMRegressor의 test 데이터 rmse :  286,328,413
```

- **Cf)** 이익지수의 평균은 **2.04**억, 표준편차는 **6.04**억

STEP 5
결론 도출

결과 (최종 후보지 선정)

2020년의 Feature를 이용해 RandomForest Regressor로 이익지수가 가장 높은 5개 행정동 선정

- 서초제2동, 구로제3동, 사직동, 서초제3동, 창제1동

서초2동 : 상당수가 주거지역(아파트)로 구성되어, 부촌인 곳에 북동부지역은 삼성계열사 사옥이 자리잡고있다. 높은 이익을 얻을 수 있으나 임대유지 등 지출이 커야한다.

구로3동 : 디지털1단지로 불리는 지역으로 동의 대부분을 구로디지털단지가 차지 시흥대로 주변으로 다수 상권이 형성되어 있고, 직장인 원룸촌이 많이 형성되어 있다.

STEP 5
결론 도출

결과 (최종 후보지 선정)

2020년의 Feature를 이용해 RandomForest Regressor로 이익지수가 가장 높은 5개 행정동 선정

- 서초제2동, 구로제3동, 사직동, 서초제3동, 창제1동

사직동 : 정부서울청사, 서울지방검찰청 등 강북 도심지역으로 근처 청운효자동과 평창동과 함께 서울 북부의 대표 부촌 지역
대한민국 대표 공공기관 주변으로 상권이 크게 발달되어 있는 것이 특징이다.

서초3동 : 북부지역은 법원,대검찰청 등 법조단지, 남부는 주택지역과 남부터미널이 위치해 있다.
서리풀 악끼거리나 예체능 공연을 보러오는 젊은 세대 중심의 상권이 발달 된 것이 특징이다.

STEP 5
결론 도출

결과 (최종 후보지 선정)

2020년의 Feature를 이용해 RandomForest Regressor로 이익지수가 가장 높은 5개 행정동 선정

- 서초제2동, 구로제3동, 사직동, 서초제3동, 창제1동

창1동 : 앞선 4개의 지역과는 다르게 주거지와 아파트 지역이 대다수이며 쌍문역 중심으로 상권이 발달 되었다. 초안산근린공원이 동 지역의 상당 부분을 차지해 녹지 공간이 많아 직장인구 보다 주거인구를 target으로 하는 창업 계획이 필요하다.

STEP 5
결론 도출

결과

2. 프로젝트의 활용

- **프로젝트의 결과대로 무조건 해당 행정동에 창업하다는 것이 유리하다는 것은 아님**
  - 같은 행정동이라도 위치에 따라 **feature**들이 다를 것임

- **다만, 우선적으로 카페 창업 위치에 대한 후보군 선정에 참고할 수 있을 것으로 판단됨**

- **또한 feature들이 이익 지수와 폐업률을 잘 설명한다면, 후보군에 대한 현장조사를 할 때 특히 눈여겨 보아야 할 체크 리스트로 활용할 수 있음**

  - 버스_정거장_수, 시간대_**2**_생활인구_수, 약국_수 , 연령대_**60**_이상_직장_인구_수, 중식음식점_개수, 보증금 평균, 임대건수

STEP 5
리뷰 및 보완점

리뷰 및 보완점

- **생각보다 데이터 가공이 쉽지 않았다.**

  - 출처가 다른 여러 데이터를 모아 하나의 데이터셋으로 만드는 작업이 쉽지 않았다.
  - 특히 행정동을 통일 시키는 작업이 어려웠다.

- **상대적으로 모델링에 시간투입을 많이 하지 못했다.**

  - 데이터 가공에 시간이 많이 투입되어 모델링 시간이 적어 아쉬웠다.
  - 폐업률에 대한 성능이 좋지 않아 사용하지 못하고 폐기
  - 이익지수에 대한 성능도 좀 더 높일 수 있지 않았을까 하는 아쉬움이 있다.

야 너두! 카페 창업 할 수 있어

서울시 **카페 상권분석** 결과 예측

감사합니다