공공 자전거 데이터 분석

출퇴근 시간대 공공자전거 수요 예측

20202682 김원중
20170920 김명수
20170429 김현종

0. 목차

1. 배경

따릉이 소개

“서울시 공공자전거(따릉이)의 정확한 수요 예측 필요”

- **따릉이: 서울시의 무인 공공자전거 대여서비스**
- 틈새 교통수단으로 자리 잡은 따릉이
  → **이용 건수의 절반이 출퇴근 시간대에 집중**
- 이용자가 증가하면서 대여소의 효율성 문제 제기
  → **정확한 수요 예측의 필요성**

HOME > 뉴스 > 서울시 > 서울시

공공자전거 ‘따릉이’ 운영 효율성 지적

그러면서 “현재와 같은 이용률에서는 유지관리에 대한 적자운영으로 자전거 서비스 및 품질이 개선되기 어렵다”면서 “현재 따릉이 구축과 운영을 유동인구 중심에서 실수요자 중심에 따른 위치 선정과 필요수량을 배치해야 한다”고 강조했다.

서울 ‘따릉이’ 누적대여 3천만건 돌파했다…56% 출퇴근 이용

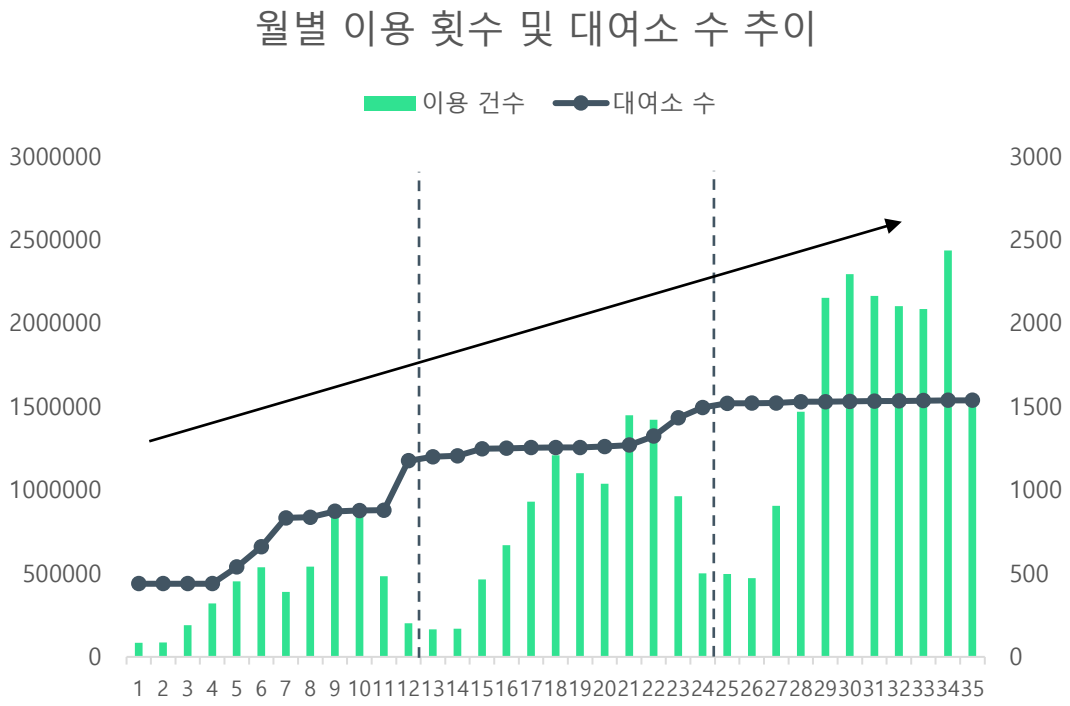
이로운넷=양승희 기자 | 승인 2019.11.04 10:00 | 댓글 0

운영 4년차 데이터 분석해 4일 발표…166만명 회원, 6명 중 1명 가입
도심지 자전거도로 확충 필요, 외국인 관광 코스로도 주목

2. 탐색적 자료 분석 (EDA)

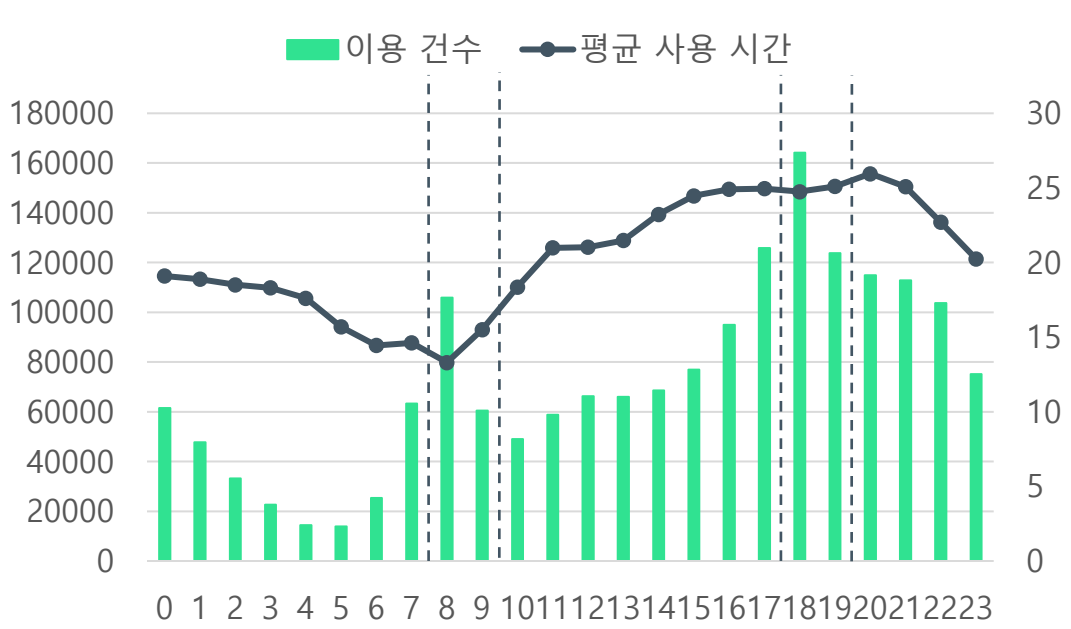
연간 이용 현황 분석

"이용 횟수는 계절성 주기를 띄며 매해 상승함"



- 연도별로 이용 횟수는 매해 상승하며, 보관소도 확충되고 있다.
- 이용 횟수는 여름, 가을에 상승하고 겨울, 봄에 하강하는 계절성 주기를 띈다.

"출퇴근 시간대에 이용이 급증함"

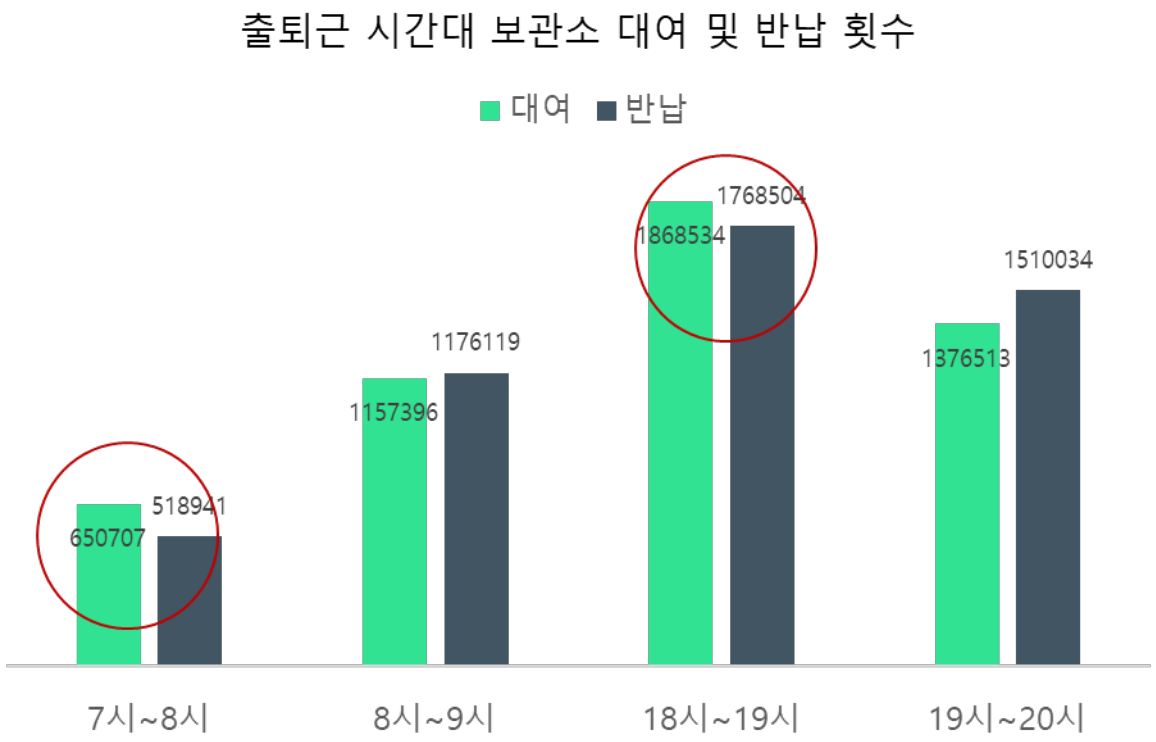


- 출퇴근 시간대와 연관된 8시~9시, 18시~19시에 이용이 급증하는 경향을 보임
- 오전에 비해 오후에 평균 사용 시간이 높은 것을 볼 수 있음

2. 탐색적 자료 분석 (EDA)

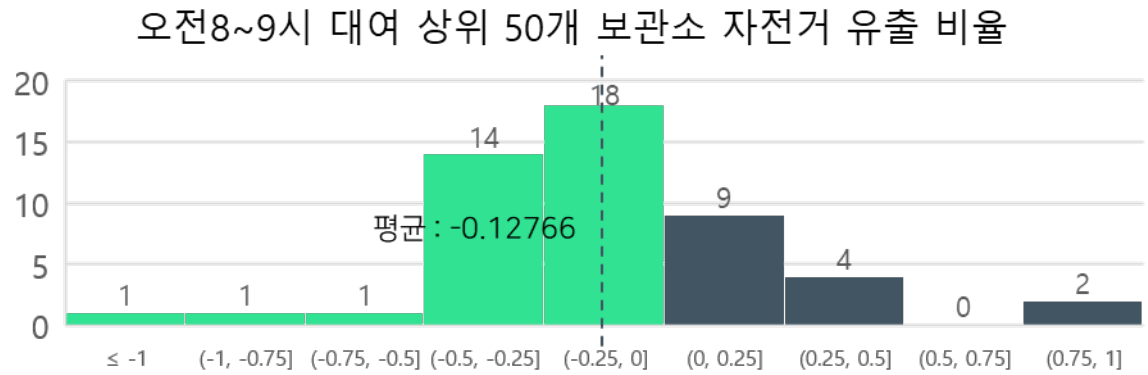
출 퇴근 시간대 운영 효율성 분석

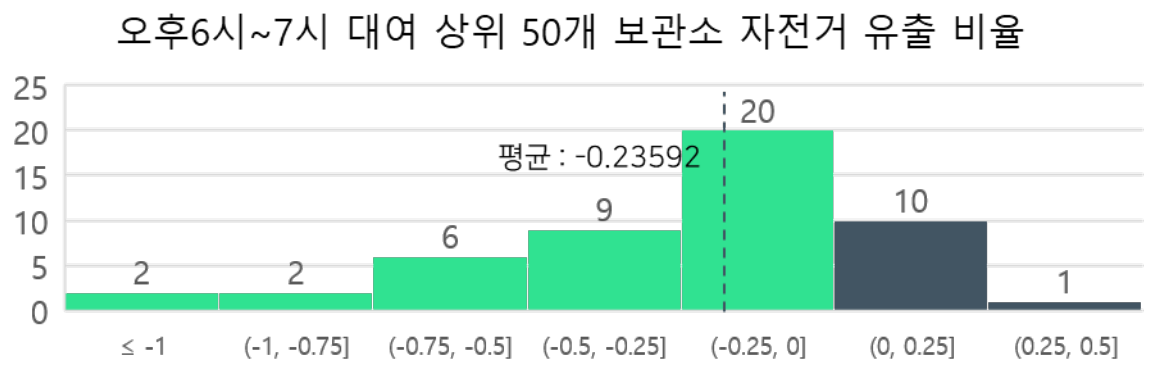
"출 퇴근 시간대의 대여와 반납의 차이가 발생"



- 시간의 구간을 더 짧게 잡으면 더 큰 차이가 있을 것으로 예상됨
- 대여가 높은 보관소에 대해서 Paired-t test 결과 또한 유의미한 차이가 있는 것으로 나타남 (8시~9시 P-value : 1.36E-07 …)

"대여가 높은 보관소는 대여,반납 순환이 이뤄지지 않음"

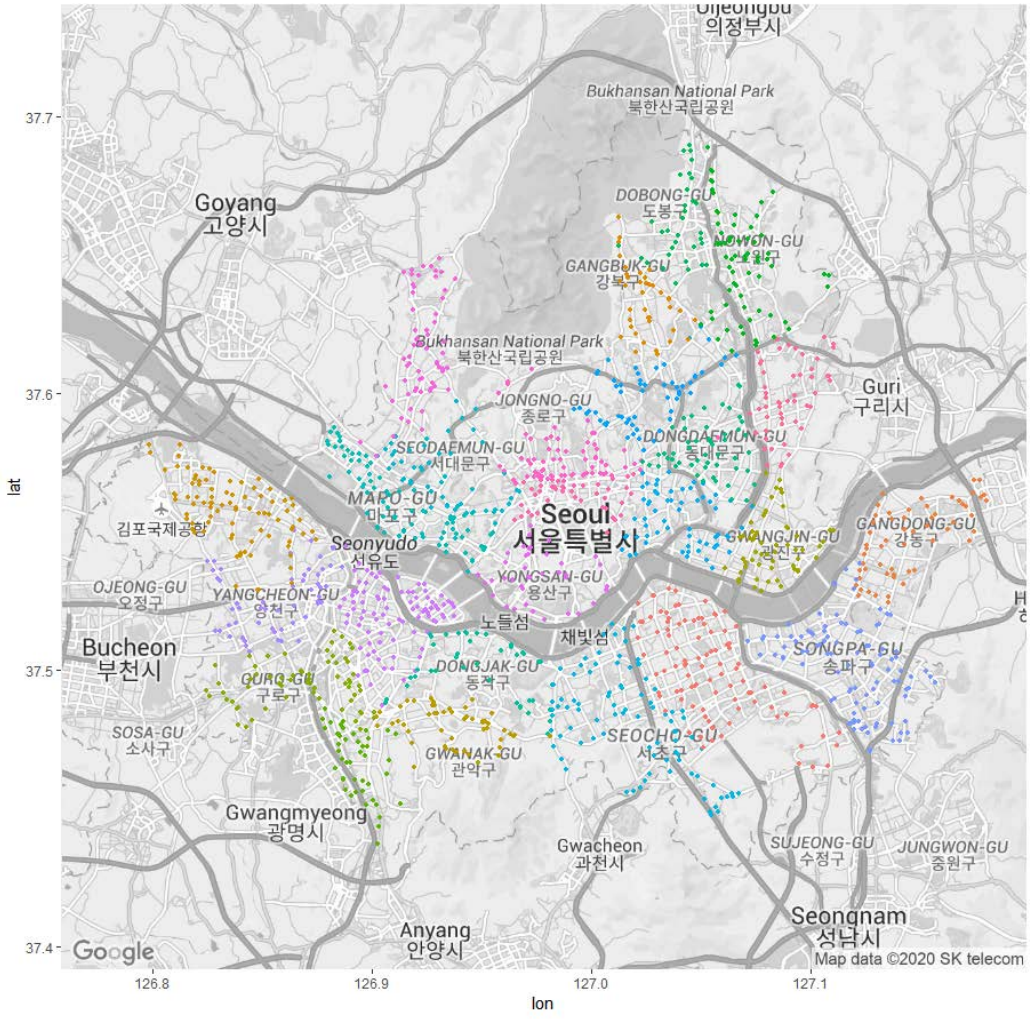




- 유출 비율을 (반납-대여)/(거치대 수)로 계산함
- 대여 이용이 높은 보관소는 대여와 반납의 격차가 커 보다 정밀한 수요 예측 및 재배치 필요

2. 탐색적 자료 분석 (EDA)

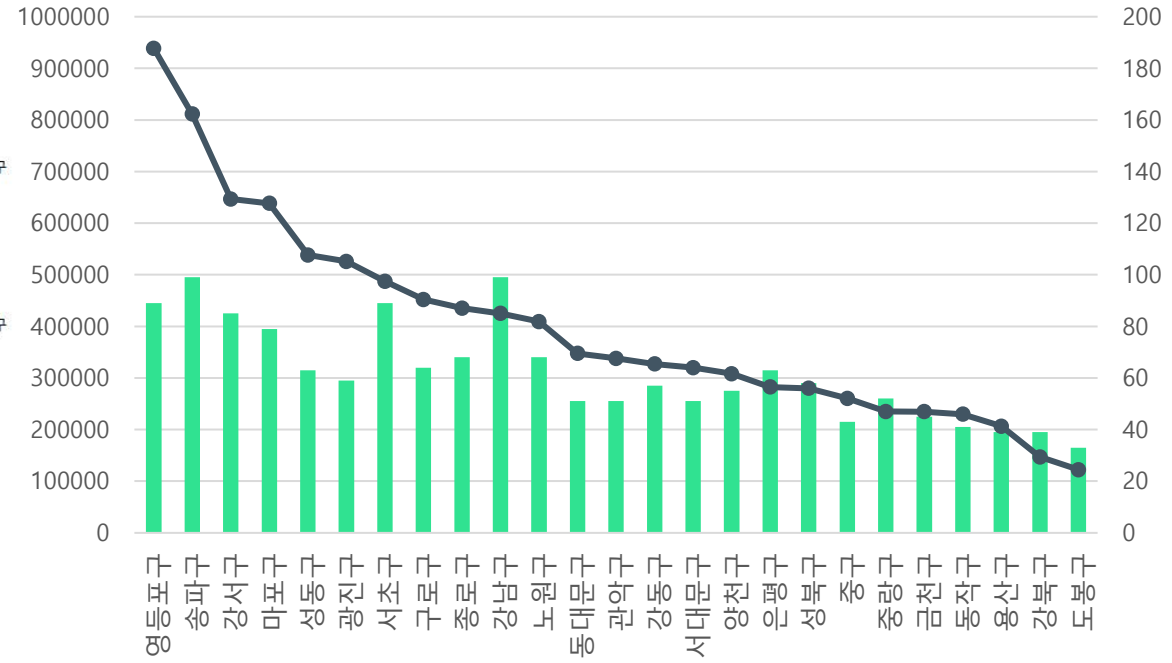
출 퇴근 시간대 자치구별 이용 현황 분석



서울시 따릉이 대여소의 분포

"출퇴근 시간대에 특정 구의 이용 빈도가 높음"

자치구 별 출퇴근 시간대 이용횟수 및 대여소 수



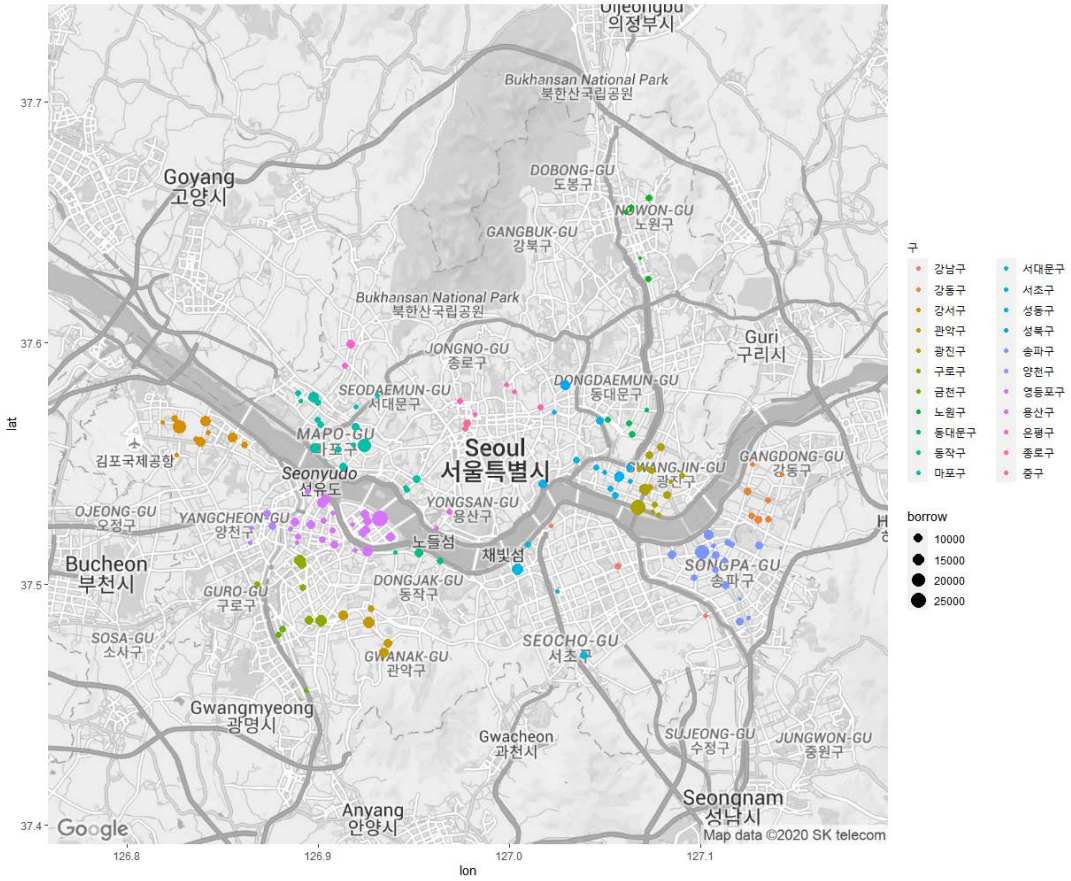
- 거치대 수는 대여소 별로 5대 ~40대 사이로 차이가 있다.
- 출퇴근 시간대에 자전거를 가장 많이 사용하는 구는 영등포구이다.

2. 탐색적 자료 분석 (EDA)

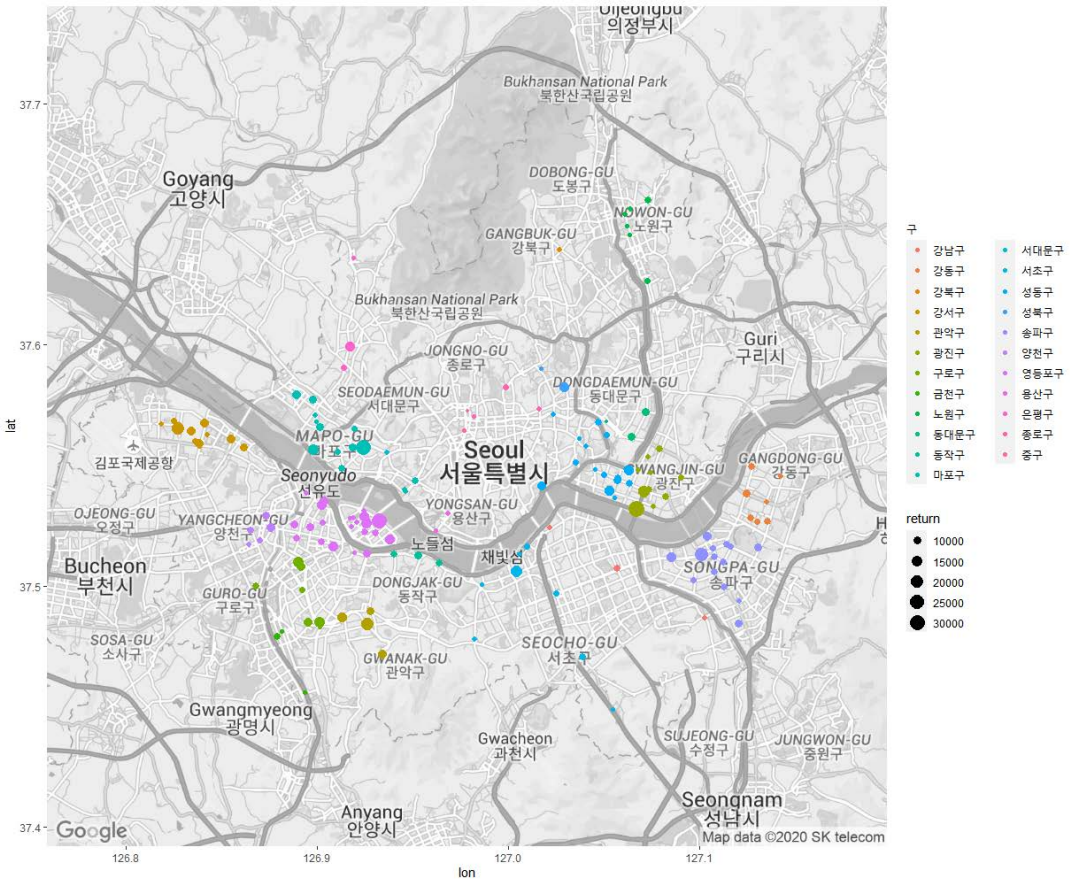
자치구별 이용 현황 분석

"한강 근처 자전거 도로를 주변으로 반납 대여가 많음"

대여량 상위 10% 대여소



반납량 상위 10% 대여소



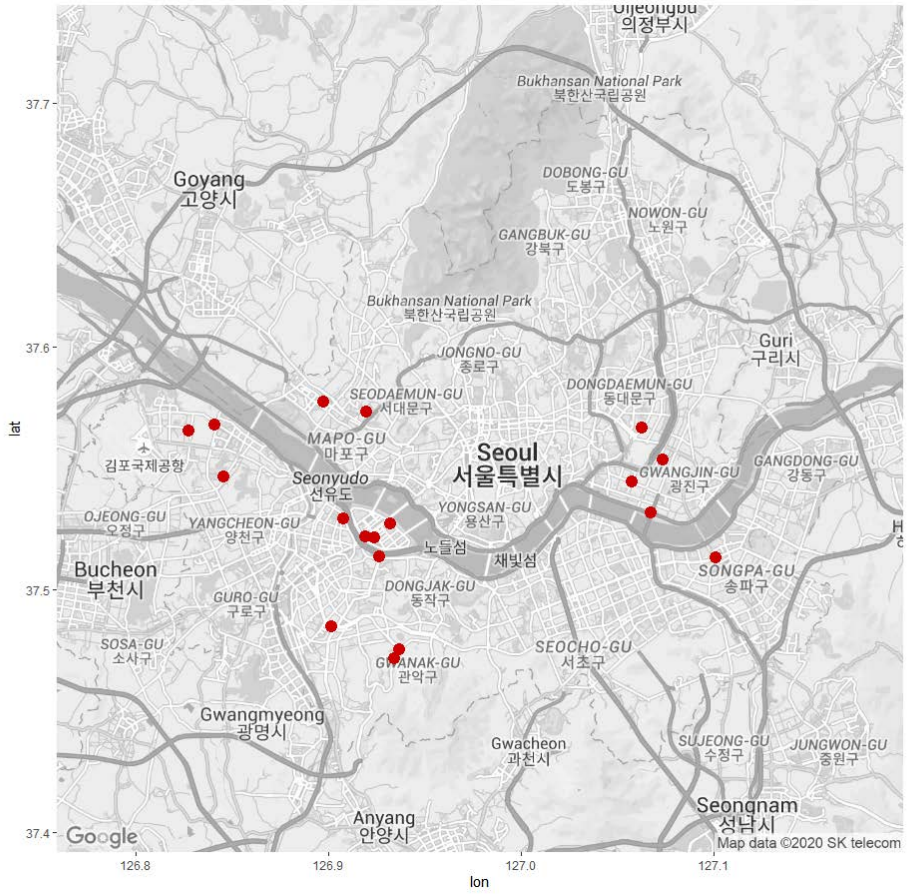
- 기간: 2018년 11월 ~ 2019년 10월

2. 탐색적 자료 분석 (EDA)

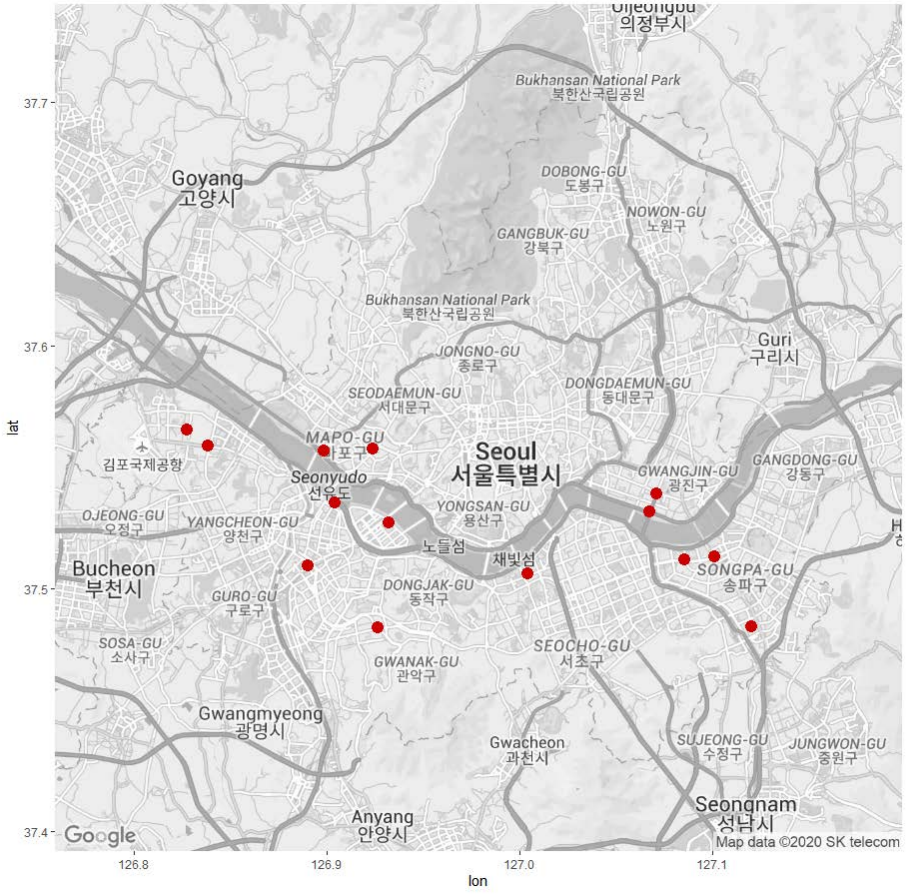
자치구별 이용 현황 분석

**"출근 시간대에는 직장, 지하철역 근처,
퇴근 시간대에는 지하철역 및 한강 근처에 반납 및 대여가 많음"**

출근 시간대 이용 상위 20개 대여소



퇴근 시간대 이용 상위 20개 대여소

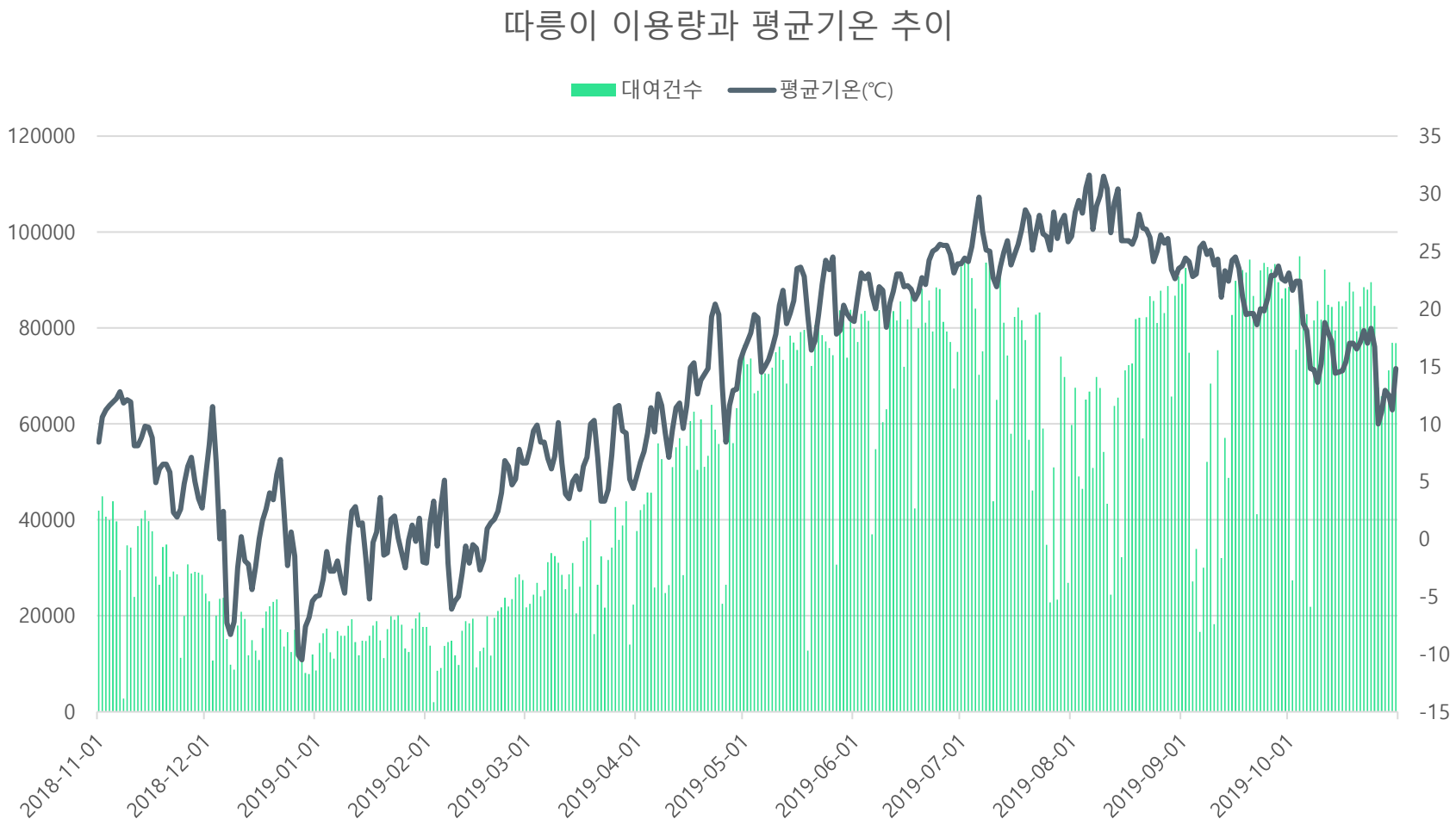


- 기간: 2018년 11월 ~ 2019년 10월

2. 탐색적 자료 분석 (EDA)

날씨와의 연관성 분석

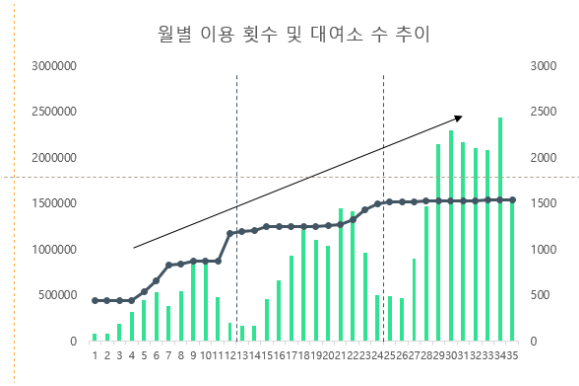
“따릉이 이용량과 기온 사이의 연관성을 볼 수 있음”

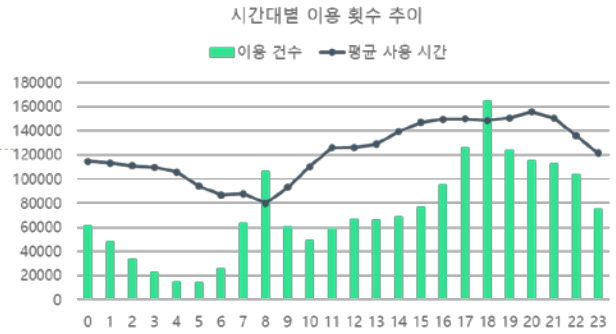


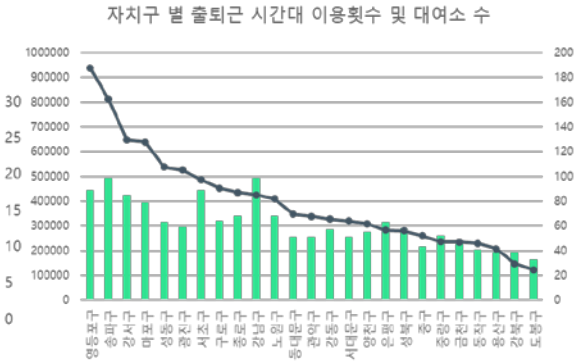
- 기간: 2018년 11월 ~ 2019년 10월

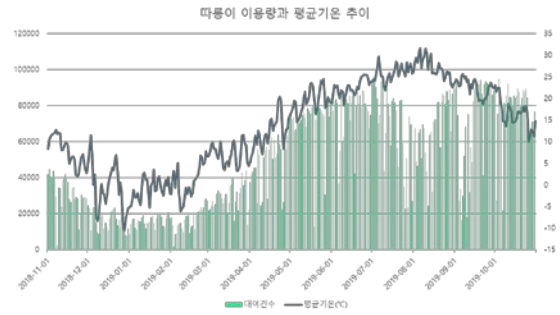
2. 탐색적 자료 분석 (EDA)

연구 범위 설정









최근 1년 간의 데이터
(2018년 11월 ~ 2019년 10월)

+

출퇴근 시간대
(7시~9시 & 18시~20시)

+

영등포구 보관소 데이터
(총 89개의 보관소)

날씨 데이터
(기온, 습도, 강수량, 미세먼지, 풍속)

3. 데이터 수집 및 전처리

데이터 수집

	A	B	C	D
1	자전거번호	대여일시	대여 대여소번호	대여 대여소명
2	SPB-21062	2019-09-03 8:44	646	장한평역 1번출구 (국민은행앞)
3	SPB-20126	2019-09-04 0:10	1372	KEB은행 고대점
4	SPB-00958	2019-09-04 0:41	1337	돈암성당 옆
5	SPB-06836	2019-09-04 8:48	529	장한평역 8번 출구 앞
6	SPB-16611	2019-09-05 8:46	646	장한평역 1번출구 (국민은행앞)
7	SPB-18891	2019-09-05 8:51	529	장한평역 8번 출구 앞
8	SPB-22839	2019-09-09 8:33	646	장한평역 1번출구 (국민은행앞)
9	SPB-16146	2019-09-10 8:41	646	장한평역 1번출구 (국민은행앞)
10	SPB-18250	2019-09-10 8:51	529	장한평역 8번 출구 앞

서울특별시 공공자전거 대여정보 [서울 열린데이터 광장]

	A	B	C	D	E	F	G	H
1	구	ID	대여소명	대여소주소	lat	lon	기준시작일자	거치대수
2	마포구	101	101. (구)합	서울특별시	37.54956	126.9058	2015-09-06 23:40	5
3	마포구	102	102. 망원	서울특별시	37.55565	126.9106	2015-09-06 23:42	20
4	마포구	103	103. 망원	서울특별시	37.55495	126.9108	2015-09-06 23:43	14
5	마포구	104	104. 합정	서울특별시	37.55063	126.915	2015-09-06 23:44	13
6	마포구	105	105. 합정	서울특별시	37.55001	126.9148	2015-09-06 23:45	5
7	마포구	106	106. 합정	서울특별시	37.54865	126.9128	2015-09-06 23:46	10
8	마포구	107	107. 신한	서울특별시	37.55751	126.9185	2015-09-06 23:47	5
9	마포구	108	108. 서교	서울특별시	37.55275	126.9186	2015-09-06 23:51	10
10	마포구	109	109. 제일	서울특별시	37.54769	126.92	2015-09-07 1:27	10

서울특별시 공공자전거 대여소 정보 [서울 열린데이터 광장]

	A	B	C	D	E
1	날짜	지점	평균기온(℃)	최저기온(℃)	최고기온(℃)
2	2020-05-01	108	20.2	16.4	26.2
3	2020-05-02	108	20.3	18	23.9
4	2020-05-03	108	21.8	17	27.4
5	2020-05-04	108	20.2	14.8	25.3
6	2020-05-05	108	16	13.1	19.3
7	2020-05-06	108	19.4	11.1	27.6
8	2020-05-07	108	20.7	14.9	26.5
9	2020-05-08	108	19.9	14.3	27.3
10	2020-05-09	108	14.2	12.7	16.8

기온 데이터 (서울시/일별)
[기상청 기상자료개방포털]

	A	B	C	D	E	F
1	측정일자	권역코드	권역명	측정소코드	측정소명	미세먼지(㎍/㎥)
2	20180101	100	도심권	111121	중구	32
3	20180101	104	동남권	111273	송파구	52
4	20180101	104	동남권	111262	서초구	49
5	20180101	104	동남권	111261	강남구	34
6	20180101	103	서남권	111301	양천구	37
7	20180101	103	서남권	111281	금천구	39
8	20180101	103	서남권	111251	관악구	38
9	20180101	103	서남권	111241	동작구	40
10	20180101	103	서남권	111231	영등포구	47

미세먼지 농도 데이터 (구별/일별)
[서울 열린데이터 광장]

	A	B	C	D	E	F
1	format: day	hour	value location:60_127 Start : 20181101			
2	1	0	50			
3	1	100	44			
4	1	200	36			
5	1	300	32			
6	1	400	29			
7	1	500	29			
8	1	600	32			
9	1	700	38			
10	1	800	45			

습도/강수량/풍속 데이터 (구별/시간별)
[기상청 기상자료개방포털]

3. 데이터 수집 및 전처리

데이터 전처리

대여 정보 데이터를 바탕으로 대여 및 반납 대수 집계

2018년 11월 1일 ~ 2019년 10월 31일 사이의 대여 정보의 대여날짜, 시각 대여소를 바탕으로 대여 및 반납 대수를 집계함

위치 정보를 기반으로 대여소와 기후 데이터를 Join

2018년 11월 1일 ~ 2019년 10월 31일 사이의 기후관련 데이터를 대여소의 지역, 위치 정보와 날씨데이터의 위치 정보를 바탕으로

[14]
```
# select "borrow" data that has time between 7,9 and 18,20, 4개 구 선택
bo = data['borrow_date'].apply(lambda x: x if str(x.hour)=='7' or str(x.hour)=='8' or str(x.hour)=='18' or str(x.hour)=='19' else np.nan)
.dropna().index
data_borrow = data.iloc[bo,:][['borrow_date','borrow_id','number','weekdate']]
data_borrow = data_borrow[data_borrow['borrow_id'].isin(id)]
```

[15]
```
# Date format change
borrow_hour = data_borrow['borrow_date'].apply(lambda x : str(x.hour)+'00')
borrow_year = data_borrow['borrow_date'].apply(lambda x : str(x.year))
borrow_month = data_borrow['borrow_date'].apply(lambda x : '0'+str(x.month) if x.month < 10 else str(x.month))
borrow_day = data_borrow['borrow_date'].apply(lambda x : '0'+str(x.day) if x.day<10 else str(x.day))
data_borrow['time']= borrow_hour
data_borrow['date']=borrow_year+borrow_month+borrow_day
data_borrow.rename(columns = {'borrow_id' : 'id'}, inplace = True)
data_borrow.drop(['borrow_date'],axis='columns',inplace=True)
```

[18]
```
# groud by borrow_id and time and counts
grouped1 = data_borrow.groupby(['id','date','weekdate','time'])['number'].count()
borrow = pd.DataFrame(grouped1)
borrow.columns = ['count']
borrow
```

```
for filepath in glob.iglob('r_gu/*.csv'):
    df = pd.read_csv(filepath)
    indicator = filepath[-6]
    count_row = df.shape[0]
    if count_row != 1460:
        print(filepath, 'row number not 1460')
        exit()

    for index, row in df.iterrows():
        d = day_count(int(row.iloc[3]))
        g = d_gu[filepath[7:-8]]
        t = d_time[int(row.iloc[1])]
        idx = 100 * d + 4 * g + t
        value = row.iloc[2]

        if indicator == '1':
            main.loc[idx, '습도'] = value
        elif indicator == '2':
            main.loc[idx, '강수량'] = value
        elif indicator == '4':
            main.loc[idx, '풍속'] = value
        else:
            print('indicator error')
            exit()
```

3. 데이터 수집 및 전처리

데이터 전처리

① 대여 이력 데이터 결측치 제거

대여 이력 데이터 중 2019년 9월 7일의 데이터가 없음.
그 시기의 태풍 링링의 영향으로 보이며 제거.

② 날씨 데이터 결측치 전날 데이터로 대체

미세먼지 데이터에 총 10일 분량의 데이터가 없음.
미세먼지 농도는 연속적이라고 가정, 전 날의 데이터로 대체.

③ 대여/반납 횟수 이상치 제거

보관소 별 모델을 학습시키면서, 모델의 성능을 높이기 위해
양극단의 $mean \pm 3 * \sigma$ 밖의 값을 제거. (이상치 비율 1.04%)
이는 공휴일 혹은 지역 내 행사 등으로 발생.

최종 데이터 완성

Row : 129940
Variables : 9

	A	B	C	D	E	F	G	H	I	J	K
		대여소ID	구	일자	요일	시간	기온	습도	강수량	풍속	미세먼지
1		200	영등포구	20181101	4	700	8.4	39	0	1.3	43
2		200	영등포구	20181101	4	800	8.4	42	0	1.1	43
3		200	영등포구	20181101	4	1800	8.4	70	0	2.1	43
4		200	영등포구	20181101	4	1900	8.4	75	0	2.4	43
5		200	영등포구	20181102	5	700	10.6	40	0	1.8	46
6		200	영등포구	20181102	5	800	10.6	38	0	2	46
7		200	영등포구	20181102	5	1800	10.6	78	0	1.5	46
8		200	영등포구	20181102	5	1900	10.6	75	0	1.2	46
9		200	영등포구	20181103	6	700	11.2	49	0	2.1	57
10		200	영등포구	20181103	6	800	11.2	49	0	1.6	57
11		200	영등포구	20181103	6	1800	11.2	81	0	0	57
12		200	영등포구	20181103	6	1900	11.2	73	0	1.4	57
13		200	영등포구	20181104	7	700	11.6	43	0	2.5	48
14		200	영등포구	20181104	7	800	11.6	45	0	1.9	48
15		200	영등포구	20181104	7	1800	11.6	89	0	1.6	48

4. 모델 선정 및 분석

모델 선정

변수 설정

독립변수	정량형 변수	기온(℃), 습도(%), 강수량(mm), 풍속(m/s), 미세먼지 농도(㎍/㎥)
	범주형 변수	요일(월 ~ 금), 시간(1 hour)
종속변수	따릉이 대여량(건수) 따릉이 반납량(건수)	

예측 모델

- 회귀 (Regression) : Ridge, Lasso, Elastic Net, Polynomial Regression, Support Vector Regression
- 분류 (Classification) : Random Forest, Xgboost, Logistic Regression

5. 분석 결과 및 결론

분석 결과 & 예측 성능

"Ridge와 Xgboost 모델의 성능이 높게 나타남"

MAE (Mean Absolute Error)

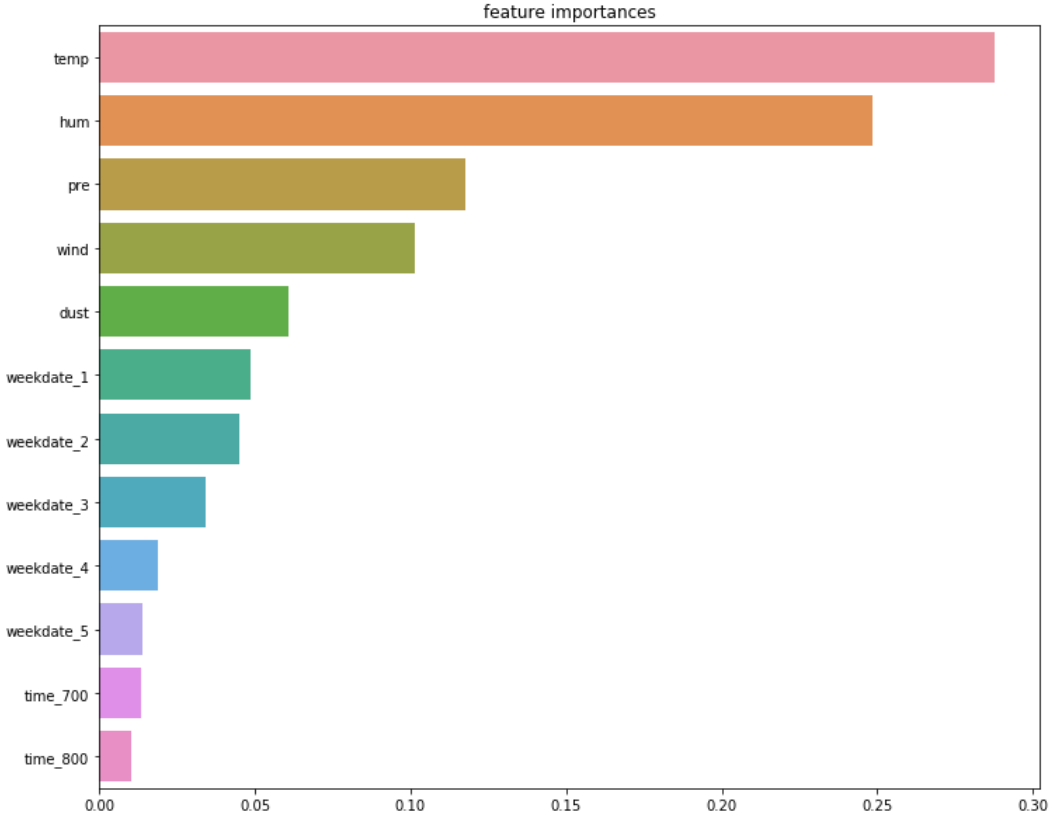
	출근시간대 대여	출근시간대 반납	퇴근시간대 대여	퇴근시간대 반납
Ridge	**1.62**	**1.71**	**2.63**	**2.13**
Lasso	1.79	1.94	2.78	2.21
Elastic Net	1.78	1.98	2.79	2.21
Polynomial Regression	1.67	1.71	3.19	2.23
Logistic Regression	1.86	2.02	3.1	2.51
Random Forest	1.81	1.84	3.13	2.37
Xgboost	**1.78**	**1.9**	**3.08**	**2.36**

5. 분석 결과 및 결론

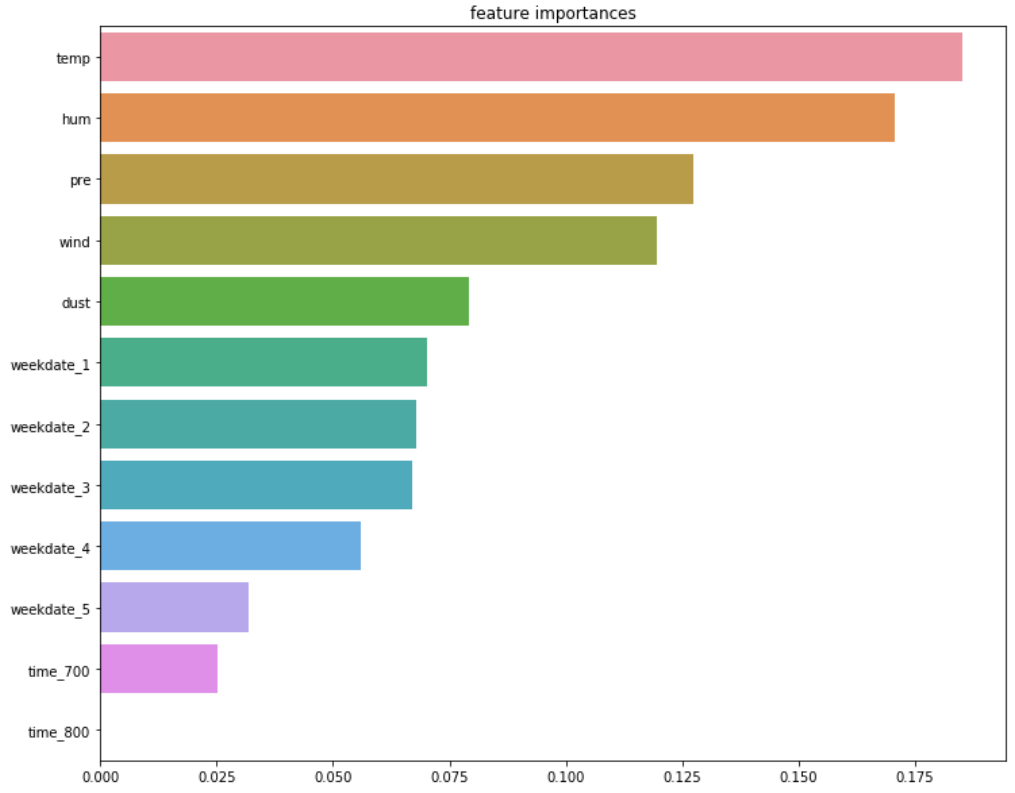
분석 결과 & 예측 성능

"기온과 습도의 영향력이 가장 큰 것으로 나타남"

Random Forest



Xgboost

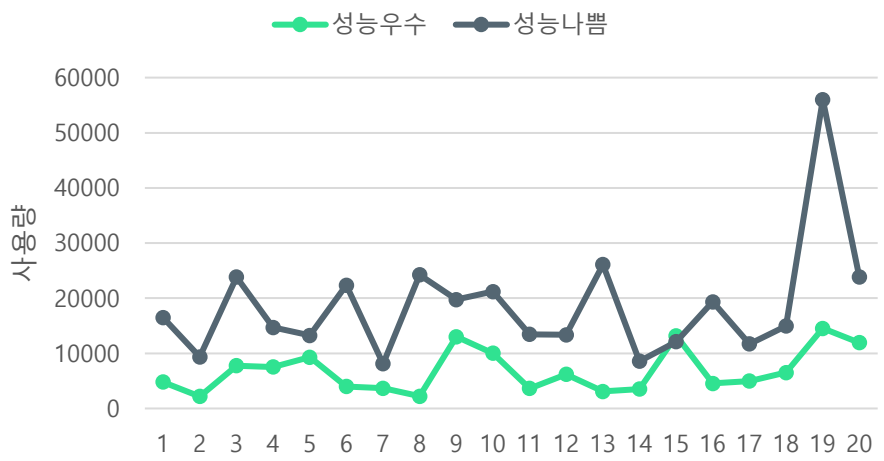


5. 분석 결과 및 결론

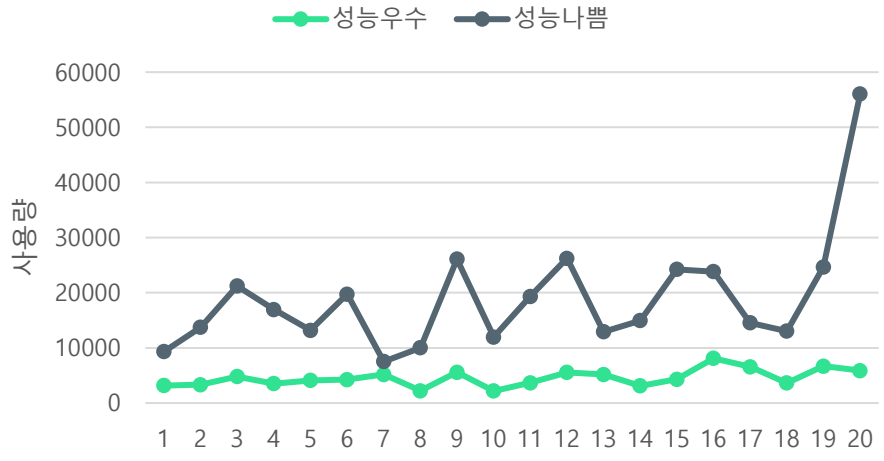
분석 결과 & 예측 성능

"사용량이 높은 보관소의 예측 성능이 떨어짐"

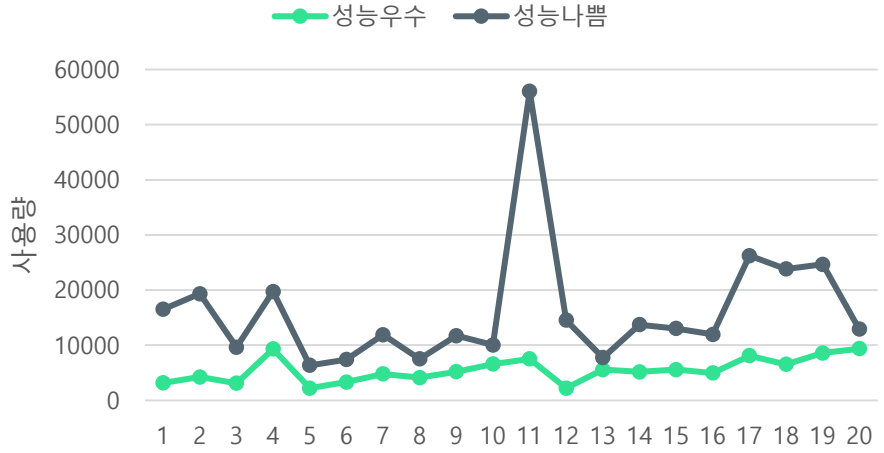
출근시간 대여량 예측



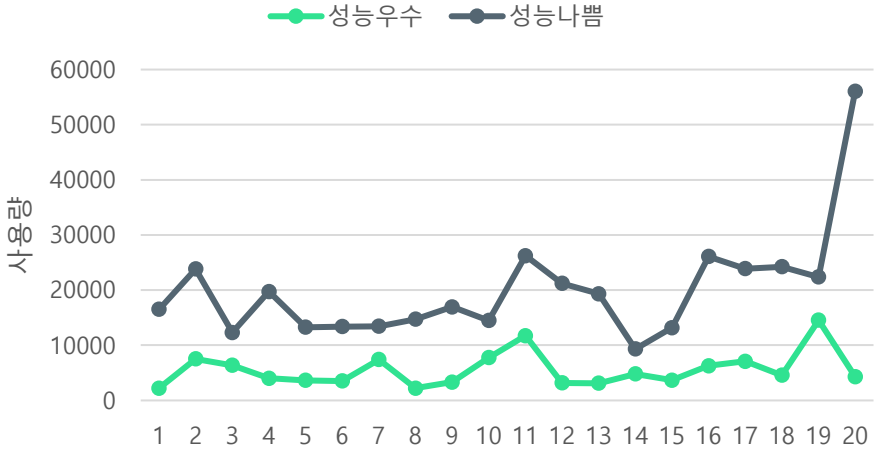
퇴근시간 대여량 예측



출근시간 반납량 예측



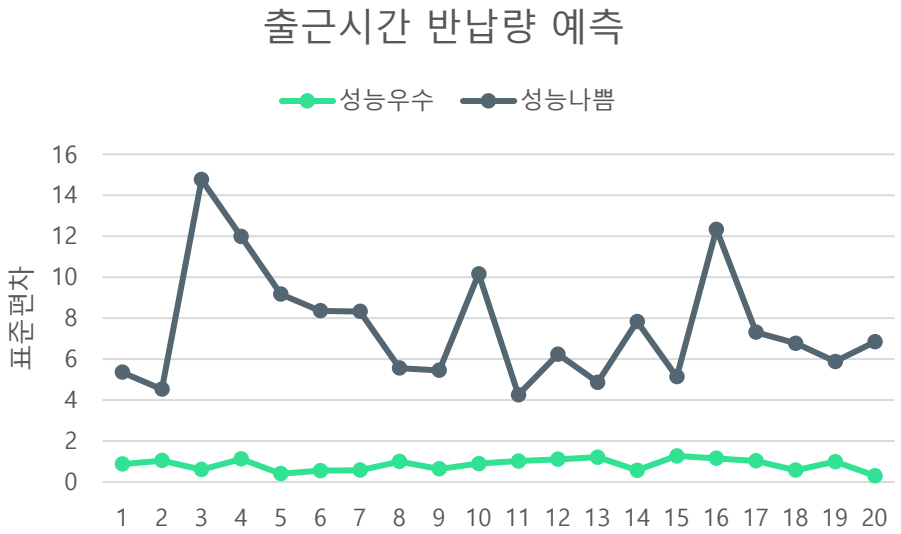
퇴근시간 반납량 예측

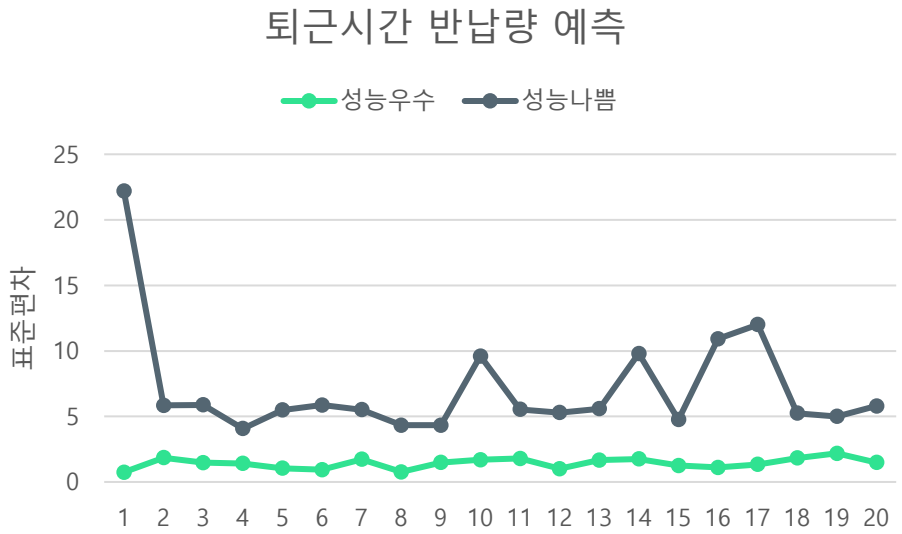


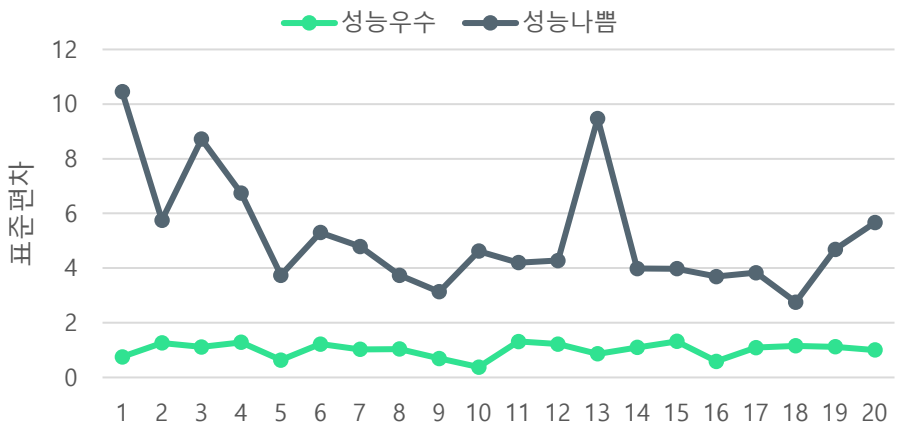
5. 분석 결과 및 결론

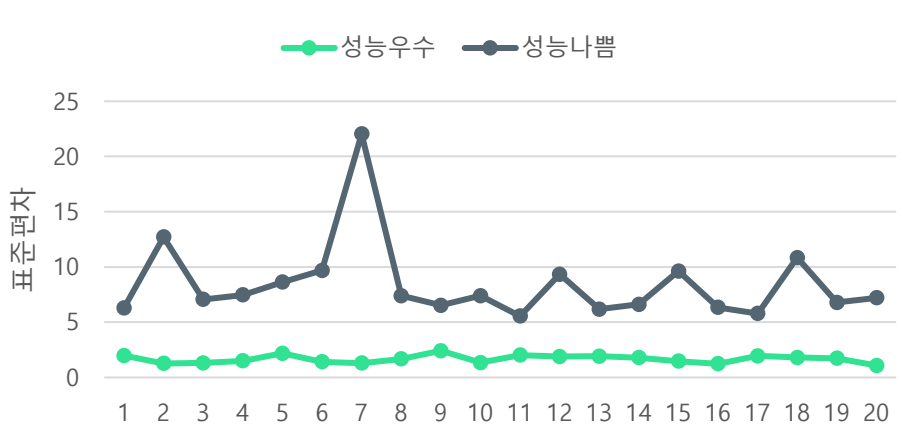
분석 결과 & 예측 성능

“반납 및 대여량의 표준 편차가 큰 대여소의 예측 성능이 떨어짐”









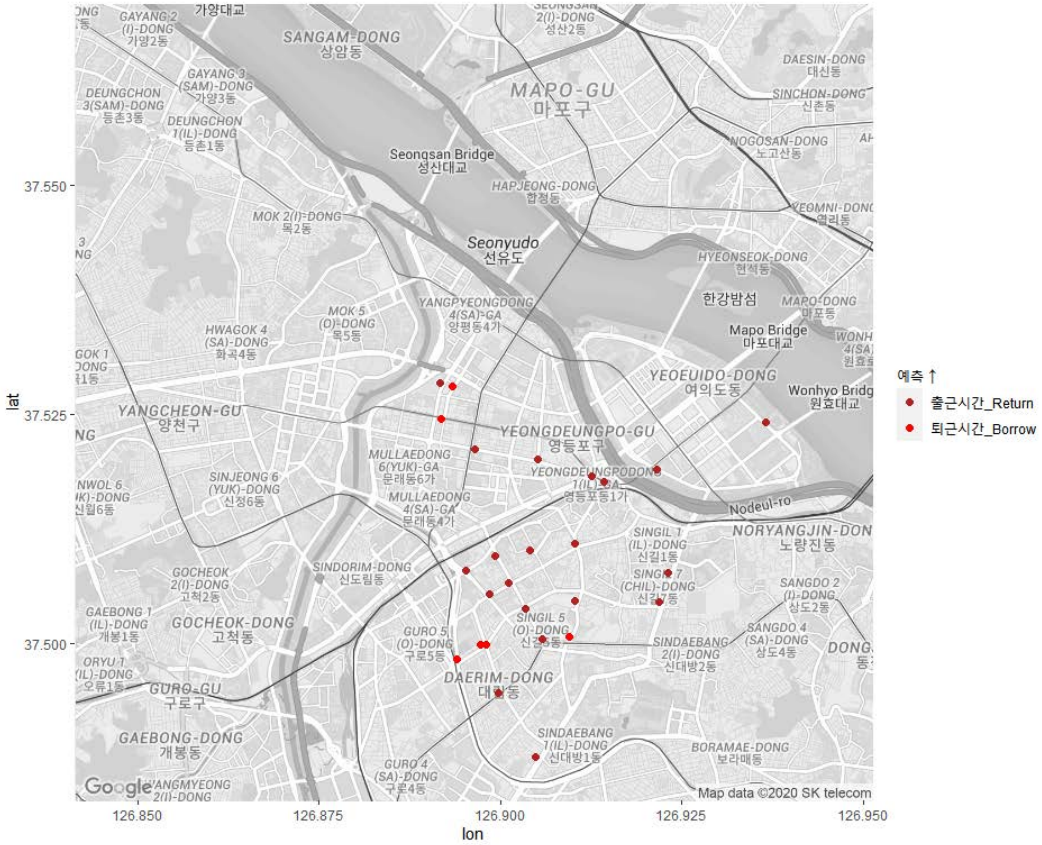
5. 분석 결과 및 결론

시사점 & 추후 연구

"교통 시설, 주택지, 회사 밀집 지역으로 대여소 분류를 통한 모델 정밀화"

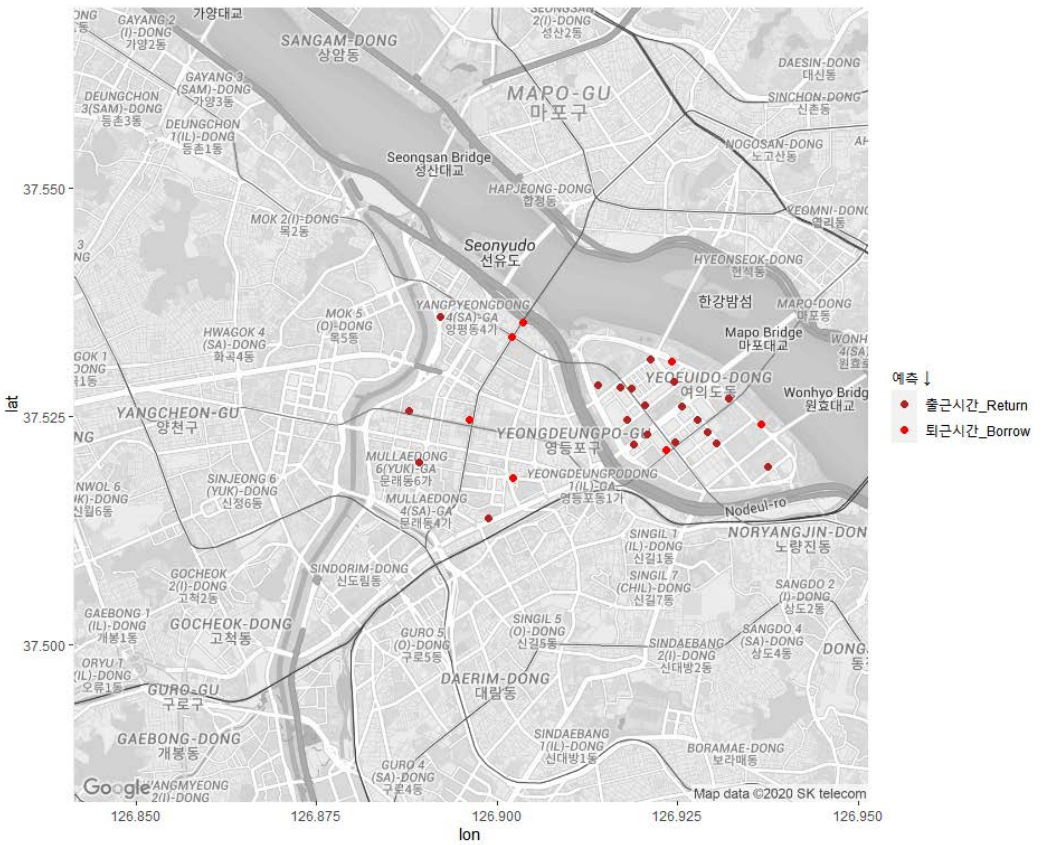
주거지역을 중심으로 예측이 잘되는 보관소가 분포

출근시간 반납↓ 퇴근시간 대여↓



회사 밀집 지역을 중심으로 예측이 안되는 보관소가 분포

출근시간 반납↑ 퇴근시간 대여↑



THANK YOU