

# 수요와 리밸런싱을 고려한 최적 자전거 거스레이션 제안

## 1. 공모 배경

과제의 목표는 공공 자전거의 활성화와 불만족도를 낮추는 것이다. 이 해당 목표에 대해 크게 3가지의 방향을 갖고 분석을 진행하였다.

### 1.1 동기

공공성을 가지는 과제의 목표가 마음에 들었고, 공간 데이터를 기반으로 새로운 데이터 분석에 도전하고 싶었다. 도전을 통해 개인의 성장과 더불어 좋은 결과를 도출하여 공공의 이익에 일조하고 싶다.

### 1.2 분석 방향

#### 1. 잠재적 수요 발굴

- 새롭게 자전거 거치대를 배치하였을 때 발생하는 수요를 파악할 필요가 존재한다.
- 접근성의 문제 때문에 놓치고 있는 수요를 발굴해야 한다.
- 주변 위치에 따른 잠재적 수요를 측정한다.

#### 2. 공공성 확보

- 공공 자전거는 많은 고양시민들이 이를 즐길 수 있게 하는 것이 중요한 요소이다.
- 서비스의 사각 지대를 어느정도 고려하는 것이 필요하다.

#### 3. Rebalancing 문제

- 자전거 거치 대수의 불균형 문제를 해결하는 것이 Rebalancing 문제이다.
- 자전거를 많이 배치하는 것 외에도 이의 수량을 조절하는 것이 필요하다.
- 재배치 비용을 줄이기 위해서라도, Rebalancing을 고려해야 한다.
- 기존의 거치대를 기반으로 새로운 거치대를 배정 했을 때의 상호작용을 고려한다.

- 용어 정의

- 총 자전거 이용량 (반입량+반출량)의 일별 평균 : 절대 수요  

$$(return + lease)/totalDay$$
- 총 자전거 변화량 (반입량-반출량)의 일별 평균 : 상대 수요  

$$(return - lease)/totalDay$$
- 과제에 소요된 시간
  - 8/10~9/10 하루 5시간 30일 3명 450시간

## 1.3 사용한 라이브러리

1. seaborn
2. matplotlib
3. geopandas
4. nx
5. folium
6. scipy

## 1.4 데이터 정제

- 1.4.0 코드 정의서
- 1.4.1 정류장 기반 table
- 1.4.2 운행이력 기반 table
- 1.4.3 행정구역 데이터 로드
- 1.4.4 인구 거주 분포도 로드
- 1.4.5 빌딩 연면적 분포도 로드
- 1.4.6 행사장, 박물관, 지하철역, 버스정류장, 학교, 체육시설 , 주차장 위치 Table
  - 1.4.6.1 행사장/공연장/박물관&미술관
  - 1.4.6.2 버스정류장
  - 1.4.6.3 지하철 역
  - 1.4.6.4 공공시설
  - 1.4.6.5 체육시설

- 1.4.6.6 주차장
- 1.4.7 건물 / 도로 / 인도 데이터 로드
  - 1.4.7.1 도로명 주소 건물
  - 1.4.7.2 도로
  - 1.4.7.3 인도
- 1.4.8 고도 데이터 로드
- 1.4.9 도시 계획 데이터 로드

## 2. EDA

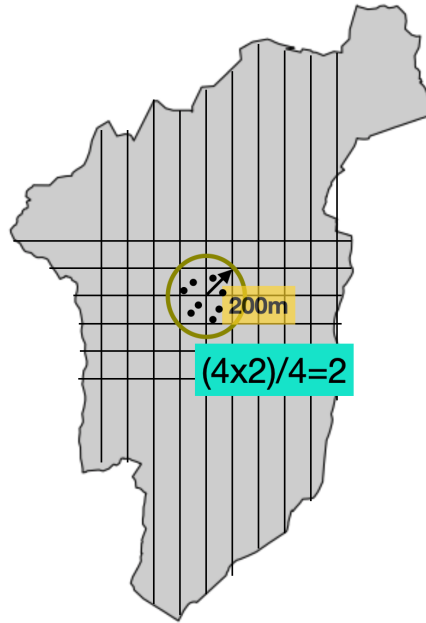
탐험적 데이터 분석을 통해 해당 문제가 어떠한 특징을 갖고 있고 문제점을 가지고 있는지 이해한다.

### 2.0 EDA 전용 데이터 셋 만들기

- 2.0.1 정류장 데이터에 고도 데이터를 추가
- 2.0.2 정류장 데이터에 운영이력 처리한 변수를 추가
- 2.0.3 행정동 당 자전거 거치대 개수 정보를 추가
- 2.0.4 위치 데이터 시각화

### 2.1 공공성의 관점

- 2.1.1 인구 대비 배치도
  - 2.1.1.1 행정동별 인구수
  - 2.1.1.2 자전거 거치대 대비 인구 수
- 2.1.2 배치/미배치 지역 구분
  - 2.1.2.1 미배치 지역 데이터화
  - 2.1.2.2 배치 지역 데이터화



## 2.2 수요의 관점

- 2.2.1 행정동 별 구분
  - 자전거 개수
  - 총 자전거 이용량 (반입량+반출량)의 일별 평균 : 절대 수요
  - 총 자전거 변화량 (반입량-반출량)의 일별 평균 : 상대 수요
- 2.2.2 상업/거주 인구 대비 이용량

## 2.3 접근성의 관점

- 2.3.1 거치대 기반 고도
- 2.3.2 거치대 근처 교통 시설
  - 버스정류장
  - 지하철역
- 2.3.3 거치대 근처 공공시설
  - 교육시설
  - 문화시설
  - 체육시설
- 2.3.4 고양시 도로 파악

- 시군구 도로

## 2.4 Rebalancing의 관점

- 2.4.1 시간대별 이용량

- 2.4.1.1 월별
- 2.4.1.2 요일별
- 2.4.1.3 일별
- 2.4.1.4 시간별

- 2.4.2 거치대 별 반입/반출량

- 2.4.2.1 거치대 별 상대 수요 분포

각각의 거치대 이용량의 일별 평균(상대수요)을 기반으로 거치대 별 반입/반출에 대해 영향력을 파악한다.

- 2.4.2.2 특이 station

앞서 분석을 진행하며, 대부분의 거치대에서 반출이 많다는 사실을 파악했다. 거치대 중 특이한 것이 있는지 분석하고, 이를 어떻게 활용할지 파악한다.

앞서 상대 수요 평균이 -2.3 이 나오는 것은 모든 거치대에서 반출이 평균적으로 -2.3개 정도 많다고 생각할 수 있다. 하지만 자전거의 총량이 변화하지 않는다는 가정을 도입한다면 이는 현실적으로 가능하지 않다. 따라서 0번 station의 의미를 파악해 본다.

결론적으로 두가지 사실을 파악할 수 있다.

1. 0번 station은 반입만 있는 거치대이다.
2. 반입의 평균이 다른 거치대 상대 수요의 평균과 정확하게 일치한다.

이러한 점을 근거로 우리는 이 0번 거치대는 중앙에서 거치대의 Rebalancing을 해소하기 위한 운반이라고 가정하였다. 이러한 점을 가정한다면, 0번 거치대 이외의 모든 거치대의 상대수요는 평균적으로 2.3 더해져야 할 것이다.

- 2.4.3 네트워크 분석

## 2.5 결론

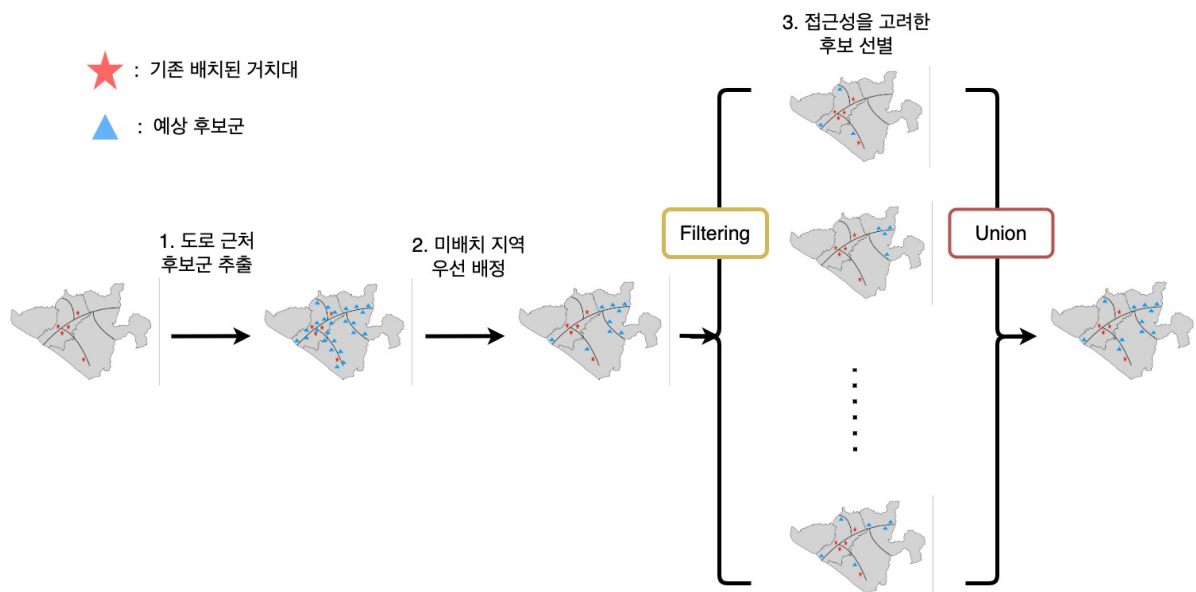
1. 상업 지역과 주거 지역을 나누어서 반영해야 한다.
2. 새롭게 배치되는 거치대는 다양한 접근성을 고려해야 한다.

3. 현재 미배치된 곳이 많은 것으로 파악되고, 공공성을 확보하기 위해 배치 우선순위를 배정해야 한다.
4. 정류장 별로 반입/반출량이 상이한 것으로 파악되고, 이러한 불균형을 해소하는 장소를 선정해야 한다.
5. 0번 station의 반입 평균과 나머지 정류장의 반입출 평균이 같다. 이 점을 고려해야 한다.

## 3. 후보군 생성 알고리즘

EDA 결과를 기반으로 예상되는 후보군을 추출한다.

### 3.0 전체 프로세스



#### 1. 도로 근처의 후보군을 추출한다.

- 자전거 거치대의 접근성을 고려하여 도로 주변에 이를 배치한다. 대다수의 사람들은 도로/인도를 기준으로 통행한다. 이러한 통행 동선을 고려하여 도로에서부터 후보군을 선택한다.

#### 2. 미배치 지역 우선 배정

- 가정 : 현재 거치대가 cover하는 구역은 반경 200m이다.

- 이러한 가정을 기반으로, 배치가 된 구역에 추가적인 거치대를 배정하는 것은 비효율적이라 판단하였다. 따라서 이렇게 cover가 되지 않은 구역에 먼저 후보 거치대를 배정한다. 이 단계에서 추가적으로 1) 상업지역 또는 2) 주거지역 또는 3) 빌딩연면적이 일정 이상인 또는 3) 거주 인구가 일정 이상인 거치대 만을 추가적인 조건으로 후보군을 선정하여 후보군의 범위를 좁혔다. 상업지역 뿐만 아니라 빌딩 연면적이 일정 이상인 지역을 고려함으로써 상업 지역을 나타내는데 힘을 실어주고자 했다. 주거지역 뿐만 아니라 거주 인구의 조건을 추가한 이유는 앞서 EDA의 결과로부터 주거 지역에 해당하는 부분만을 고려할 경우, 모든 주거 지역을 대변할 수 없다는 결론을 도출했기 때문이다.

### 3. 접근성을 고려한 후보군 선별

- 선정된 후보군은 사용자의 편의성을 위해 접근성을 고려해야 한다.
- 앞선 EDA의 결과로부터, 접근성이 좋지 않은 후보군을 제거한다.
- 발생하는 각각의 후보군을 최종적으로 합집합 처리하여 예상 후보군을 만든다.

## 3.1 도로 근처 후보군 추출

사람들이 지나가는 도로를 시군구 도로로 제한하였다. 이는 EDA의 결과로 부터, 시군구 도로가 고양시의 주거지역, 상업지역 모두를 통과하는 Code라는 것을 확인했기 때문이다. 다른 조건을 기반으로 수행하지 않은 이유는, 시군구 도로가 고양시의 대부분의 도로의 특징을 모두 대변했기 때문이다. 가장 넓은 Coverage를 갖는 도로로부터 후보군을 추출함으로써 후보군의 일반화를 도모하고자 했다.

도로 주변 후보군을 고르는 간격은 300m를 기준으로 추출했다. 이 이유는 천천히 걸을 때 사람의 평균 속도를 시간당 3.5km라 가정했을 때, 5분 동안 갈 수 있는 거리이기 때문이다.

도로의 경계를 뽑기 위해서 도로의 폭을 고려했다. 제공된 데이터를 사용하고자 하였으나, 데이터의 문제가 있다는 것을 확인하고 기본적으로 1m를 기준으로 확장하여 판단하였다. 후에 도로 근처에 이를 옮김으로써 해결한다.

## 3.2 미배치 지역 우선 배정

공공성을 고려하여 현재 배치되지 않은 지역을 기반으로 후보군을 추출하였다.

1. 후보군 df에 미배치지역 주거인구를 merge한다.
2. 이 중 미배치 지역에 속하지 않는 후보군(미배치 지역 주거인구 = 0)을 제거한다.

3. 미배치 지역 중 너무 적은 인구가 거주할 경우(미배치 지역 주거인구 = 200) 제거한다.
4. 현재까지 제작된 후보군에 200m안 거주 건물 수의 평균을 더해준다.
5. 주거 지역 (house\_count\_mean), 빌딩 연면적, 거주인구 일정 이상의 필터를 적용한다.
6. 후보군 df에서 상업 지구에 해당하는 후보군을 추출한다.
7. 5번의 필터를 적용한 df에 6번을 추가한다.

### 3.3 접근성을 고려한 후보군 선별

접근성은 아래와 같이 총 7개의 filter를 적용했다.

- 3.3.1 버스정류장 수
- 3.3.2 버스정류장 승하차 평균
- 3.3.3 지하철 수
- 3.3.4 지하철 승하차 평균
- 3.3.5 학교 수
- 3.3.6 문화 시설
- 3.3.7 고도

### 3.4 결론

우리는 굉장히 추상적인 후보군 추출을 앞선 EDA의 결과를 종합하여 상당히 가능성이 높은 후보군을 다수 추출하였다. 해당 후보군을 추출하는데 고려한 점은 다음과 같다.

1. 미배치 지역
2. 상업지역
3. 거주지역
4. 빌딩 연면적
5. 주거 인구
6. 주변 버스 정류장 개수
7. 주변 버스 정류장 승하차 인원
8. 주변 지하철 역 개수



9. 주변 지하철 역 승하차 인원
10. 주변 교육시설 개수
11. 주변 체육시설 개수
12. 고도

## 4. 수요 예측 모형

제안된 후보군의 수요를 예측하기 위한 모델을 제작한다.

### 4.1 수요 정의

자전거 공유 서비스를 사용하는데 있어서 수요는 다양한 측면에서 바라볼 수 있다.

이 때, 우리는 이 수요라는 것을 총 두가지 관점에서 바라보았다.

첫번째, 거치대 반입량, 반출량의 절대적인 양을 사용하였다. 이 지표는 하나의 거치대에서 방문하는 사용자의 총량을 대변한다.

두번째, 거치대 반입량, 반출량의 차이를 사용하였다. 이 지표는 하나의 거치대에서 유입, 유출되는 자전거의 변화량을 의미한다.

이 때, 우리는 이러한 두가지 지표의 시간 단위를 일로 설정하였다.

결과적으로 사용하는 수요의 개념은 다음과 같다.

- 용어 정의
  - 총 자전거 이용량 (반입량+반출량)의 일별 평균 : 절대 수요
    - $\$(\text{return} + \text{lease}) / \text{totalDay}\$$
  - 총 자전거 변화량 (반입량-반출량)의 일별 평균 : 상대 수요
    - $\$(\text{return} - \text{lease}) / \text{totalDay}\$$

### 4.2 Modeling

#### 4.2.1 사용하는 변수

EDA결과를 기반으로 다양한 변수를 생성하여 사용하였다.

▼ I변수목록

1. 버스 정류장 개수

- 'busstop\_cnt\_200m'
- 'busstop\_cnt\_300m'
- 'busstop\_cnt\_500m'

2. 버스 정류장 승하차 합계/평균

- 'busstop\_GETON\_CNT\_sum\_200m'
- 'busstop\_GETON\_CNT\_sum\_300m'
- 'busstop\_GETON\_CNT\_sum\_500m'
- 'busstop\_GETON\_CNT\_mean\_200m'
- 'busstop\_GETON\_CNT\_mean\_300m'
- 'busstop\_GETON\_CNT\_mean\_500m'

3. 지하철 역 개수

- 'subway\_cnt\_500m'
- 'subway\_cnt\_1000m'
- 'subway\_cnt\_1500m'

4. 지하철역 승하차 합계/평균

- 'subway\_IN\_sum\_500m'
- 'subway\_IN\_sum\_1000m'
- 'subway\_IN\_sum\_1500m'
- 'subway\_IN\_mean\_500m'
- 'subway\_IN\_mean\_1000m'
- 'subway\_IN\_mean\_1500m'
- 'subway\_OUT\_sum\_500m'
- 'subway\_OUT\_sum\_1000m'
- 'subway\_OUT\_sum\_1500m'

- 'subway\_OUT\_mean\_500m'
- 'subway\_OUT\_mean\_1000m'
- 'subway\_OUT\_mean\_1500m'
- 'subway\_NET\_TOTAL\_sum\_500m'
- 'subway\_NET\_TOTAL\_sum\_1000m'
- 'subway\_NET\_TOTAL\_sum\_1500m'
- 'subway\_NET\_TOTAL\_mean\_500m'
- 'subway\_NET\_TOTAL\_mean\_1000m'
- 'subway\_NET\_TOTAL\_mean\_1500m'

#### 5. 교육시설 개수

- 'school\_cnt\_500m'
- 'school\_cnt\_1000m'
- 'school\_cnt\_1500m'

#### 6. 체육시설 개수

- 'sports\_cnt\_500m'
- 'sports\_cnt\_1000m'
- 'sports\_cnt\_1500m'

#### 7. 고도

- 'altitude'

#### 8. 거주 인구

- 'GID\_POP'

#### 9. 200m 내 인구 합계

- 'pop\_200\_sum'

#### 10. 200m 내 건물 연면적 합계

- 'b\_area\_200\_sum'

#### 11. 200m내 주거 건물 수 평균

- 'HOUSE\_COUNTS\_200\_mean'

## 12. 상업/거주 지역 구분

- 'areatype'

## 13. 네트워크 분석을 통한 군집

- 'CLUSTER'

### 4.2.2 모형 선택

extratree, lightgbm, xgboost로 를 테스트 해보았고, 그중 가장 좋은 성능을 보인 extratree를 사용하였다. 훈련 시간 관계로 extratree만 훈련하도록 하였다.

### 4.2.3 결과

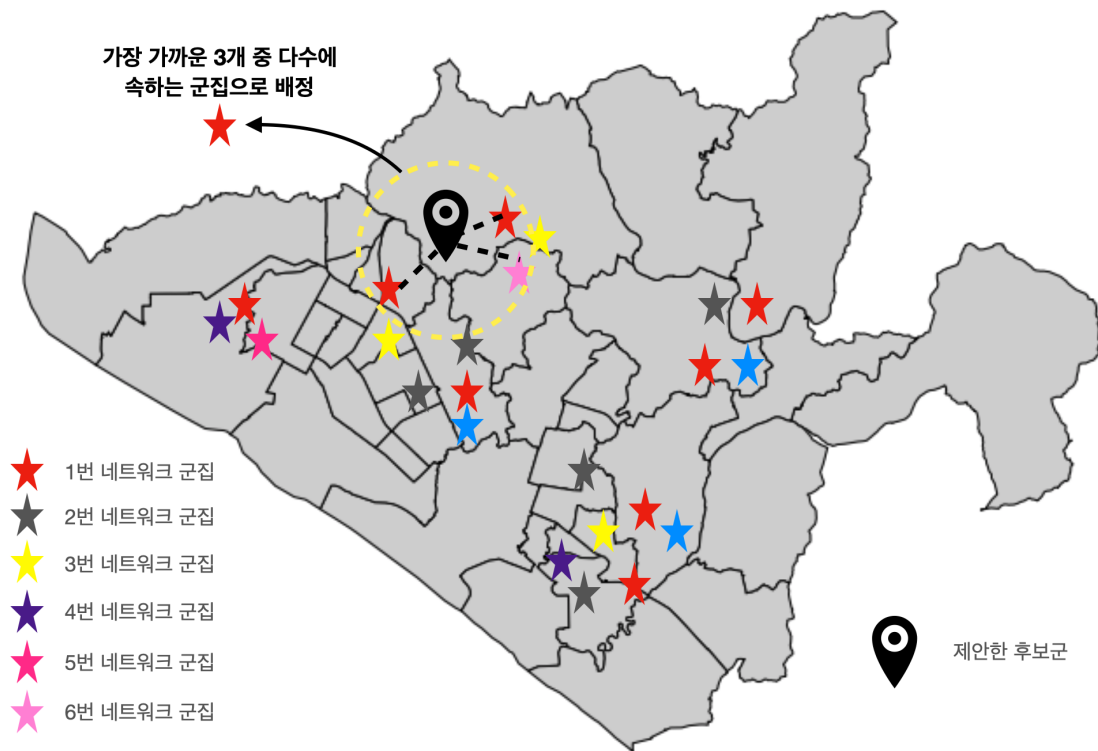
현재 설치된 거치대를 기준으로 절대수요, 상대수요를 예측한 결과 높은 성능의 모델이 도출되었다. 데이터 개수가 많이 부족하지만, 상대적으로 가벼운 모델인 decision tree 기반 모델을 사용하여 최대한 일반화 성능을 높이는 방향으로 모델링을 진행하였다.

## 4.3 후보군 수요 예측

이제 만들어진 후보군을 기반으로 훈련한 모델에 입력함으로써 절대수요와 상대수요를 예측한다.

### 4.3.1 후보군 Network 군집 배정을 위한 설정

앞서 진행한 EDA에서 네트워크 분석을 통해 거치대를 총 6개의 군집으로 나누었다. 우리가 제작한 후보군을 해당 군집에 적용한다. 이 때, KNN 알고리즘을 사용하여 유클리드 거리 기반으로 가까운 군집 번호를 부여한다.



4.3.2에서 이 클러스터를 배정해 주기 위해, 기존 거치대의 정보를 KNN 알고리즘에 설정한다.

### 4.3.2 사용하는 변수

수요 예측 모델을 훈련했을 때 사용한 모든 변수를 후보군에 붙인다.

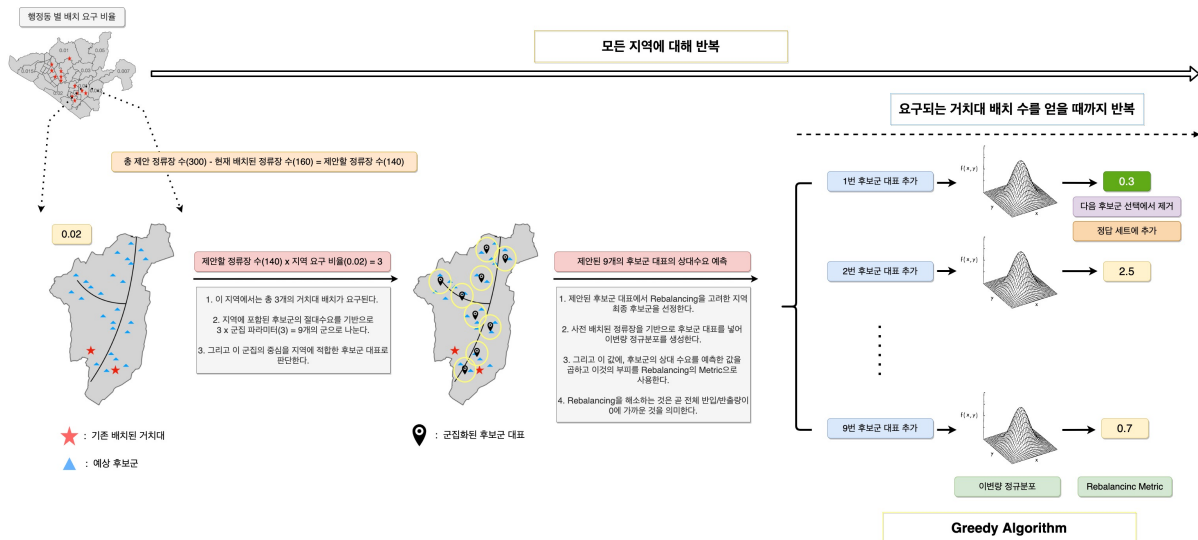
### 4.3.3 수요 예측

후보군을 기반으로 예상되는 절대 수요와 상대 수요를 구한다.

## 5. 최적 후보군 선정 알고리즘

예측된 절대 수요와 상대 수요를 기반으로 수요 증대와 Rebalancing을 고려한 최적 후보군을 선정한다.

### 5.0 전체 프로세스



해당 알고리즘을 구현하는 데 있어 고민한 점은 크게 4가지이다.

가정 : 자전거의 이동 범위는 지역을 크게 벗어나지 않는다.

1. 지역별로 상이한 거치대 배치 비율을 만들어야 한다.
2. 제작한 거치대 배치 비율을 기반으로 지역별 필요 거치대 개수를 구한다.
3. 해당 지역에 존재하는 거치대 후보에서, 절대 수요를 가장 잘 대표하는 후보군 대표를 선정한다.
4. 이렇게 발생한 후보군 대표 중, 상대 수요의 총합이 가장 작은(불균형 문제를 최소화 하는) 것들을 최종 후보군으로 선정한다.

## 5.1 지역별 요구 거치대 비율

지역별 요구 거치대 비율을 구현하는 데 있어 고민한 점은 크게 2가지이다.

1. 공공성을 고려하여 거치대수가 부족한 지역(행정동)에 우선 배정한다.  
(station\_count)
2. 수요가 밀집된 지역에 추가 배치하여 보완한다.(TOTAL\_POP)

위의 두가지 요인을 모두 고려한 **지역별 요구 거치대 비율** 이라는 변수를 생성하였다.  
위 변수를 제작한 수식은 다음과 같다.

$$normalization\left(\frac{1}{stationCount} \times populationRatio\right)$$

station\_count가 크면, 거치대는 적게 설치해야 한다. 따라서 역수 처리 하였다.  
population\_ratio가 크면, 거치대는 더 설치해야 한다. 따라서 그대로 두었다.

두 지표를 곱한 후에 이를 정규화하여 해당 지역의 요구 거치대 비율로 사용하였다.


## 5.2 Rebalancing을 고려한 최적 후보군 선정

최종적으로 예측된 절대 수요와 상대 수요를 기반으로 수요 증대와 Rebalancing을 해결하는 장소를 찾는다.

### 5.2.1 데이터 셋 제작

5.1에서 제작한 지역별 요구 거치대 비율을 기반으로 진행하기에 앞서, 이를 쉽게 반영하기 위한 데이터 프레임을 제작한다.

#### 변수 설명

<u>Aa</u> variable	 means
<u>total</u>	최종 행정동에 배치될 거치대의 총 개수
<u>exist</u>	행정동 내에 존재하는 거치대의 개수
<u>need</u>	요구 거치대 비율을 기반으로 필요한 거치대의 개수
<u>min_x</u>	해당 지역의 x 하한
<u>max_x</u>	해당 지역의 x 상한
<u>min_y</u>	해당 지역의 y 하한
<u>max_y</u>	해당 지역의 y 상한

x, y에 대한 정보는 추후 Metric을 구하는데 사용될 것이다.

### 5.2.2 절대 수요 기반 클러스터링

절대 수요를 기반으로 필요한 후보군의 개수 x 군집 변수(3)에 해당하는 개수의 클러스터링을 진행한다.

지역 별로 요구되는 거치대 개수x3에 해당하는 군집으로 후보군을 분리함으로써, 비슷한 절대 수요를 가지는 후보군을 하나로 묶어, 해당 지역의 특징을 반영한 대표군을 만들고자 하였다.

### 5.2.3 상대 수요 기반 Rebalancing

상대 수요는 간단하게 말하면, 해당 거치대에 반입량의 순수합이다. 즉 양수인 경우 반입이 많고, 음수인 경우 반출이 많다. 거치대의 이러한 특징은 시스템 운영 비용에 직접적으로 영향을 미친다. 따라서 이러한 상대 수요의 총 합이 0이 되는, 즉 Rebalancing에 도움을 주는 지역을 선정하는 것이 매우 중요하다.

## 5.2.4 Metric 정의

위에서 제작한 대표군은 최종적으로 배치될 지역 후보군이다.

하지만 공유 자전거 문제의 특징상 하나의 새 거치대 배치는 지역내에 존재하는 다른 모든 자전거에 영향을 미친다. 이러한 의존성이 있는 문제에 대한 해답으로 우리는 상대 수요를 기반으로 한 이변량 정규분포의 부피로 제안한 거치대의 Rebalancing 영향도를 측정하였다.

$$metric = \int_{y_{min}}^{y_{max}} \int_{x_{min}}^{x_{max}} |\Phi(x_1, x_2)|$$
$$\Phi(x_1, x_2) = \sum_i^n \alpha_i * \left( \frac{\exp^{-\frac{1}{2}(x-\mu_i)^t \Sigma_i (x-\mu_i)}}{\sqrt{2\pi|\Sigma_i|}} \right)$$

여기서  $\sum \alpha$ 는 현재 제안된 후보군을 포함한 상대 수요의 총합이다. 이렇게 정의한다면, 이 Metric 함수(= 상대 수요의 총합 x 이변량 정규 분포)의 부피는 지역 전역에 있는 상대 수요의 총량을 대변한다고 할 수 있다. 이러한 지표를 사용하였을 때, 상대 수요의 총량이 0이 되도록 하는 대표 후보군을 선택하는 방법으로 최적 거치대를 선정하였다. 해당 알고리즘에서는 시간 복잡도 문제로 Greedy알고리즘을 사용하였다.

## 5.2.5 pseudo-algorithm

```
Set 지역, 지역 비율, 이미 배치된 수, 총 배치대 수
Set 후보군, 지역 후보군, 지역 후보군 절대 수요
Set 배치된 거치대
Set 최종 후보군

for 지역:
    추가 배치대 수 = 지역 비율 * 총 배치대 수 - 이미 배치된 수
    클러스터 수 = 3(군집 파라미터) * 추가 배치대 수
    클러스터 = Kmeans(cluster = 클러스터수, weight = 지역 후보군 절대 수요)
    지역 후보군 = 클러스터.중심

    for 추가 배치대 수:
        최소 리밸런싱 = INF
        for 지역 후보군:
            배치된 거치대.push_back(지역 후보군[i])
            최소 리밸런싱 = MIN(최소 리밸런싱, Metric(배치된 거치대))
            배치된 거치대.pop_back(지역 후보군[i])
        배치된 거치대.push_back(지역 후보군[min_index])
        지역 후보군.remove(지역 후보군[min_index])
    END
END

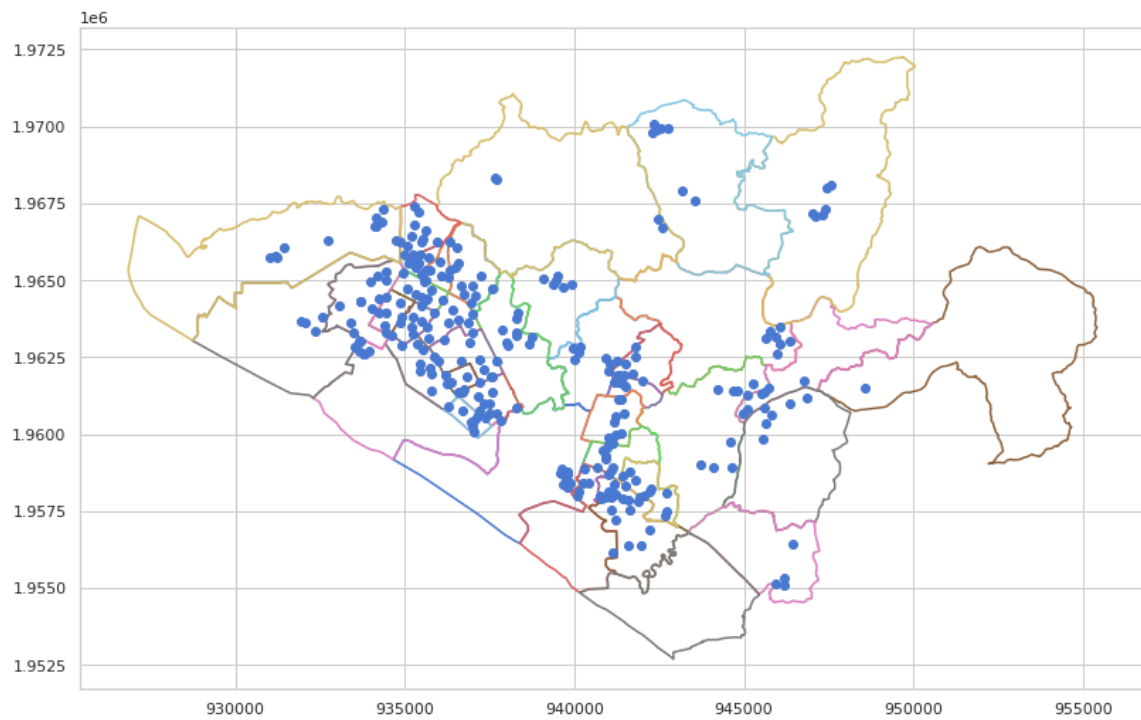
최종 후보군.push_back(배치된 거치대)

END
```

## 5.3 결과 시각화/해석

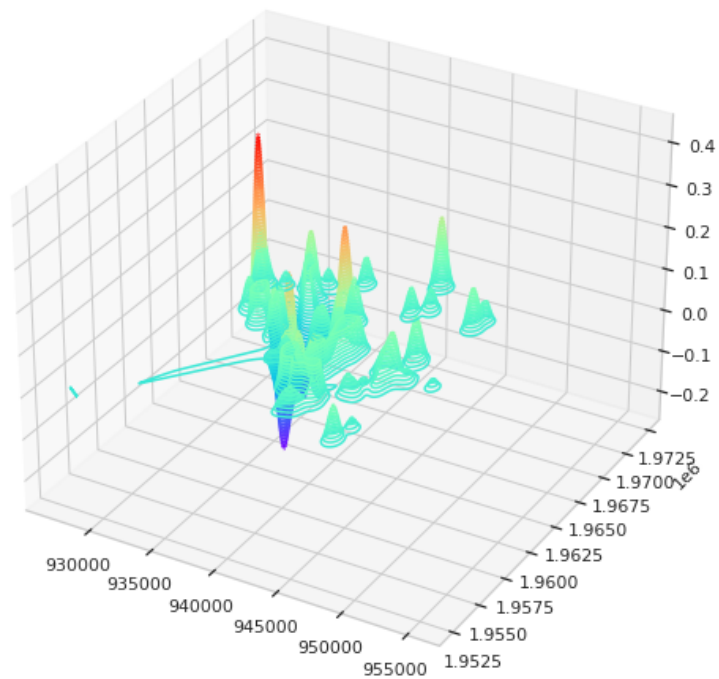


- Rebalancing을 해결하는 최적 후보군 시각화



- 고양시 전체 Rebalancing 정도

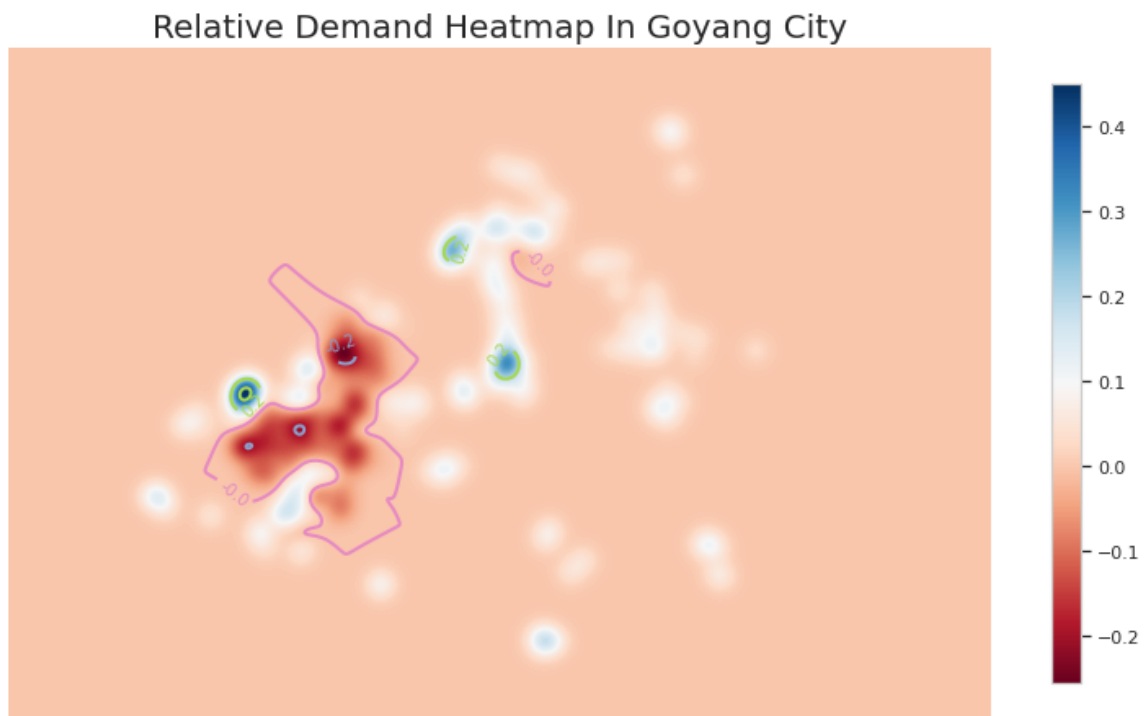
Relative Demand 3D Plot



극단적으로 치운친 봉오리가 많이 없는 것으로 보아 리벨런싱이 잘 되었다고 생각할 수 있다.

하지만 중간에 극단적인 부분이 있다. 이 부분을 확대해서 관찰할 필요성이 있다.

- 후보군 포함 상대 수요 히트맵

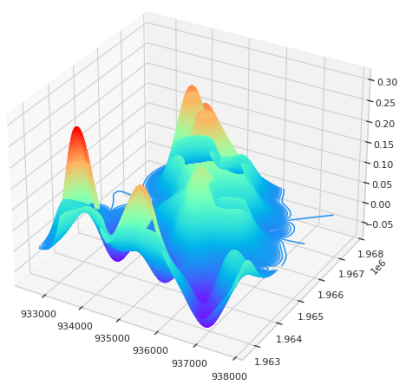


중간에 극단적인 부분이 호수공원, 킨텍스, 대화역, 주엽역, 라페스타, 웨스턴돔 부분인 것을 확인했다.

해당 부분만 떼어내서 관찰해보자.

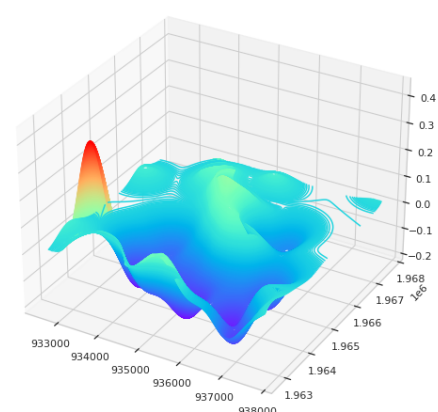
- 후보군 미포함 plot

Relative Demand 3D plot Of Minus Region With Installed Station



- 후보군 포함 plot

Relative Demand 3D plot Of Minus Region With Candidates



후보군을 포함하지 않았을 때, 상대 수요의 분포를 보면 굉장히 들쭉날쭉함을 알 수 있다. 제안한 후보군을 추가한 이후 그래프가 완만해졌고, 이는 곧 어느정도 불균형을 해소했음을 의미한다.

하지만 여전히 거치대에 따른 상대 수요의 불균형이 존재하고, 이를 해결하기 위해 자전거 운송 차량을 배치할 필요성이 엿보인다.

## 5.4 위치 보정/최종 답안 제작

마지막으로 제안된 후보군의 위치를 보정해주는 작업을 진행한다.

가장 가까운 도로로 후보군들을 이동시킨다. 그리고 마지막으로 답안 양식으로 후보군을 추가한다.

- 후보군 도로 근처 이동
- 자전거 거치대 용량 결정

거치대 용량의 평균이 26인 것을 확인했다.

우리는 이 거치대 용량의 평균에 맞는 숫자를 배정하려 했다.

결과적으로 상대 수요를 기반으로 정렬하고 이를 3등분하여 순차적으로 30, 25, 20 개의 거치대 수량을 배정하여 이 평균치에 맞도록 하였다.

# 6. 보완사항

## 6.1 수요 예측을 위한 데이터의 부족

기존에 설치된 거치대를 기반으로 새로운 후보군의 수요를 예측했다.

하지만 기존의 정류장 수 160개를 기반으로 이러한 수요 특성을 파악하기에는 모집단의 수가 너무 적다는 문제가 있다. 최대한 일반화 성능을 높이는 방향으로 진행했지만, 보다 많은 수의 데이터가 있었다면 더 정확한 제안을 하지 않았을까 한다.

## 6.2 전역 Rebalancing

시간 복잡도의 문제로, 지역을 기반으로 Rebalancing을 해결하고자 하였다. 우리가 선택한 Metric은 근본적으로 3차원 그래프의 부피를 구하는 작업이다. 이 작업은 굉장히 많은 연산량을 요구한다. 따라서 지역의 Rebalancing을 최소화하는 후보군을 선택하고, 이것들의 모임을 전체 고양시의 거치대의 불균형 문제를 해결하는 답안으로 제출했다. 더 좋은

Metric을 설정하거나 빠른 연산이 가능하다면 전역 Rebalancing을 해결하는 답안을 찾고 싶다.

## 6.3 Greedy Algorithm

지역적으로 후보군을 찾는데 있어서 역시 완전 탐색을 통한 최적 후보군을 찾지 못했다. 이 역시 연산량에서 근간한 문제로, 완전 탐색으로 대표 후보군  $3N$ 개로부터  $N$ 개의 최적 후보군을 찾기 위해서는  $O(n!)$  정도의 시간 복잡도를 요구한다. 또한 각각의 Metric을 구하는 것도 6.2에서 말했듯 굉장히 높은 연산량을 요구하기 때문에 이를 시도하지 못했다.

## 6.4 Rebalancing 차량 정보 부족

최적 입지를 선정하는데 있어서 수요만을 고려한다면, 우리가 진행한 평균 수요를 기반으로 한 방법은 매우 합리적이다. 하지만 Rebalancing 문제에 있어서 평균 수요를 가지고 이를 고려하는 것은 정적인 불균형 문제만을 해소하기 때문에 약간의 부족함이 있다. 만약 시간대 별로 자전거 재배치 차량의 동선, 수량 등에 대한 데이터가 있었다면, 좀더 나은 Rebalancing 해결 방안을 제시할 수 있지 않았을까 하는 아쉬움이 남는다.