

# 서울의 대중교통 사각지대 분석과 PM 도입 지역 선정

빅데이터 분석 서비스 개발\_박성희  
빅데이터 분석 서비스 개발\_박주희  
빅데이터 분석 서비스 개발\_박성준  
빅데이터 분석 서비스 개발\_이동규

# ‘삼박이’ 팀소개



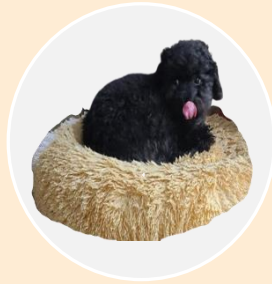
박성희  
교차분석

팀장



박성준  
데이터 시각화 및 군집분석

팀원



박주희  
회귀분석

팀원



이동규  
API 분석

팀원



# 프로젝트 수행 도구



GIS



Excel

# 데이터

DATA  
공공데이터포털  
.GO.KR



서울 열린데이터 광장  
SEOUL OPEN DATA PLAZA



OPEN API



# 일정관리

	작업 이름	완료율	기간	시작	완료	선행 작업	자원 이름
0	☐ 서울의 대중교통 사각지대 분석과 PM 도입 지역 선정	0%	21 일	20-10-24 (토)	20-11-17 (화)		
1	☐ 주제 선정/기획	0%	5 일	20-10-24 (토)	20-10-29 (목)		
2	📅 주제 선정	0%	1 일	20-10-24 (토)	20-10-25 (일)		박성준,박성희,박주희,이동규
3	📅 기획안 작성	0%	1 일	20-10-26 (월)	20-10-26 (월)	2	박성준,박성희,박주희,이동규
4	필요한 데이터 수집	0%	3 일	20-10-27 (화)	20-10-29 (목)	3	박성희,박성준,박주희,이동규
5	주제 발표 준비 및 일정수립	0%	0 일	20-10-29 (목)	20-10-29 (목)	4	박성준,박성희,박주희,이동규
6	☐ 데이터 수집 및 분석	0%	7 일	20-10-30 (금)	20-11-07 (토)		
7	📅 자차 대비 대중교통 이용시간 긴 지역 선정	0%	7 일	20-10-30 (금)	20-11-07 (토)	5	박성희
8	📅 생산가능인구(20세-60세)	0%	7 일	20-10-30 (금)	20-11-07 (토)	5	박성희
9	📅 지하철 출입구와의 거리	0%	7 일	20-10-30 (금)	20-11-07 (토)	5	박성준,이동규
10	📅 자전거도로의 길이	0%	7 일	20-10-30 (금)	20-11-07 (토)	5	박주희,박성희
11	📅 근접한 지하철 역까지 접근성이 안좋은 지역 선정	0%	7 일	20-10-30 (금)	20-11-07 (토)	5	박성준
12	📅 대중교통 이용객수	0%	7 일	20-10-30 (금)	20-11-07 (토)	5	박성준,이동규
13	📅 공공자전거 이용자 수 대비 거치대 수	0%	7 일	20-10-30 (금)	20-11-07 (토)	5	박주희
14	☐ 데이터 분석 및 검증	0%	4 일	20-11-09 (월)	20-11-12 (목)		
15	📅 sample 테스트로 데이터 검증	0%	3 일	20-11-09 (월)	20-11-11 (수)	13	
16	📅 PM 도입이 필요한 지역 선정	0%	2 일	20-11-10 (화)	20-11-11 (수)	13	박성준,이동규
17	📅 Google API를 활용한 데이터 시각화	0%	2 일	20-11-11 (수)	20-11-12 (목)	13	박성희,박주희
18	☐ PM 도입 지역에서 얻을 수 있는 개선효과	0%	1 일	20-11-14 (토)	20-11-14 (토)		
19	📅 Wordcloud 데이터 시각화	0%	1 일	20-11-14 (토)	20-11-14 (토)	17	박성준,박성희,박주희,이동규
20	☐ 분석 결과 도출 및 발표 준비	0%	4 일	20-11-14 (토)	20-11-17 (화)		
21	📅 데이터 분석과 시각화를 통한 결론 도출	0%	2 일	20-11-14 (토)	20-11-15 (일)	17	박성준,박성희,박주희,이동규
22	📅 PPT 제작	0%	2 일	20-11-16 (월)	20-11-17 (화)	21	이동규
23	📅 발표 준비	0%	2 일	20-11-16 (월)	20-11-17 (화)	21	박성희,박성준,박주희
24	📅 발표	0%	0 일	20-11-17 (화)	20-11-17 (화)		



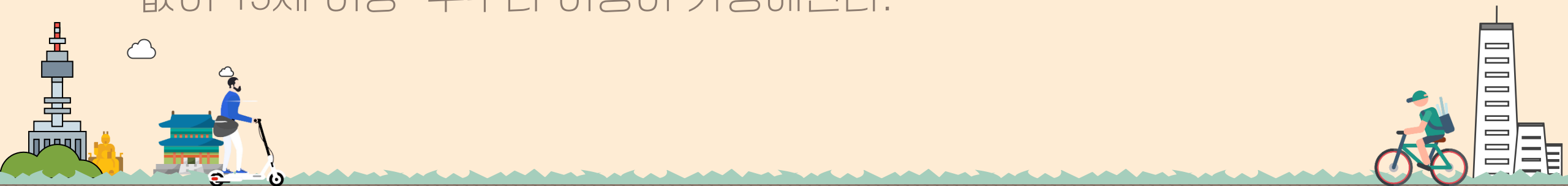
# PM(Personal Mobility)

## 개념

전기를 동력으로 하는 1인용 이동수단으로  
전동 휠, 전동 킥보드, 자전거, 초소형 전기차 등이 이에 해당된다.

## 개인형 이동 장치 관련 법 개정

2020년 12월에는 도로교통법이 개정됨에 따라 최고 속도 시속 25km 미만,  
총 중량 30kg 미만인 개인형 이동 장치에 한해 자전거 도로의 통행과 운전면허  
없이 13세 이상 누구나 이용이 가능해진다.



# Background



## 연구 범위

공간적 범위 : 서울시의 424개 행정동  
시간적 범위 : 2019년, 2020년



## 연구 목적

PM 서비스가 교통문제를 분담에 기여할 때, 도입지 결정을 위한 방법론 연구



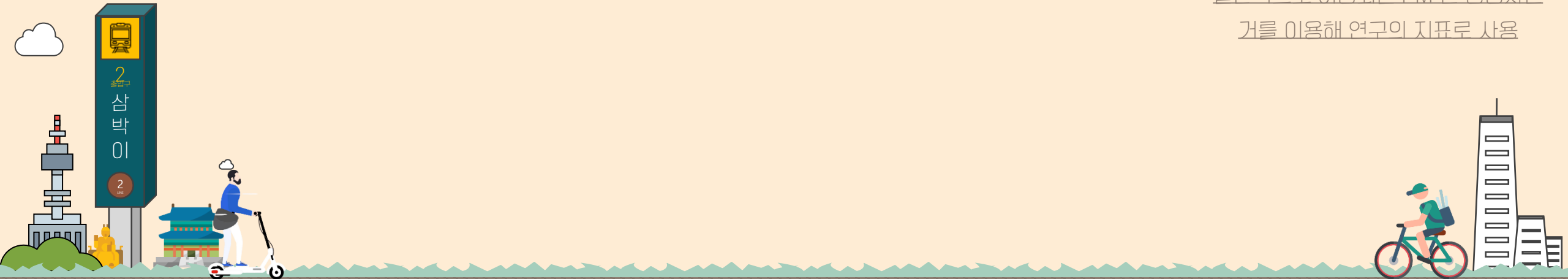
## 연구 내용

PM 서비스 입지에 대한 영향 요인별  
표준화한 수치 기반 서비스 위치선정



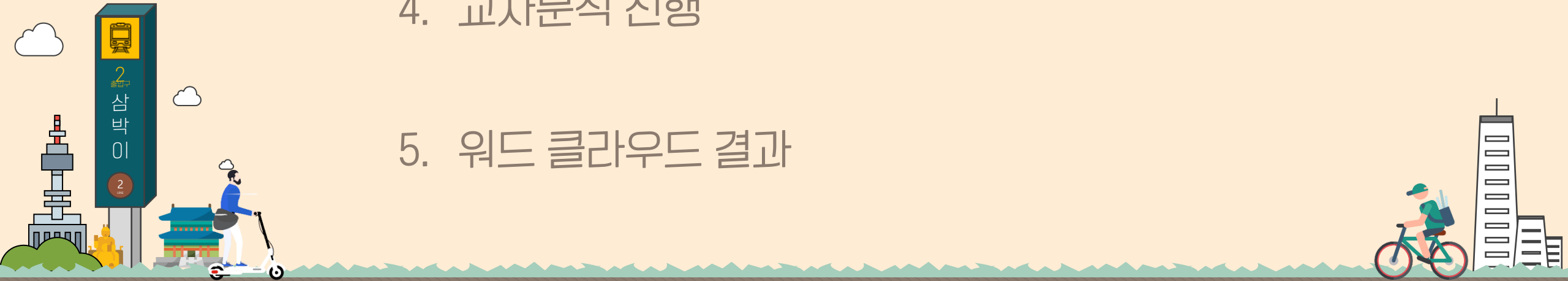
## Premise

PM 서비스가 형평성을 중요시할 경우  
의사결정에 고려할 수 있을 것으로 기대  
일반적으로 이용되는 PM 인 공공자전  
거를 이용해 연구의 지표로 사용



# 목차

1. 고려해야할 지표 선정 및 전처리, 시각화
2. 군집분석
3. 회귀분석 진행
4. 교차분석 진행
5. 워드 클라우드 결과



# 지표 선정

PM 서비스 도입에 필요한 물리적 환경요인 분석

DATA	데이터 단위	정보 단위	영향	데이터 날짜
서울특별시 공공자전거 대여소 정보	개수	행정동 단위	+	19.12.09
서울특별시 공공자전거 대여 이력	개수	링크 ID	+	19.06 ~ 20.06
국토통계 생산가능 인구수	명수	행정동 단위	+	20.04.XX
서울시 자전거 도로 링크 데이터	길이(Meter)	링크 ID	+	18.07.26
수치표고모델(DEM)	고도	격자 30M	-	15.11.--
행정동 경계	위경도	json	X	20.07.01
행정동 면적	km <sup>2</sup>	행정동 단위	X	17.12.26
지하철역 개수	개수	행정동 단위	-	20.05.19
버스정거장 개수	개수	행정동 단위	-	20.05.20
스타벅스	개수	행정동 단위	+	20.11.16
공시지가	평균(원/m <sup>2</sup> )	행정동 단위	-	20.01.--
Tmap 자차 이동시간	분	거리	-	실시간 API 수집
TOPIS 대중교통 이동시간	분	거리	-	실시간 API 수집

데이터 선정





# 전처리 과정

	동	경사도	자전거도로길이	총거차대수	총대여수	비율	인구	면적	인구밀도	지하철역개수	버스정거장개수	접근성	교통편	자전거이용비율
0	가락1동	-15.5247	7150	115	74652	64914.78	27716	1.34	20684	1	13	12.68657	17	2.69346226
1	가락2동	-39.3115	4704	45	33583	74628.89	33920	0.96	35333	0	23	23.95833	23	0.990064858
2	가락본동	-30.7845	6932	55	25487	46340	27876	1.13	24669	2	25	29.20354	33	0.914299039
3	가리봉동	-21.8088	0	10	5621	56210	15250	0.4	38125	0	9	22.5	9	0.368590164
4	가산동	-12.949	2108	258	160033	62028.29	24455	2.52	9704	2	111	47.22222	119	6.543978736
5	가양1동	-8.259	19044	948	470514	49632.28	34154	4.7	7267	4	56	15.31915	72	13.77624876
6	가양2동	-6.3928	5339	65	62588	96289.23	15140	1	15140	1	14	18	18	4.133949802
7	가양3동	-10.0878	776	33	48481	146912.1	16199	0.99	16363	1	13	17.17172	17	2.992839064
8	가회동	-59.9773	0	33	19088	57842.42	4488	0.54	8311	0	21	38.88889	21	4.25311943
9	갈현1동	-65.7546	0	15	11612	77413.33	24536	0.97	25295	0	42	43.29897	42	0.473263776
10	갈현2동	-40.7416	1688	10	9130	91300	30150	0.96	31406	0	32	33.33333	32	0.302819237
11	강일동	-22.4913	1781	105	49664	47299.05	32127	2.8	11474	0	51	18.21429	51	1.545864849
12	개봉1동	-19.8128	67	69	47097	68256.52	33835	1.31	25828	1	50	41.22137	54	1.391960987
13	개봉2동	-11.4197	2755	35	11147	31848.57	33036	0.8	41295	0	25	31.25	25	0.337419784
14	개봉3동	-18.8147	2513	15	6479	43193.33	22005	0.81	27167	0	21	25.92593	21	0.294433083
15	개포1동	-64.5348	1840	50	8208	16416	6920	1.27	5449	0	15	11.81102	15	1.186127168

서울시 행정동 단위 물리적 환경



# 전처리 과정

## 데이터 단위 표준화

	workppl	dem	bikelength	sum_road_dem	sum_ppl_dem	sum_ppl_road	sum_3
HDong_name							
가락1동	0.805473	0.674284	1.051002	1.725286	1.479757	1.856475	2.530759
가락2동	0.286642	0.071430	0.436598	0.508028	0.358072	0.723240	0.794670
가락본동	1.247918	0.287539	0.996243	1.283782	1.535457	2.244162	2.531701
가리봉동	-1.274766	0.515020	-0.744987	-0.229967	-0.759746	-2.019753	-1.504733
가산동	1.303241	0.739563	-0.215484	0.524079	2.042805	1.087757	1.827320
...	...	...	...	...	...	...	...
효창동	-0.935525	0.071483	-0.744987	-0.673504	-0.864041	-1.680512	-1.609028
후암동	-0.834567	-1.724762	-0.744987	-2.469749	-2.559328	-1.579554	-3.304315
휘경1동	-0.644024	0.667510	-0.270494	0.397016	0.023485	-0.914519	-0.247009
휘경2동	0.126148	0.599902	0.205003	0.804905	0.726050	0.331151	0.931054
흑석동	0.358674	0.240201	-0.210209	0.029992	0.598876	0.148465	0.388666

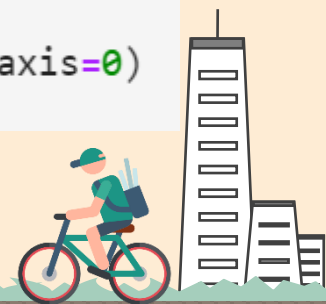
PM 서비스 도입 위치에 영향요인으로 선정된 각 요인별 단위의 표준화 과정

Z score로 환산해 인자별 단위를 통일

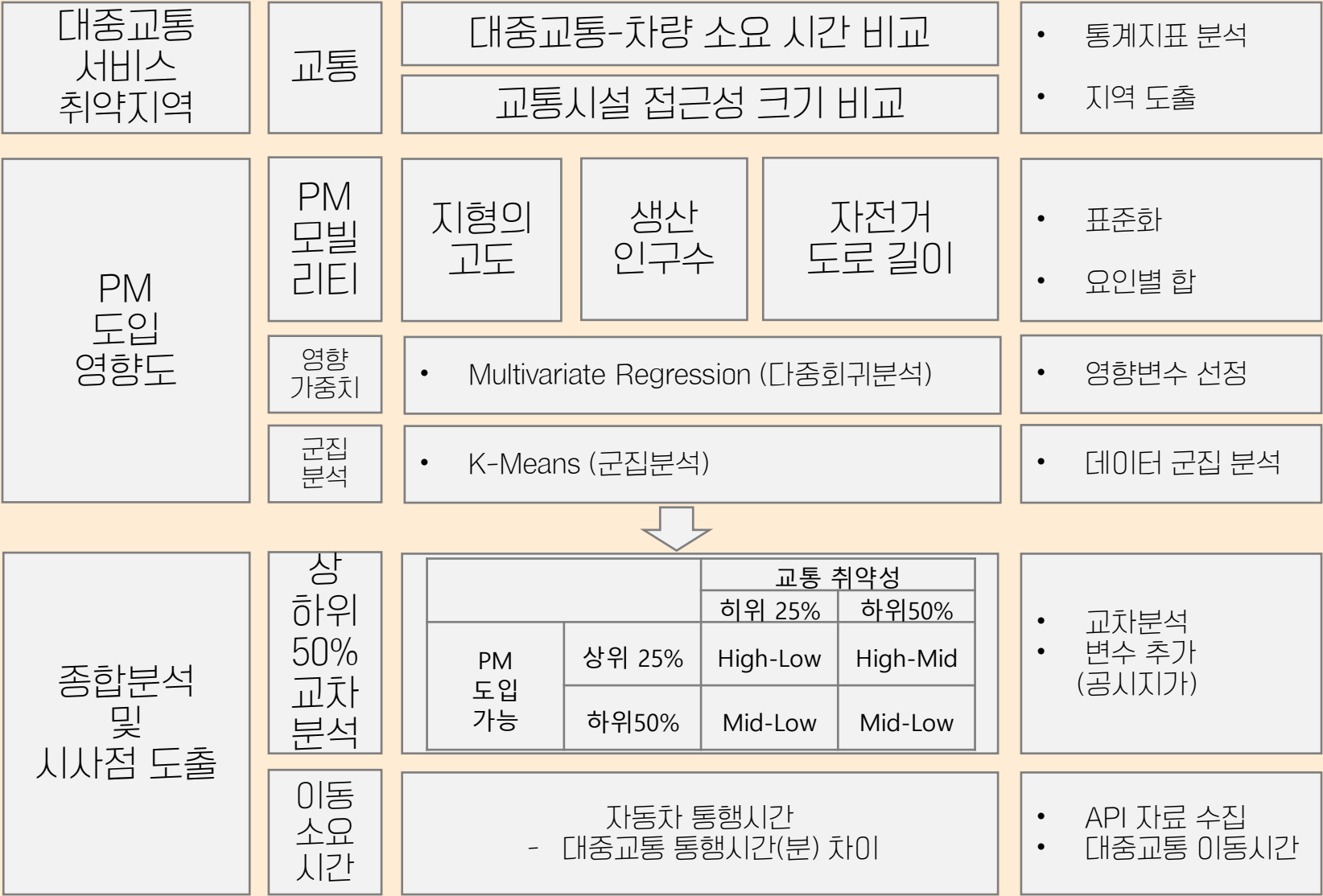
$$Z_i = \frac{x_i - \mu}{\sigma}$$

$x_i$  = 각 요인값     $\mu$  = 평균     $\sigma$  = 표준편차






```
def z_score(x):  
    return (x-x.mean())/x.std()  
# 표준화 시행  
z_dong=zscore.apply(z_score,axis=0)
```

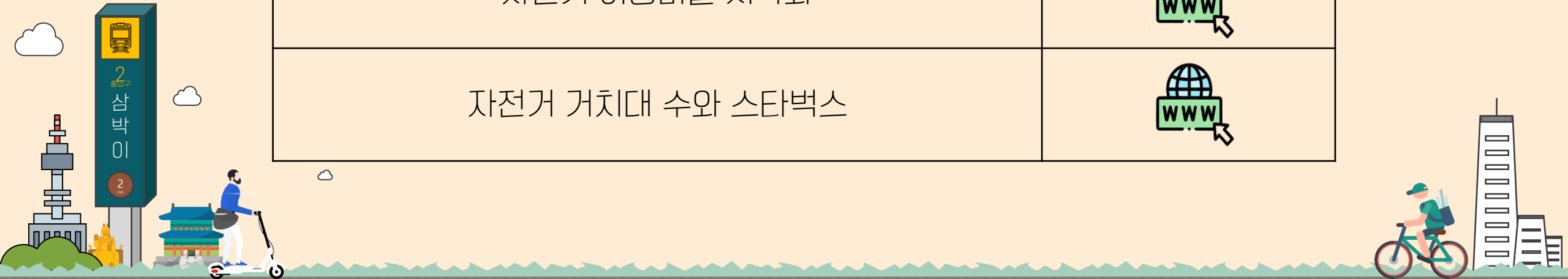


# 방법론 흐름도

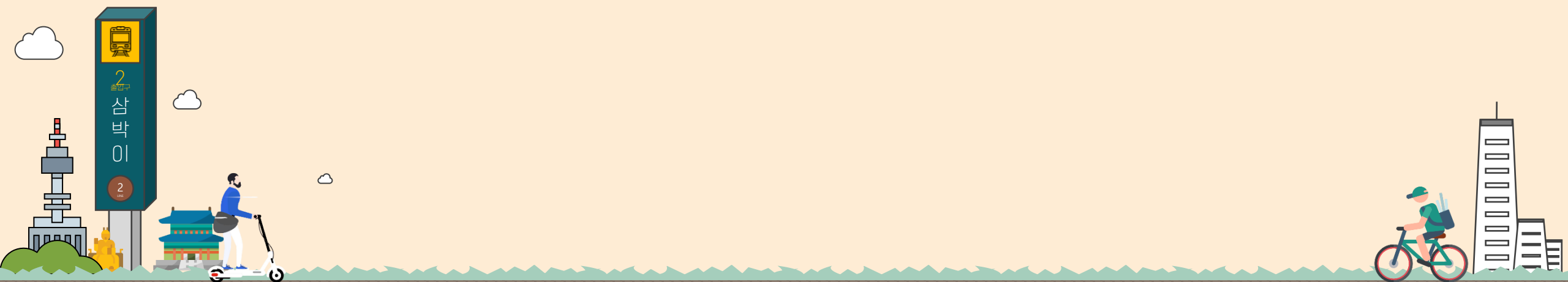


# 전처리 시각화

지하철 클러스터링과 인구밀도 시각화	
지하철과 버스 시각화	
지하철과 버스 접근성 시각화	
자전거 이용비율 시각화	
자전거 거치대 수와 스타벅스	

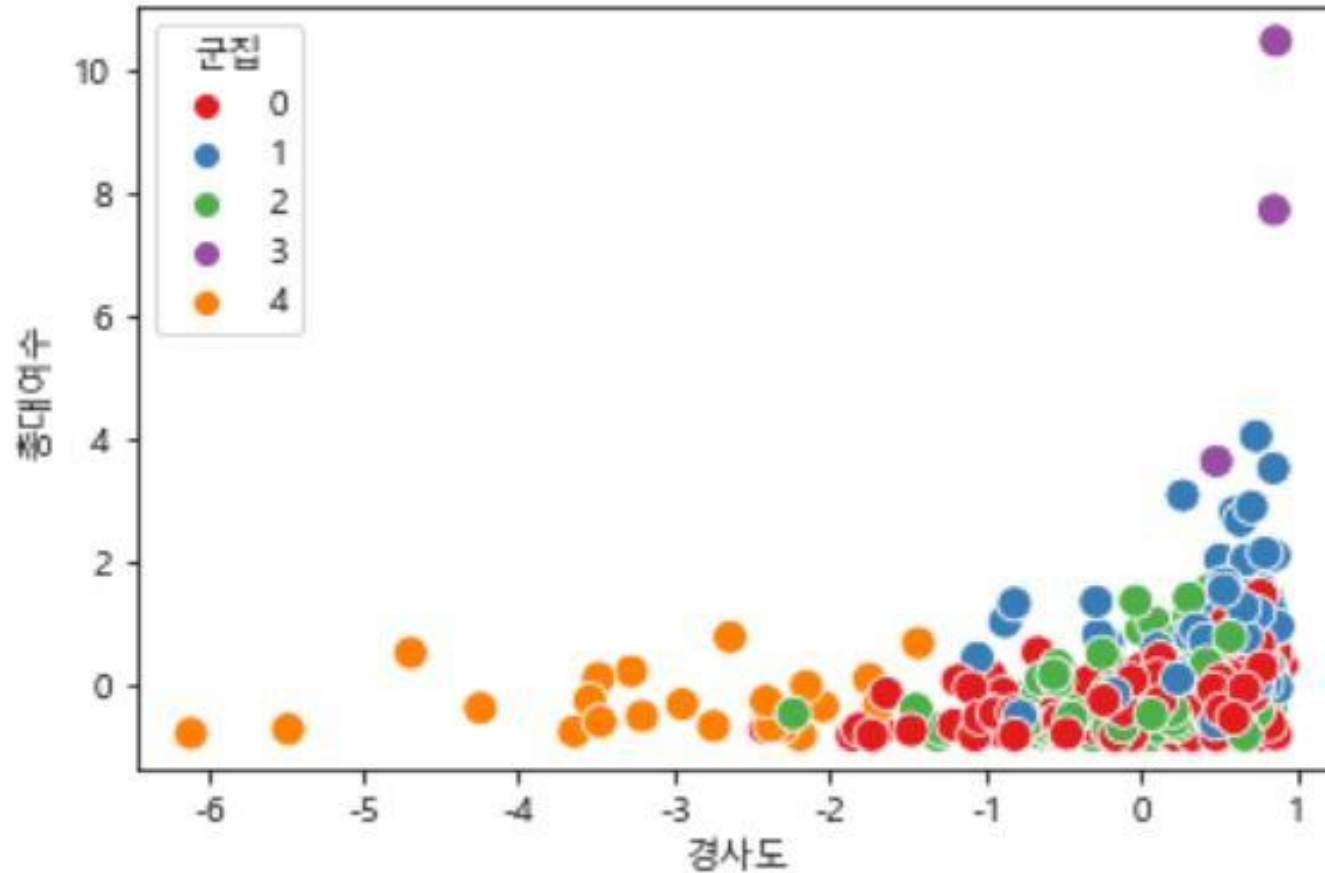


# 군집분석



# 군집분석

<AxesSubplot:xlabel='경사도', ylabel='총대여수'>

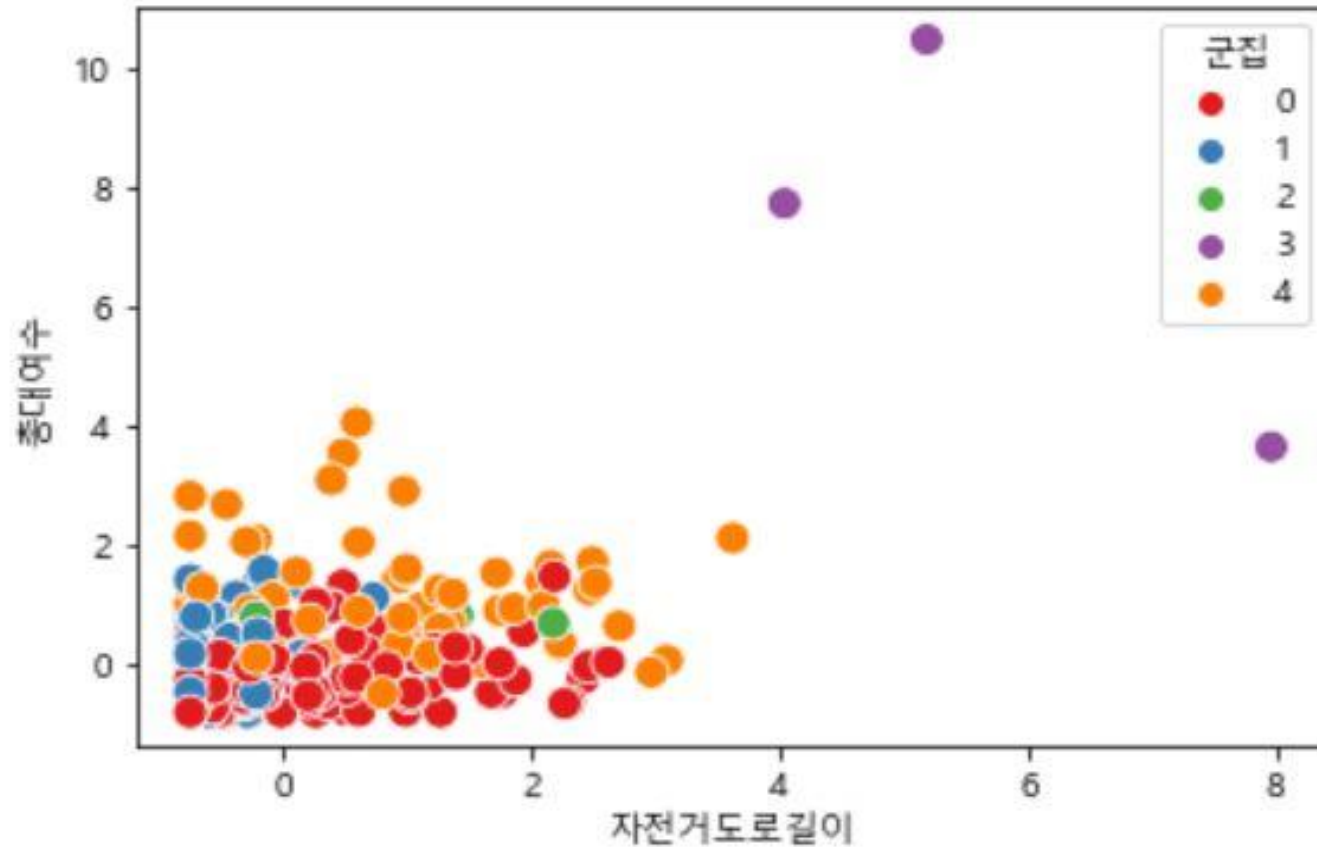


경사도와 총 대여 수 군집분석



# 군집분석

<AxesSubplot:xlabel='자전거도로길이', ylabel='총대여수'>

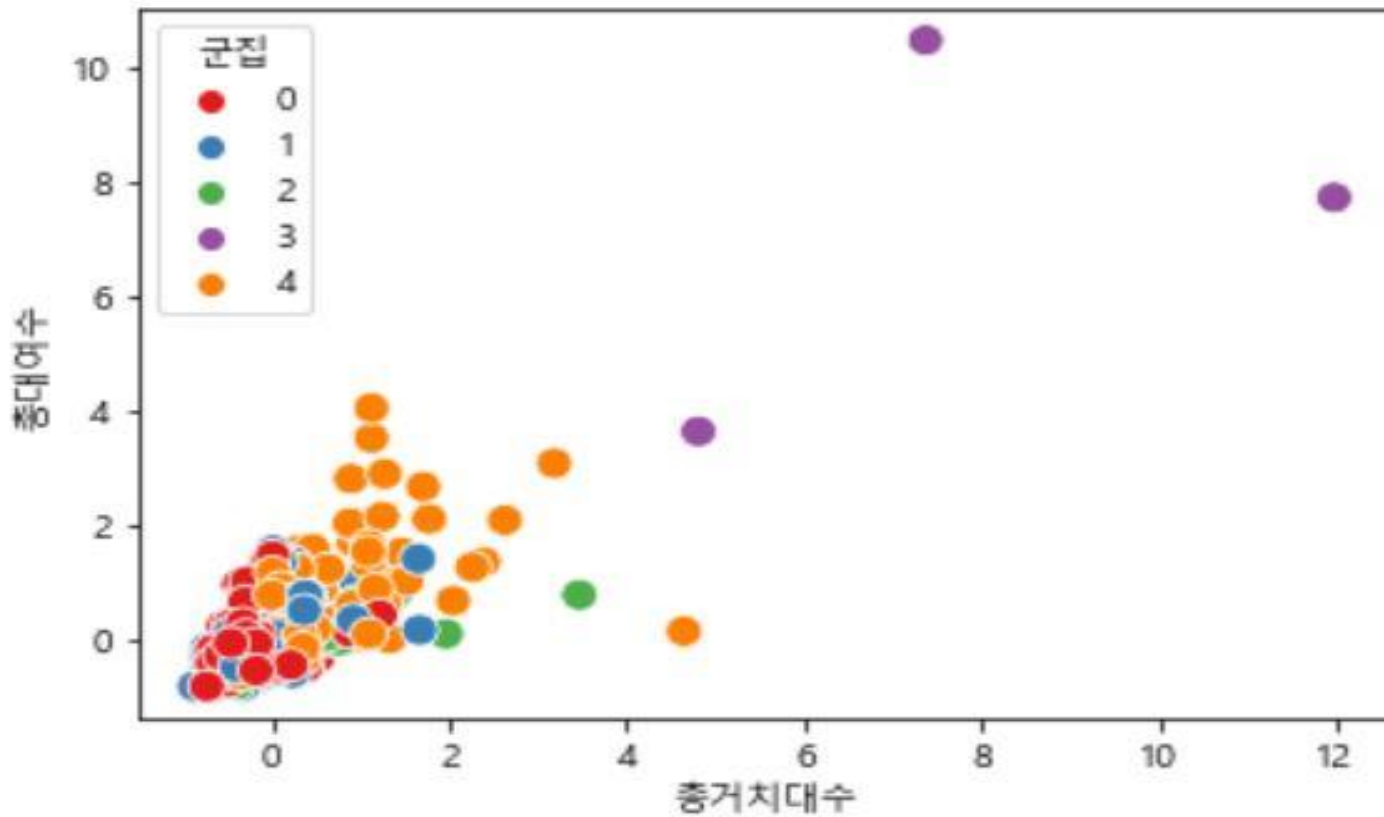


자전거도로의 길이와 총 대여 수 군집분석

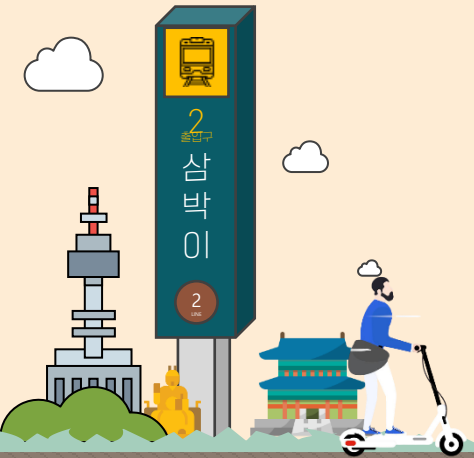


# 군집분석

<AxesSubplot:xlabel='총거치대수', ylabel='총대여수'>

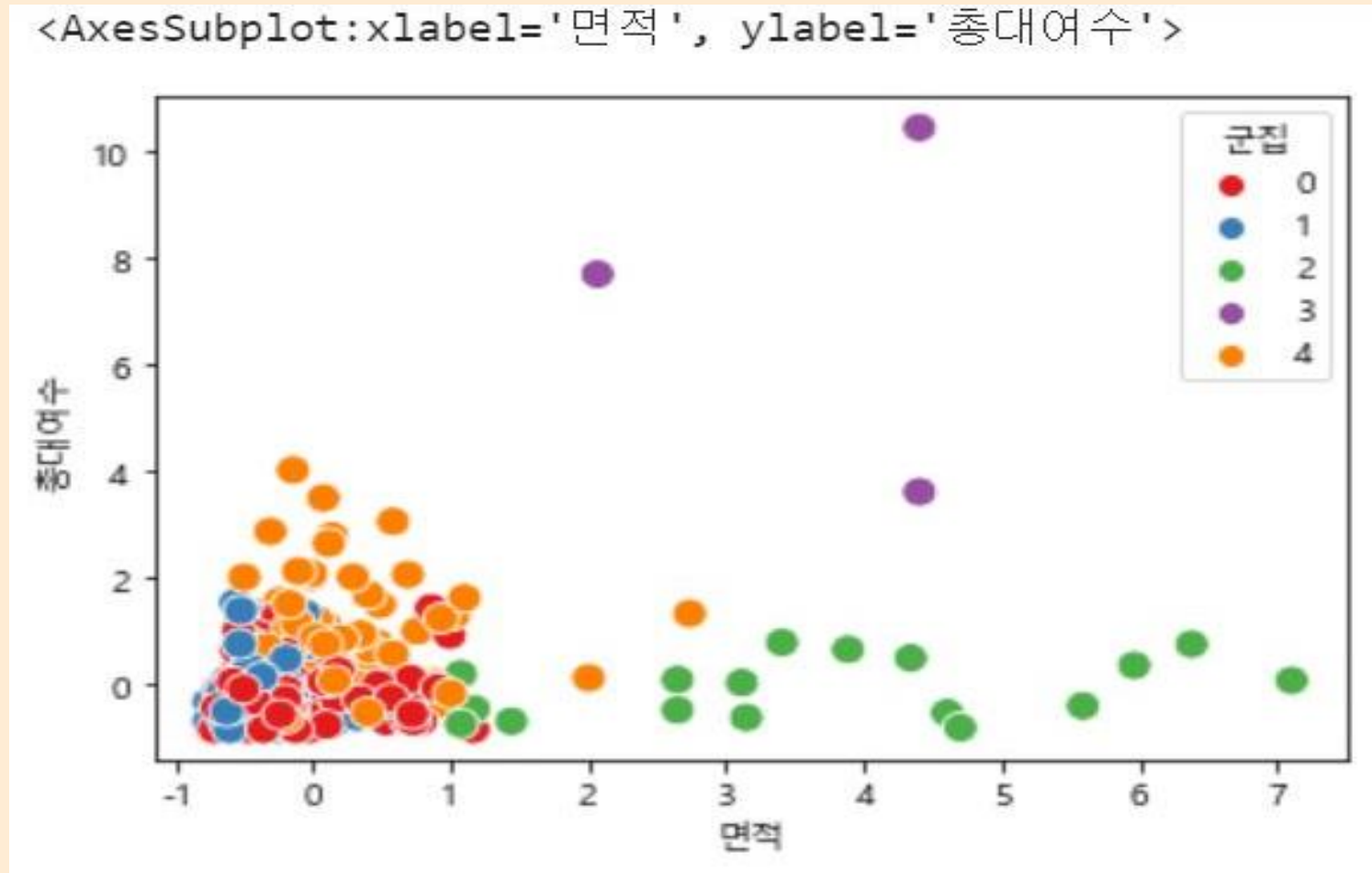


총 거치대 수와 총 대여 수 군집분석





# 군집분석

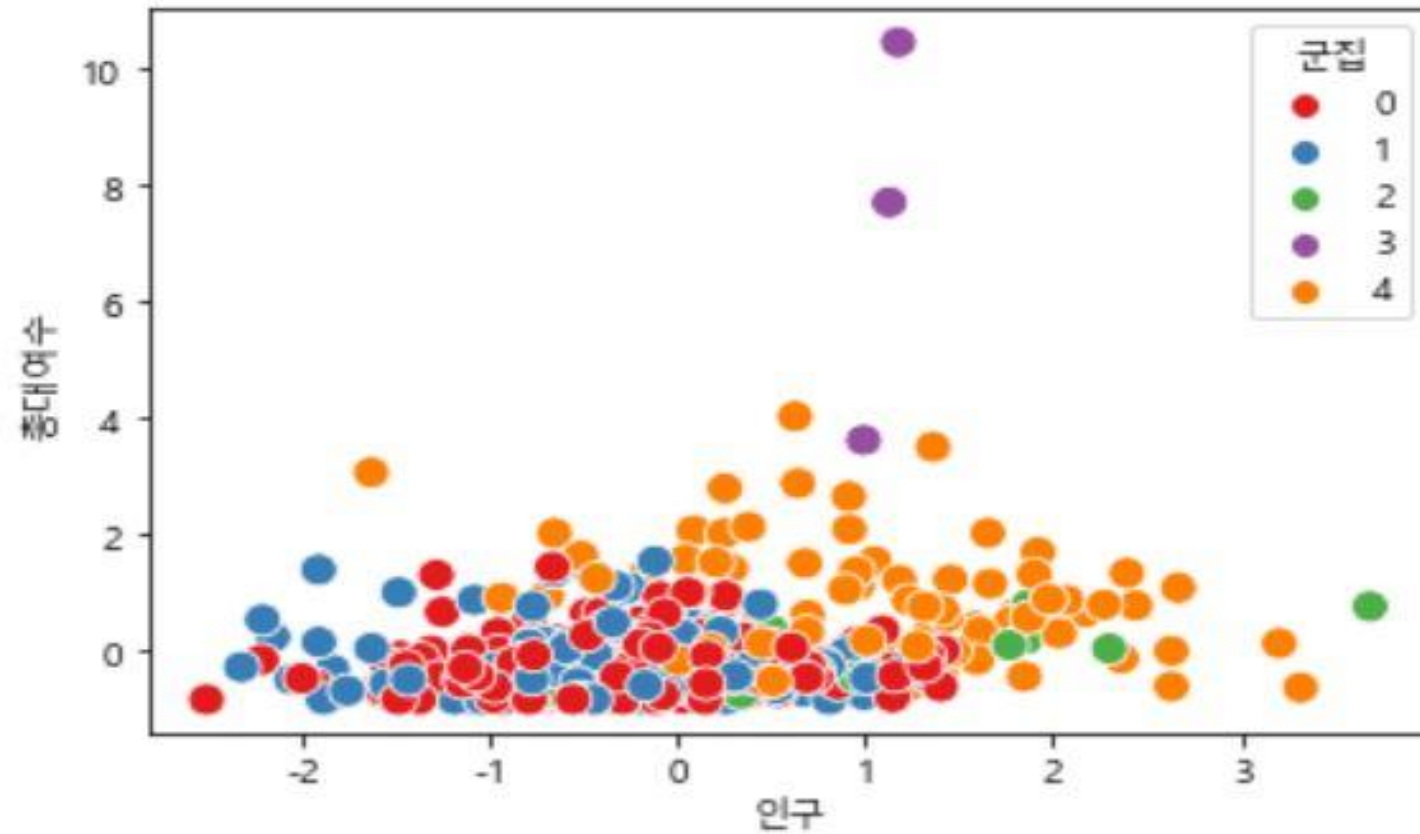


행정동 면적과 총 대여 수 군집분석

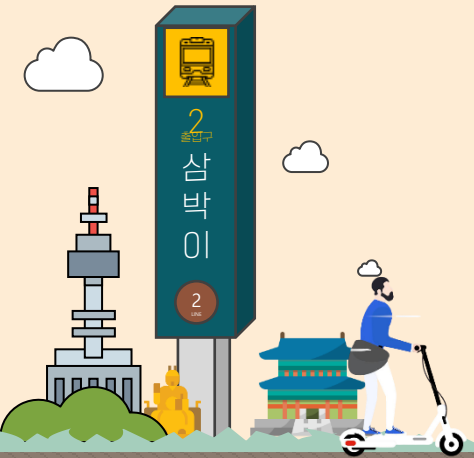


# 군집분석

<AxesSubplot:xlabel='인구', ylabel='총대여수'>

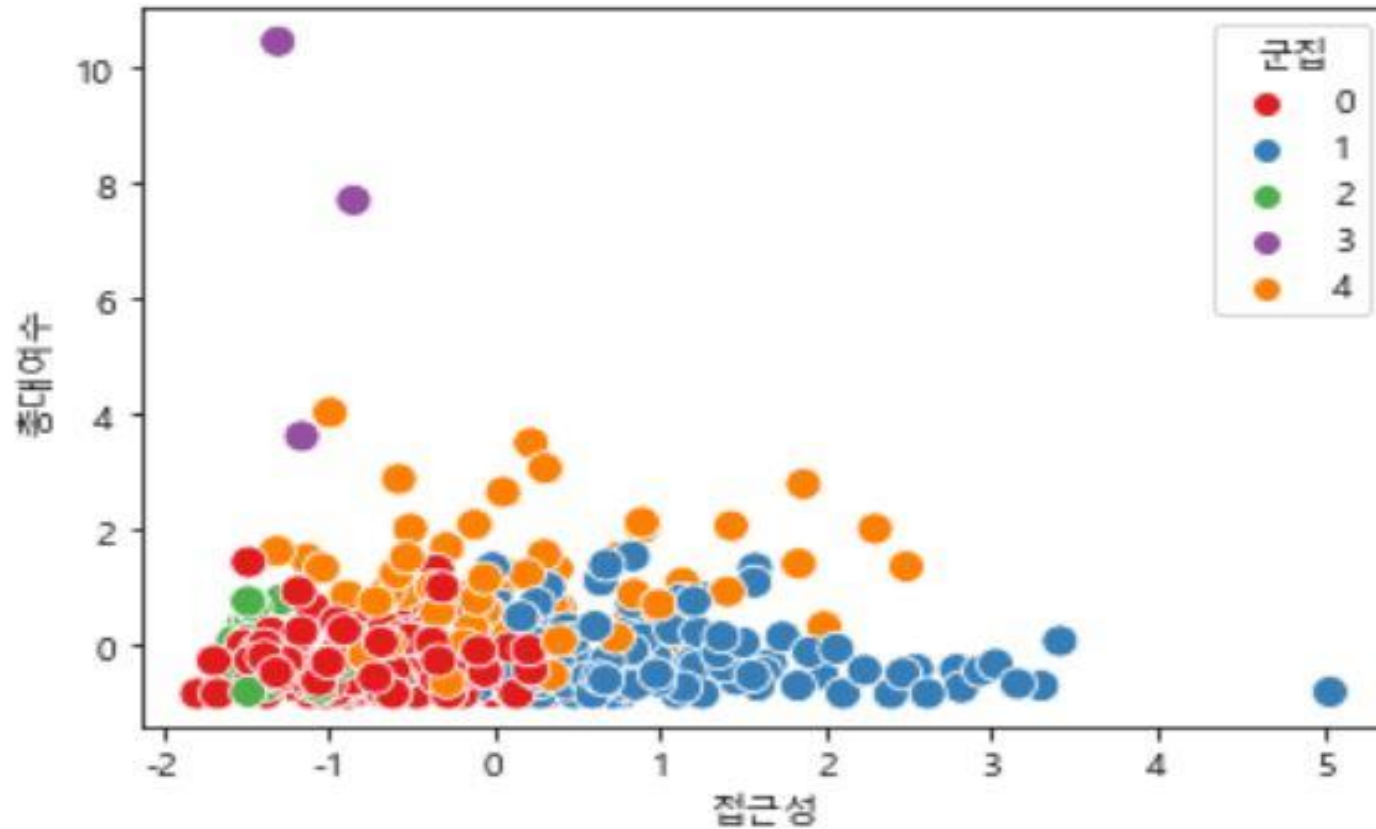


인구수와 총 대여 수 군집분석



# 군집분석

<AxesSubplot:xlabel='접근성', ylabel='총대여수'>



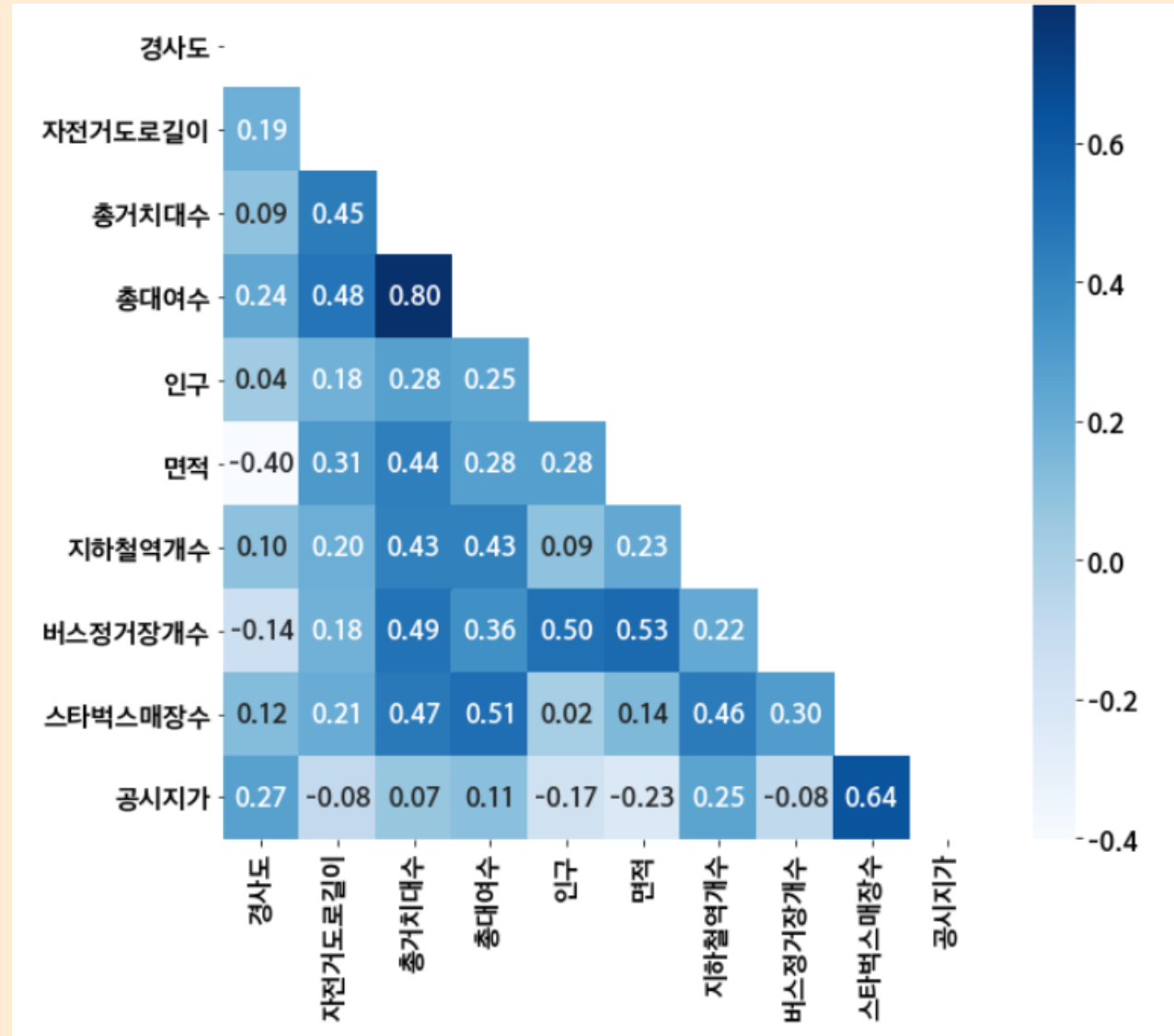
접근성과 총 대여 수 군집분석



# 다중회귀분석



# Pearson 상관분석



# 회귀분석

	경사도	자전거도로길이	총거치대수	총대여수	인구	면적	지하철역개수	버스정거장개수	스타벅스매장수	공시지가
등										
가락1동	-15.5247	7150.000000	115.0	74652.0	27716	1.34	1	13	2	1238189
가락2동	-39.3115	4704.000000	45.0	33583.0	33920	0.96	0	23	2	4597770
가락본동	-30.7845	6932.000000	55.0	25487.0	27876	1.13	2	25	2	6720623
가리봉동	-21.8088	2971.613208	10.0	5621.0	15250	0.40	0	9	0	3213180
가산동	-12.9490	2108.000000	258.0	160033.0	24455	2.52	2	111	7	2191587

## 1. 변수선택법

- 전진선택법(Forward Selection)
- 후진소거법(Backward Elimination)
- 단계적선택법(Stepwise Selection)

## 2. 다중공선성 확인

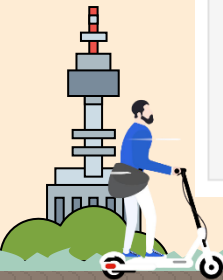


# 회귀분석 진행

## 단계적선택법

```
def Stepwise_model(X,y):
    stepModels = pd.DataFrame(columns=["AIC","model"])
    tic = time.time()
    predictors = []
    SmodelBefore = processSubset(X,y,predictors+['const'])['AIC']
    # 변수 1~10개 : 0~9 -> 1~10
    for i in range(1, len(X.columns.difference(['const']))+1):
        forwardResult = forward(X,y,predictors)
        print("forward")
        stepModels.loc[i] = forwardResult
        predictors = stepModels.loc[i]["model"].model.exog_names
        predictors = [k for k in predictors if k != 'const']
        backwardResult = backward(X,y,predictors)
        if backwardResult['AIC'] < forwardResult['AIC']:
            stepModels.loc[i] = backwardResult
            predictors=stepModels.loc[i]["model"].model.exog_names
            smodelBefore=stepModels.loc[i]["AIC"]
            predictors=[k for k in predictors if k != 'const']
            print('backward')
        if stepModels.loc[i]["AIC"] > SmodelBefore:
            break
        else:
            smodelBefore = stepModels.loc[i]["AIC"]
    toc=time.time()
    print("Total elapsed time : ", (toc - tic), "seconds")
    return (stepModels['model'][len(stepModels['model'])])
```

OLS Regression Results						
Dep. Variable:	총대여수		R-squared:	0.582		
Model:	OLS		Adj. R-squared:	0.570		
Method:	Least Squares		F-statistic:	49.88		
Date:	Mon, 16 Nov 2020		Prob (F-statistic):	5.06e-50		
Time:	14:06:45		Log-Likelihood:	-3442.6		
No. Observations:	296		AIC:	6903.		
Df Residuals:	287		BIC:	6936.		
Df Model:	8					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
총거치대수	409.2615	40.896	10.007	0.000	328.768	489.755
경사도	159.0969	46.591	3.415	0.001	67.393	250.800
자전거도로길이	1.7899	0.538	3.329	0.001	0.732	2.848
스타벅스매장수	4302.9913	1133.287	3.797	0.000	2072.383	6533.600
공시지가	-0.0022	0.001	-2.299	0.022	-0.004	-0.000
인구	0.5617	0.208	2.695	0.007	0.151	0.972
버스정거장개수	-305.3308	153.312	-1.992	0.047	-607.089	-3.573
지하철역개수	3382.5861	2240.339	1.510	0.132	-1026.993	7792.165
const	1.233e+04	6478.604	1.903	0.058	-423.312	2.51e+04
Omnibus:	90.800	Durbin-Watson:	2.096			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	578.560			
Skew:	1.079	Prob(JB):	2.33e-126			
Kurtosis:	9.500	Cond. No.	1.71e+07			



# 회귀분석 진행

```
# 성능평가  
# number of params  
print(Forward_best_model.params.shape, Backward_best_model.params.shape, Stepwise_best_model.params.shape)  
  
(9,) (9,) (9,)
```

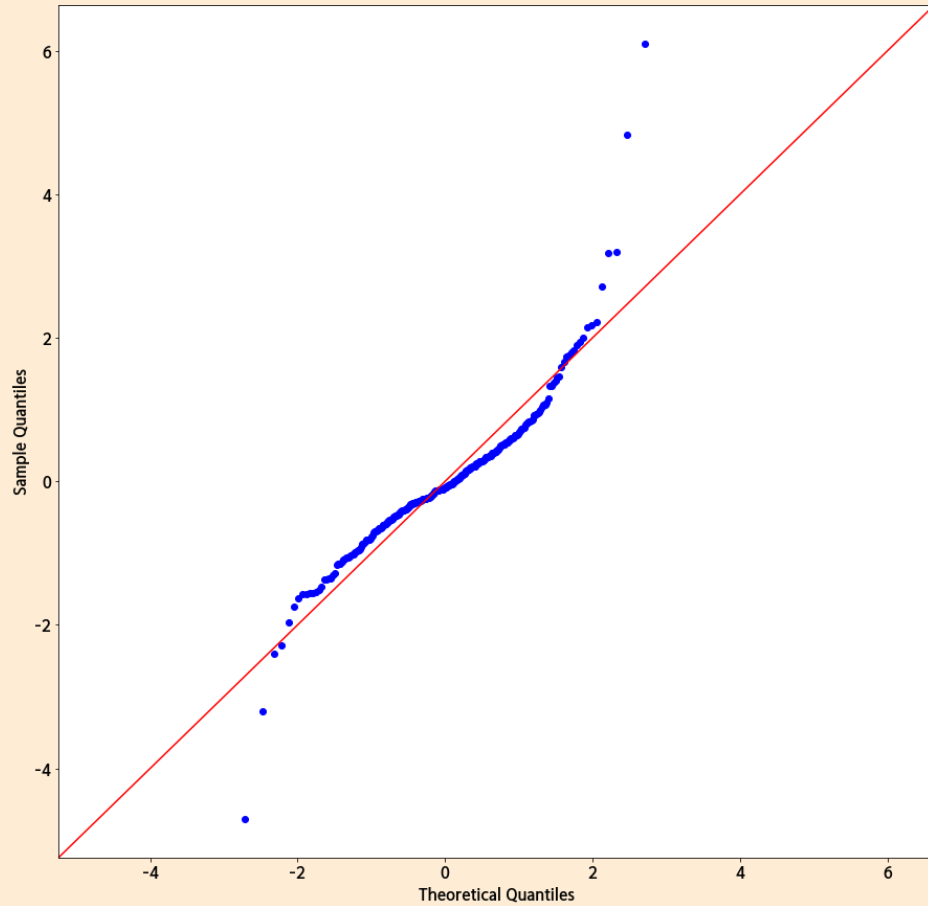
	FORWARD	BACKWARD	STEPWISE
MSE	1.29907e+09	1.29907e+09	1.29907e+09
RMSE	36042.6	36042.6	36042.6
MAE	21589.9	21589.9	21589.9
AIC	6903.26	6903.26	6903.26





# 회귀분석 진행

다중공선성 확인



	VIF Factor	features
0	17.080367	const
1	1.213639	경사도
2	1.378512	자전거도로길이
3	1.984796	총거치대수
4	1.438113	인구
5	1.366486	지하철역개수
6	1.874294	버스정거장개수
7	2.792255	스타벅스매장수
8	2.273887	공시지가

모든 VIF Factor가 10 이하임을 확인



# 회귀분석 진행

## 다중공선성 확인

```
x1 = all[['경사도', '자전거도로길이', '총거치대수', '인구', '지하철역개수', '버스정거장개수', '스타벅스매장수', '공시지가']]
target = all[['총대여수']]

x1_ = sm.add_constant(x1, has_constant = "add")

X = x1_
y = target

train_x, test_x, train_y, test_y = train_test_split(X,y, train_size = 0.7, test_size = 0.3, random_state = 1)

#print(train_x.shape, test_x.shape, train_y.shape, test_y.shape)

# 회귀모델 적합
multi_model_1 = sm.OLS(train_y,train_x)
fitted_multi_model_1 = multi_model_1.fit()

# 결과 출력
fitted_multi_model_1.summary()
```

Adj. R-squared: 0.570

	coef	std err	t	P> t
const	1.233e+04	6478.604	1.903	0.058
경사도	159.0969	46.591	3.415	0.001
자전거도로길이	1.7899	0.538	3.329	0.001
총거치대수	409.2615	40.896	10.007	0.000
인구	0.5617	0.208	2.695	0.007
지하철역개수	3382.5861	2240.339	1.510	0.132
버스정거장개수	-305.3308	153.312	-1.992	0.047
스타벅스매장수	4302.9913	1133.287	3.797	0.000
공시지가	-0.0022	0.001	-2.299	0.022



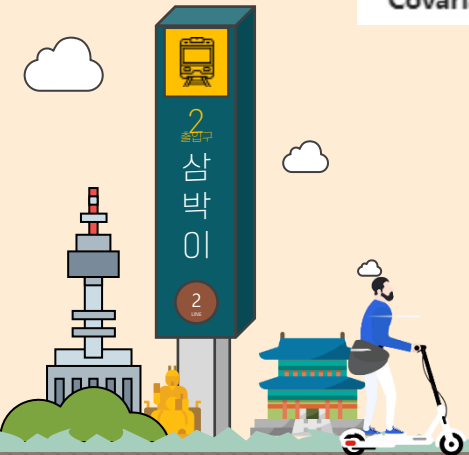
# 회귀분석 진행

## 다중공선성 확인

OLS Regression Results			
Dep. Variable:	총대여수	R-squared:	0.578
Model:	OLS	Adj. R-squared:	0.568
Method:	Least Squares	F-statistic:	56.42
Date:	Mon, 16 Nov 2020	Prob (F-statistic):	1.98e-50
Time:	17:40:14	Log-Likelihood:	-3443.8
No. Observations:	296	AIC:	6904.
Df Residuals:	288	BIC:	6933.
Df Model:	7		
Covariance Type:	nonrobust		

	coef	std err	t	P> t	[0.025	0.975]
const	1.391e+04	6407.653	2.171	0.031	1297.137	2.65e+04
경사도	160.7224	46.682	3.443	0.001	68.841	252.604
자전거도로길이	1.7326	0.537	3.224	0.001	0.675	2.790
총거치대수	418.8863	40.485	10.347	0.000	339.202	498.571
인구	0.5710	0.209	2.735	0.007	0.160	0.982
버스정거장개수	-307.4058	153.646	-2.001	0.046	-609.817	-4.995
스타벅스매장수	4672.3009	1109.032	4.213	0.000	2489.465	6855.136
공시지가	-0.0022	0.001	-2.350	0.019	-0.004	-0.000
Omnibus:	85.751	Durbin-Watson:	2.095			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	514.191			
Skew:	1.026	Prob(JB):	2.21e-112			
Kurtosis:	9.122	Cond. No.	1.68e+07			

P-value < 0.05 → 모두 유의함



# 교차분석



# 방법론 모델

## 교차분석 기준

환경 적합도 지수 =  $A_i + B_i + C_i$  (3)

$A_i$ :  $i$  동의 경사도 z값

$B_i$ :  $i$  동의 생산인구 z값

$C_i$ :  $i$  동의 자전거도로 z값

$$SI_A = \frac{4 \times Sub_A + Bus_A + Bike_A}{Area_A} \quad (2)$$

$SI_A$ : 행정동 A의 대중교통 접근성

$Area_A$ : 행정동 A의 면적

$Sub_A$ : 행정동 A의 지하철역수

$Bus_A$ : 행정동 A의 버스정류장수

$Bike_A$ : 행정동 A의 자전거 거치대수

환경지수

교통지수

	동	환경지수	교통지수	tr_index	sum_index
330	일원1동	-0.432038	-1.643420	4	4
331	일원2동	0.973068	-1.575956	4	3
332	일원본동	1.726340	-0.070685	3	2
333	자양1동	0.765226	-1.313902	4	3
334	자양2동	2.216005	-1.521985	4	2
...	...	...	...	...	...
425	효창동	-1.609028	-1.126482	4	4
426	후암동	-3.304315	-1.587342	4	4
427	휘경1동	-0.247009	-1.132175	4	4
428	휘경2동	0.931054	-0.910598	4	3
429	흑석동	0.388666	2.643238	2	3



# 방법론 모델

## 종합분석- 교차분석

		교통 취약성	
		하위50%	하위25%
물리적 환경 적합성	상위 25%	High-Mid	High-Low
	상위50%	Mid-Low	Mid-Low

Table2. 교차분석 2x2 테이블

	환경지수	교통지수
25%	-1.200534	-1.031876
50%	0.033588	-0.392293
75%	1.237509	0.667773
max	9.008577	14.527998

```
def quant_sum(x):
    if x >= 9.008577:
        return 1
    elif x >= 1.167670 :
        return 2
    elif x >= 0.003758 :
        return 3
    else:
        return 4

def quant_trf(x):
    if x >= 1.452800e+01 :
        return 1
    elif x >= 6.198185e-01 :
        return 2
    elif x >= -3.960821e-01:
        return 3
    else:
        return 4
```

	동	환경지수	교통지수	tr_index	sum_index
330	일원1동	-0.432038	-1.643420	4	4
331	일원2동	0.973068	-1.575956	4	3
332	일원본동	1.726340	-0.070685	3	2
333	자양1동	0.765226	-1.313902	4	3
334	자양2동	2.216005	-1.521985	4	2
...	...	...	...	...	...
425	효창동	-1.609028	-1.126482	4	4
426	후암동	-3.304315	-1.587342	4	4
427	휘경1동	-0.247009	-1.132175	4	4
428	휘경2동	0.931054	-0.910598	4	3
429	흑석동	0.388666	2.643238	2	3

환경적합도지수와 교통 취약성지수를 4분위수로 구분하여  
교차분석한 지점을 자전거 활성화에 영향을 미치는 지표로 선정



# 교차분석

## 종합분석 - 교차분석

```
def inter_g(s,t):
    if s==1 and t==4:
        return 1
    elif s==1 and t==3:
        return 2
    elif s==2 and t==4:
        return 3
    elif s==2 and t==3:
        return 4
```

```
for key, group in grouped:
    print('key',key)
    print('group',group)
    print(group.head())
```

```
key 3.0
group
10   갈현2동  1.416783 -0.599009      4      2      3.0
13   개봉2동  1.535743 -0.694086      4      2      3.0
20   고덕1동  2.065361 -0.424232      4      2      3.0
71   대치1동  2.767799 -1.031404      4      2      3.0
92   등촌1동  2.197835 -0.839549      4      2      3.0
93   등촌2동  1.545358 -1.641941      4      2      3.0
100  망원1동  3.109215 -0.628730      4      2      3.0
102  면목2동  1.517241 -1.087482      4      2      3.0
112  목2동    2.547823 -1.008633      4      2      3.0
136  방이2동  1.647925 -0.958877      4      2      3.0
168  삼전동   3.864709 -1.061126      4      2      3.0
176  상계8동  2.126611 -1.643420      4      2      3.0
182  상봉1동  1.223595 -0.542081      4      2      3.0
189  서빙고동 2.569377 -0.694086      4      2      3.0
196  석촌동   2.539781 -1.032033      4      2      3.0
204  서수정1동 1.175117 -1.020647      4      2      3.0
```

```
#교차분석 done : pandas , 두가지 열 apply
dfinter['interquant']=dfinter.apply(lambda x: inter_g(x.sum_index,x.tr_index),axis=1)
```

		교통 취약성	
		하위 50%	하위 25%
물리적 환경 적합성	상위 25%	2	1
	상위 50%	4	3

## 교차분석 기준

1. 교통 취약성
  2. 물리적 환경 적합성
- 결과 도출 방법 : 4 분위수 기준

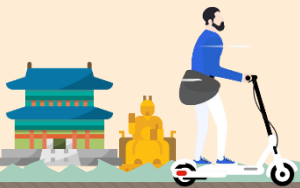
교통 취약성 하위 25%, 50%  
환경 적합성 상위 25%, 50%

# 교차분석

	동	교차분석결과	교통10분위수	공시지가 10분위수	(취약+공시)/2	우선순위결과
0	상계8동	3	1	2	5.5	2.0
1	등촌2동	3	1	2	7.5	2.0
2	잠실2동	3	2	1	8.5	2.5
3	암사2동	3	2	1	7.0	2.5
4	이촌2동	3	1	3	7.0	2.5
5	풍납1동	3	1	3	7.0	2.5
6	자양2동	3	2	3	9.5	3.5
7	잠실4동	3	2	6	19.0	5.0
8	면목2동	3	3	5	21.0	5.5
9	수서동	3	6	1	23.0	6.5
10	신정4동	3	3	7	25.0	6.5
11	천호1동	3	3	7	22.5	6.5
12	서빙고동	3	6	2	28.0	7.0
13	송파1동	3	3	9	26.5	7.5
14	개봉2동	3	6	4	29.5	8.0

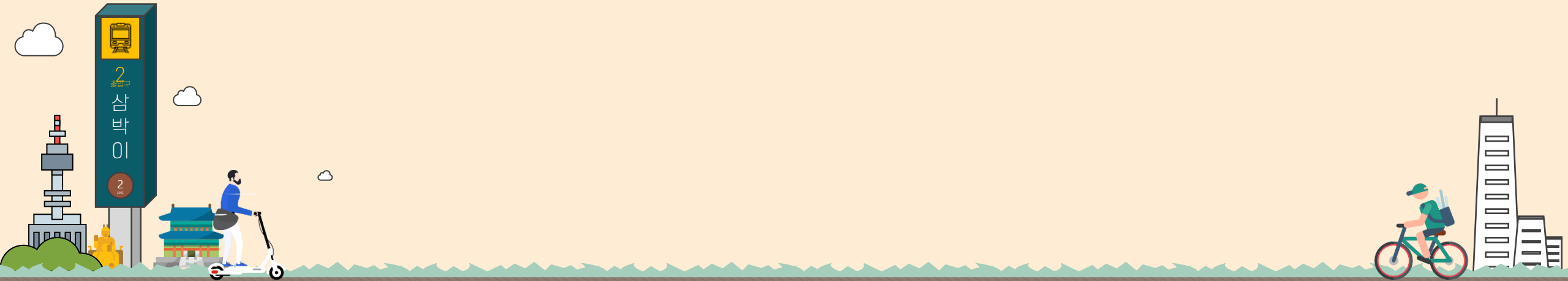
## 교통 취약계층 [형평성 고려]

교차분석 결과,  
 도입필요 지역의 소득 수준이 높은 지역인 경우가 많아  
공시지가 변수를 추가해 교통 취약계층이 많은 지역을 선정





# 도입지 우선순위 선정



# API 분석- 이동성 측면

분석 목적 : 대중교통 이동 시간 취약성 측면 위치 선정

공공 API 활용 데이터 수집 : 1. T-map API  
2. TOPIS 대중교통 환승 정보 API

분석 기준	분석 설명
형평성	자동차 통행시간 - 대중교통 통행시간(분)

분석 시간 : 오후 6시 출발

데이터 : 강남역, 시청역 업무지구 -> 교차분석 위치선정 지역  
실시간 소요시간 데이터 수집

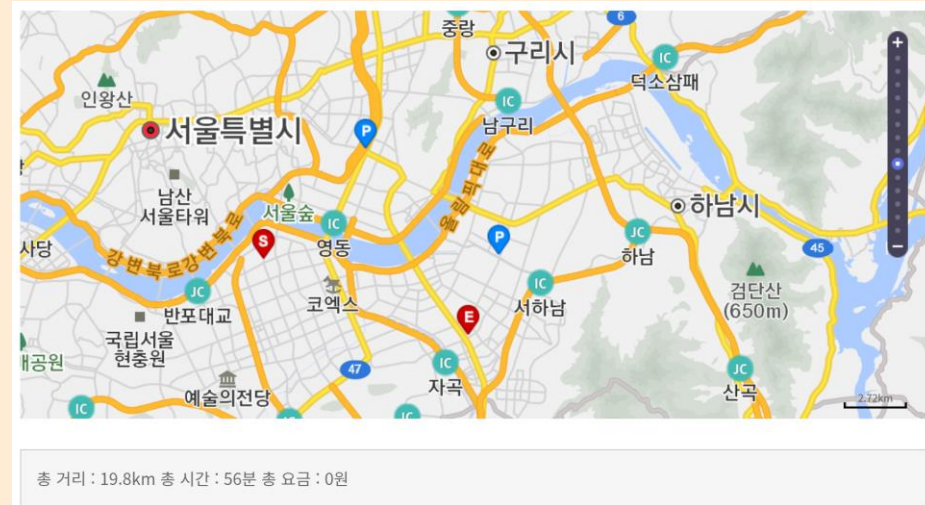
강남역 출발(오후 6시)

```
from selenium import webdriver

x = ['127.051581', '126.862524', '127.127259', '126.954635', '127.121997', '127.084334', '127.112268', '127.079023',
y = ['37.666886', '37.542747', '37.551953', '37.526423', '37.538177', '37.529217', '37.520243', '37.590064', '37.4890

driver = webdriver.Chrome('C:/Temp/chromedriver.exe')
driver.implicitly_wait(3)

#강남역 출발 - 상계8동 도착
url_startX = 'http://ws.bus.go.kr/api/rest/pathinfo/getPathInfoByBusNsub?ServiceKey=MT9k5VcAGch1xQwYHcvv1eJ9TCzetVpOF
url_startY = '&startY=37.4980854411503'
url_endX = '&endX='
url_endY = '&endY='
url = url_startX + url_startY + url_endX + x[0] + url_endY + y[0]
driver.get(url)
```



# 교통취약지 분석

〈통근 시간대 대중교통 및 자차 이동시간〉

오후 6시		출발지			
		Tmap api(강남역)	Tmap api(시청역)	대중교통 api(강남역)	대중교통 api(시청역)
도 착 지	상계8동	88	80	65	64
	등촌2동	77	63	76	60
	암사2동	59	72	48	68
	이촌2동	54	28	71	49
	풍납1동	57	69	41	48
	자양2동	46	58	56	64
	잠실4동	55	69	40	63
	면목2동	73	63	69	36
	수서동	48	72	44	55
	신정4동	86	62	55	43
	천호1동	61	75	55	69
	서빙고동	27	18	43	40
	송파1동	49	71	28	48
	개봉2동	76	72	58	57

〈PM 우선도입 필요 순위〉

동	강남역	시청역	합계	순위
서빙고동	16	22	19	1
이촌2동	17	21	19	1
자양2동	10	6	8	3
등촌2동	-1	-3	-2	4
천호1동	-6	-6	-6	5
암사2동	-11	-4	-7.5	6
수서동	-4	-17	-10.5	7
잠실4동	-15	-6	-10.5	7
면목2동	-4	-27	-15.5	9
개봉2동	-18	-15	-16.5	10
풍납1동	-16	-21	-18.5	11
상계8동	-23	-16	-19.5	12
송파1동	-21	-23	-22	13
신정4동	-31	-19	-25	14



# 결론

분석 의의 : PM 도입의 형평성 제고를 위한 연구

분석 요소 : 교통 취약지 문제 해결을 위한 PM 도입 지역

변수 요소 : PM 특성상 도입에 필요한 환경요소  
교통 취약지 분석을 위한 교통 접근성

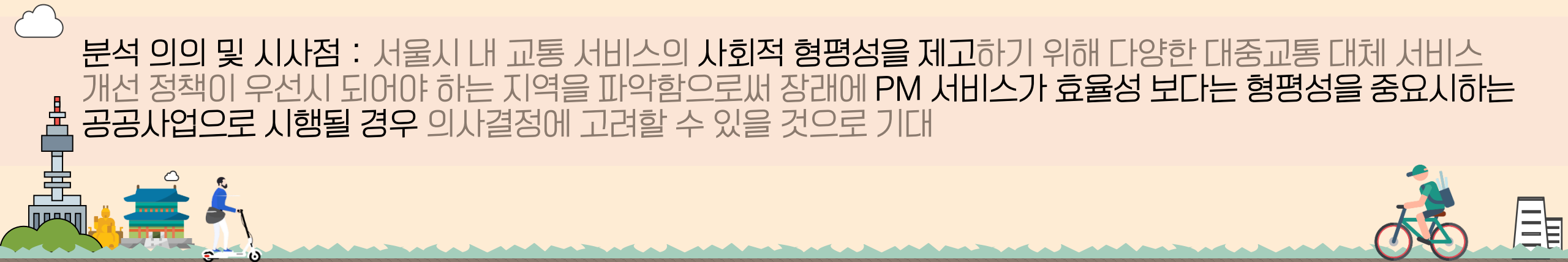
분석 과정 : 타당성 확보를 위한 예상 요소의 다중회귀 분석 후 2가지 상이한 지표를 고려한 교차분석

1차 분석 결과 : 지표 분석 결과 자차 이용률이 높을 것으로 예상되는 일부 지역이 선정됨

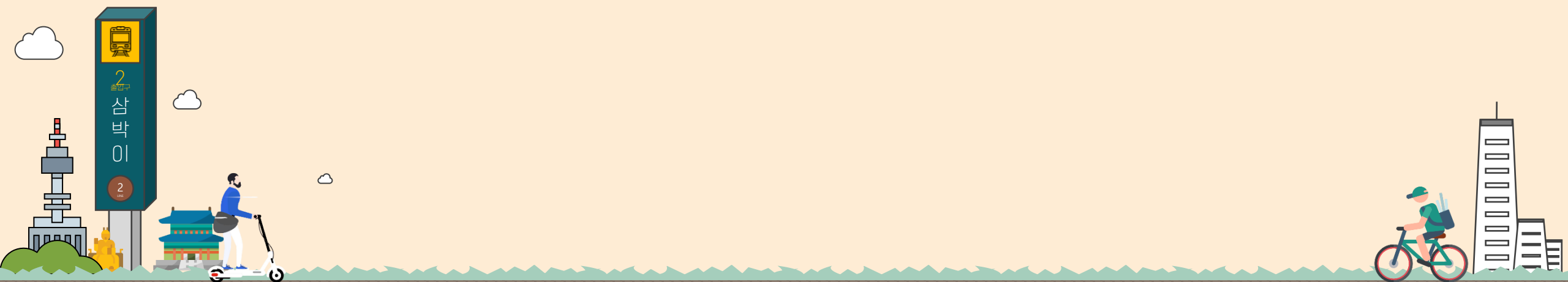
2차 분석 결과 : 취약계층 기준 교통 취약성 고려를 위한 공시지가 요소 추가

3차 분석 결과 : 자차 대비 대중교통 이용시간 분석 후 우선순위 산출 (교통이동성 고려)

분석 의의 및 시사점 : 서울시 내 교통 서비스의 사회적 형평성을 제고하기 위해 다양한 대중교통 대체 서비스 개선 정책이 우선시 되어야 하는 지역을 파악함으로써 장래에 PM 서비스가 효율성 보다는 형평성을 중요시하는 공공사업으로 시행될 경우 의사결정에 고려할 수 있을 것으로 기대



# 워드클라우드



# 워드클라우드

2020년 11월 13일 기준 네이버 뉴스 기사 제목, 기사 100개 스크래핑

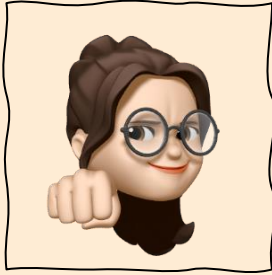
한밤중 역주행하던 전동킥보드, 오토바이와 충돌

광주광역시, 전동킥보드 안전관리 나섰다

송재혁 서울시의원, "전동 킥보드 자전거도 통행 대책 마련을"



# 느낀점



박성희\_팀장

“분석과정에서 예상과 다른 결과가 나올 때 해결책을 찾아가는 과정이 어려웠다. 논리적으로 흐름을 짜더라도 데이터의 결과는 다를 수 있기에 분석이란 더 정확도를 높이는 과정이라는 걸 느꼈다. 팀원 모두 생소한 주제에도 열심히 참여해줘서 고맙다”



박성준\_팀원

“데이터 전처리 과정이 생각보다 오래 걸렸다. 데이터 분석을 하며 Garbage in, garbage out 이라는 말을 온몸으로 체험할 수 있었다. 잘 모르는 부분을 함께 풀어나가며 프로젝트를 진행한 동료들에게 정말 고맙다는 말을 하고 싶다.”



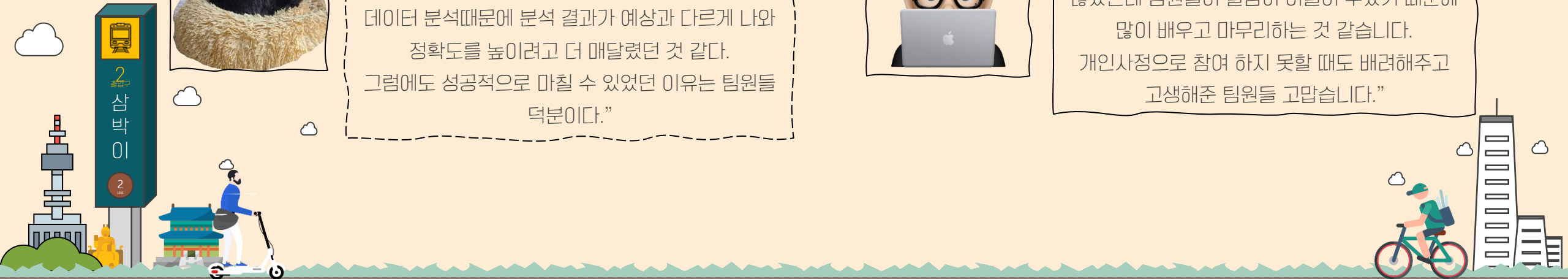
박주희\_팀원

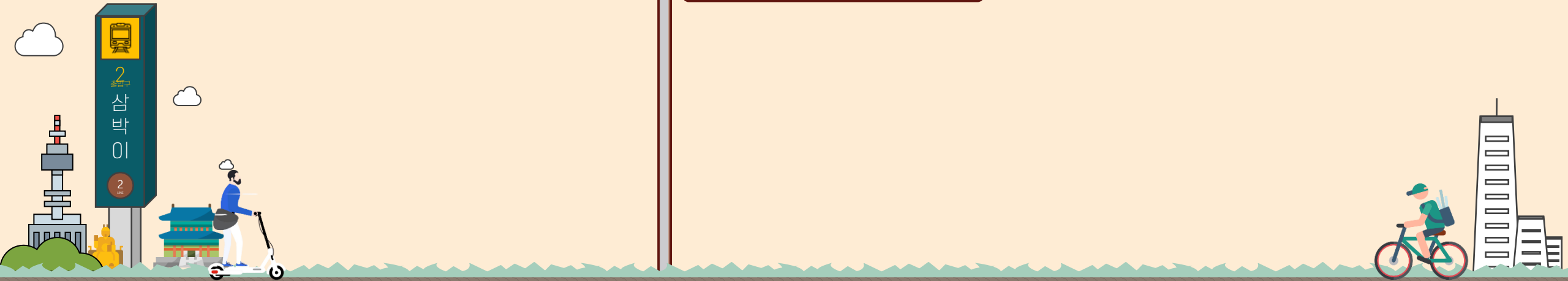
“데이터 수집과 전처리에서 시간을 생각보다 많이 쏟아서 일정관리를 잘 하지 못해 아쉽다. 데이터 분석때문에 분석 결과가 예상과 다르게 나와 정확도를 높이려고 더 매달렸던 것 같다. 그럼에도 성공적으로 마칠 수 있었던 이유는 팀원들 덕분이다.”



이동규\_팀원

“이번 프로젝트 관련해서 부족하고 모르는 부분이 많았는데 팀원들이 열심히 이끌어 주었기 때문에 많이 배우고 마무리하는 것 같습니다. 개인사정으로 참여 하지 못할 때도 배려해주고 고생해준 팀원들 고맙습니다.”









2

삼박이

감사합니다



2

삼박이

2

