

전공별 AI활용(SW)

지하철 유동인구 분석을
통한 최적의 광고 위치 선정



32203928 장현정

START

INDEX

1. 주제 소개

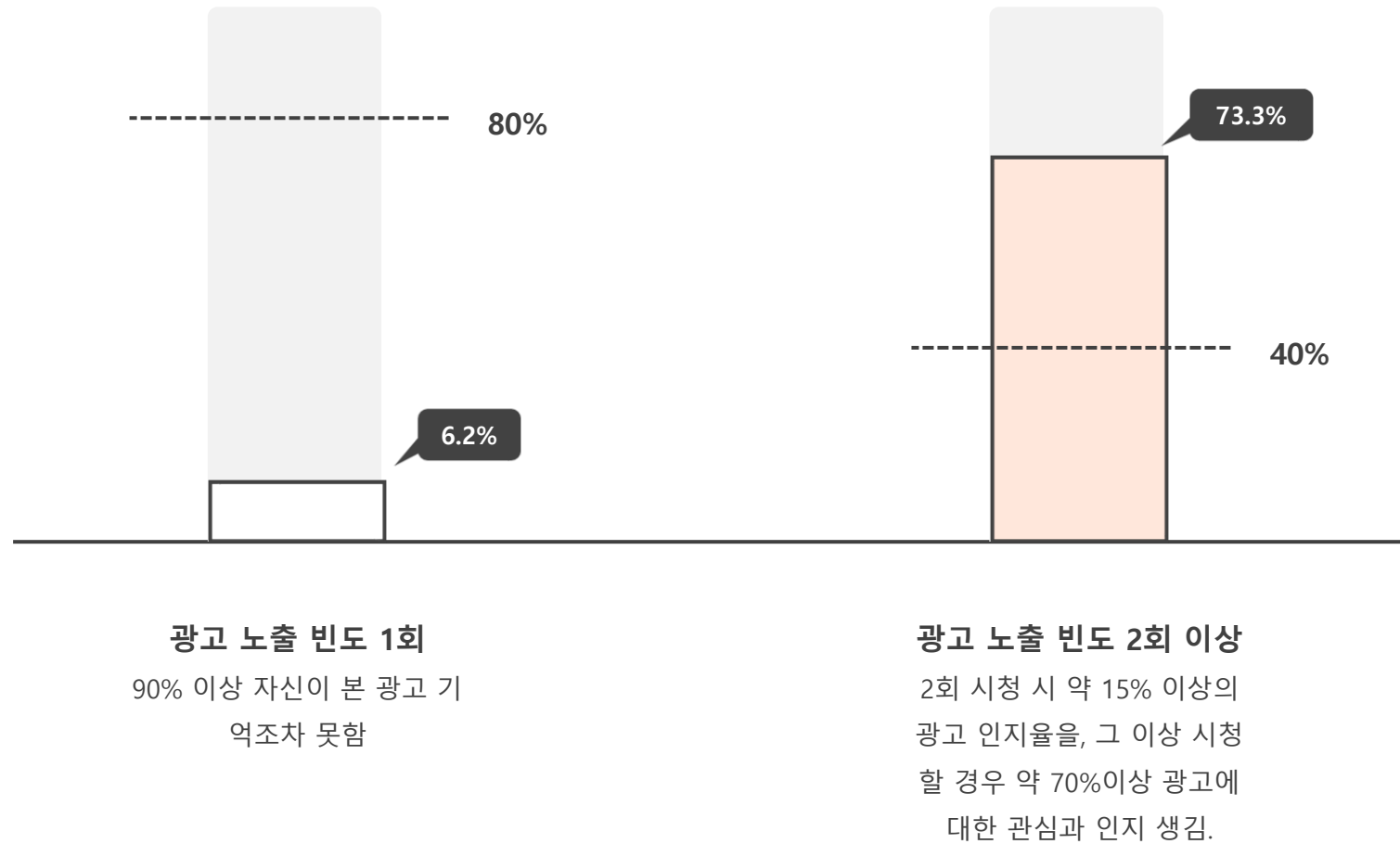
- 배경 및 목표

2. 분석 과정

- 데이터 수집
- 코드 설명
- 결과 실행

3. 결론

주제 소개



주제 소개



지하철 광고 또한 유동인구가 많을수록 사람들에게 광고 노출 기회가 많아지고 광고의 효율성을 높일 수 있음.



지하철 노선, 역, 시간대 별로 이용객 수를 분석해 가장 효과적인 광고 효과를 얻을 수 있는 시간대와 장소는 어디인지 알아보려고 한다.

분석 과정

☑ 가장 이용객이 많은 노선

	사용일자	노선명	역명	승차총승객수	하차총승객수	등록일자
0	20211101	1호선	동대문	10855	10327	20211104
1	20211101	1호선	신설동	13933	13614	20211104
2	20211101	1호선	제기동	17584	17948	20211104
3	20211101	1호선	청량리(서울시립대입구)	19604	19787	20211104
4	20211101	2호선	시청	23006	22704	20211104
...
600	20211101	2호선	신림	60916	58969	20211104
601	20211101	2호선	신대방	26338	25427	20211104
602	20211101	6호선	역촌	3832	4456	20211104
603	20211101	1호선	종로3가	24261	22213	20211104
604	20211101	1호선	종로5가	21309	21179	20211104

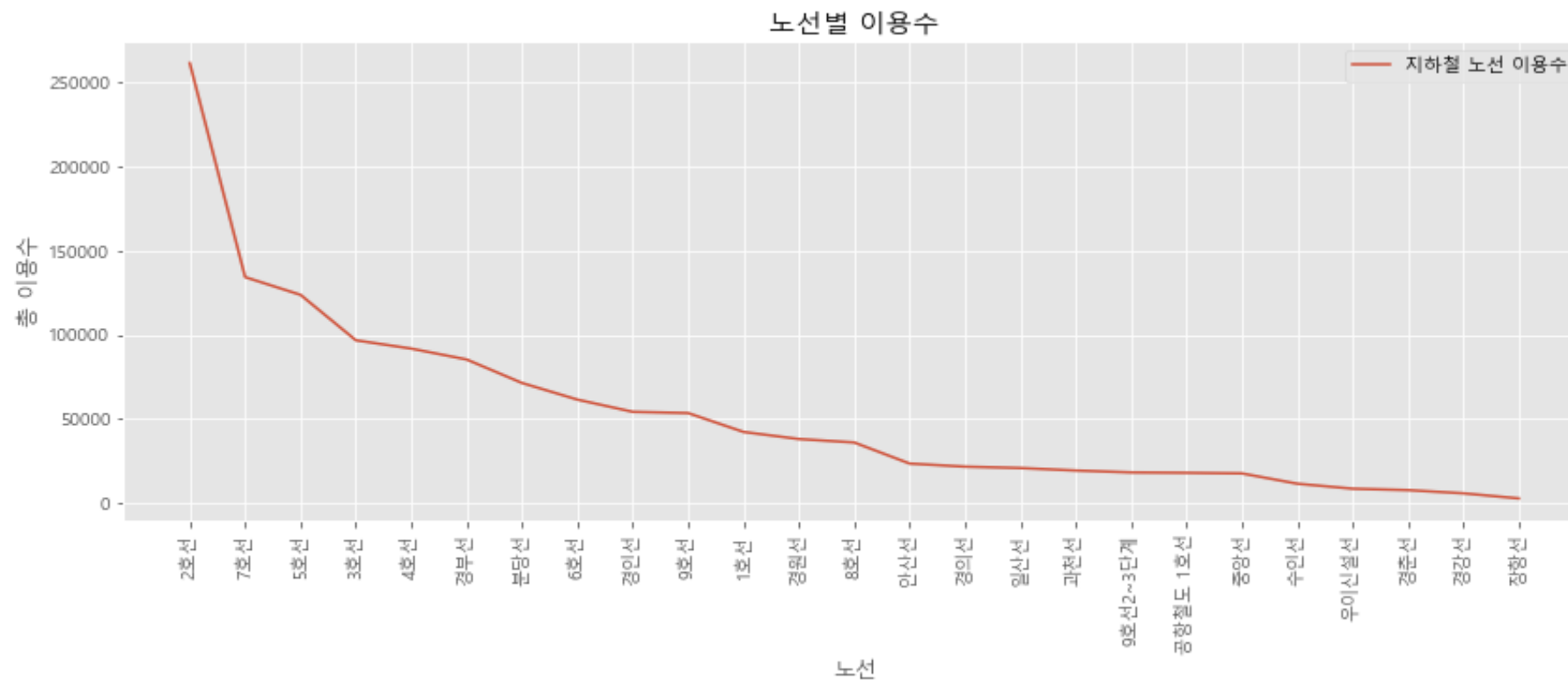
605 rows × 6 columns

서울시 지하철 호선별 역별 승하차 인원 정보

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 df = pd.read_csv('지하철.csv', encoding = 'euc-kr')
5
6 from matplotlib import font_manager, rc
7 font_path = './malgun-ttf'
8 font_name = font_manager.FontProperties(fname = font_path).get_name()
9 rc('font', family=font_name)
10
11 # 승차인원과 하차인원 더한 총 이용수를 나타내는 열 추가
12 df['총 이용수'] = df['승차총승객수'] + df['하차총승객수']
13
14 df = df.drop('등록일자', axis = 1) # 불필요한 열 삭제
15 subway = df.groupby('노선명').sum()[['총 이용수']].sort_values(by = '총 이용수', ascending = False)
16
17 plt.style.use('ggplot')
18 plt.figure(figsize = (14, 5))
19
20 plt.plot(subway.index, subway.values // 10)
21 plt.title('노선별 이용수')
22 plt.xlabel('노선')
23 plt.ylabel('총 이용수')
24 plt.xticks(rotation = 'vertical')
25 plt.legend(labels=['지하철 노선 이용수'], loc = 'best')
26
27 plt.show()
```

분석 과정

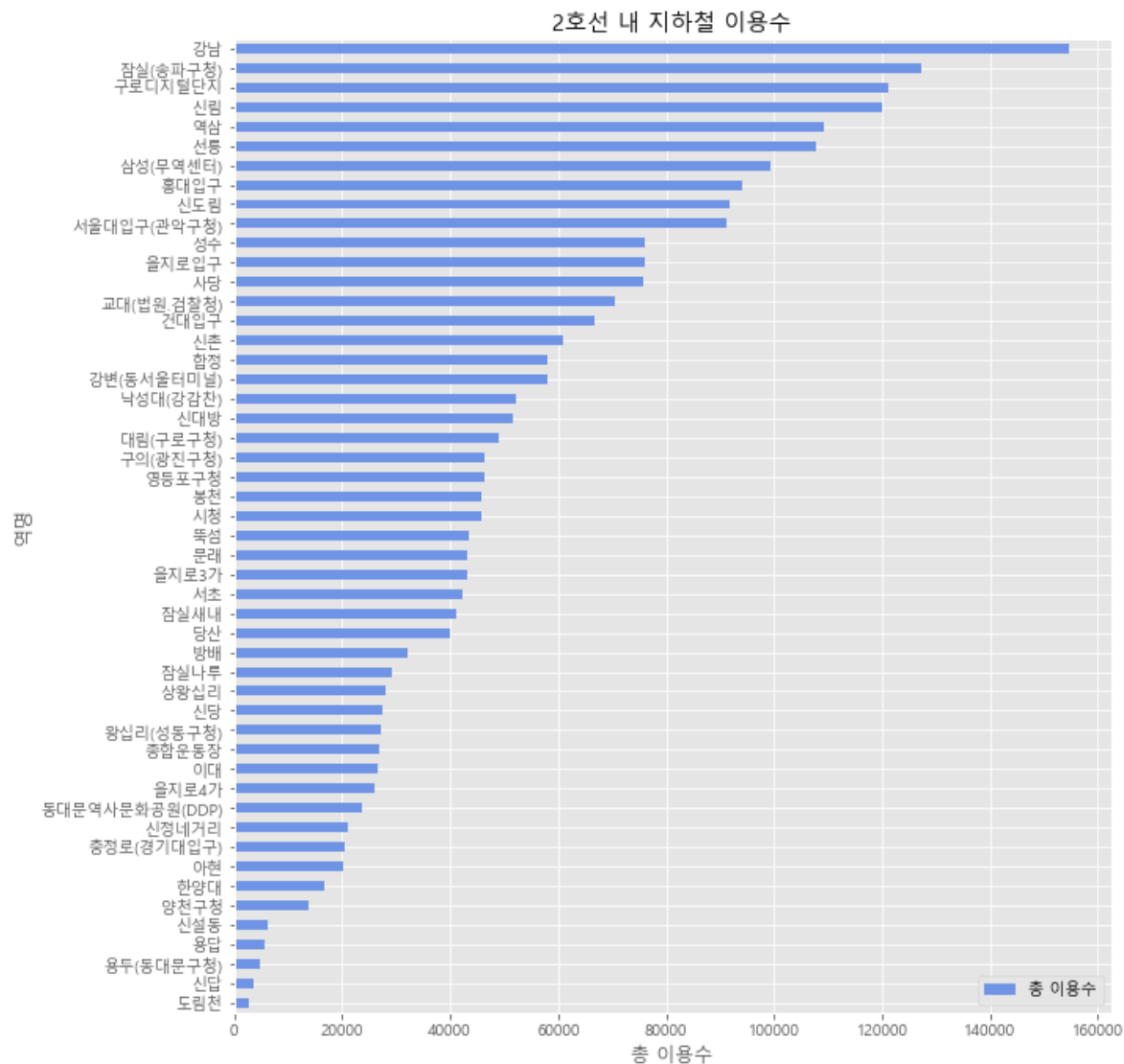
☑ 가장 이용객이 많은 노선



분석 과정

✓ 2호선 중 가장 이용객이 많은 역

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 df = pd.read_csv('지하철.csv', encoding = 'euc-kr')
5
6 from matplotlib import font_manager, rc
7 font_path = './malgun-ttf'
8 font_name = font_manager.FontProperties(fname = font_path).get_name()
9 rc('font', family=font_name)
10
11 # 승차인원과 하차인원 더한 총 이용수를 나타내는 열 추가
12 df['총 이용수'] = df['승차총승객수'] + df['하차총승객수']
13
14 df = df.drop('등록일자', axis = 1) # 불필요한 열 삭제
15
16 station = (df['사용일자'] == 20211101) & (df['노선명'] == '2호선')
17 df_station = df[station]
18 station_num = df_station.groupby('역명').sum()[['총 이용수']].sort_values(by = '총 이용수')
19
20 plt.style.use('ggplot')
21
22 station_num.plot(kind = 'barh', color = 'cornflowerblue', width = 0.5, figsize = (10, 11))
23
24 plt.title('2호선 내 지하철 이용수')
25 plt.xlabel('총 이용수')
26 plt.ylabel('역명')
27
28 plt.show()
```



분석 과정

강남역 시간대별 지하철 이용수

시도별	04	04	05	05	06	06	07	...	23	00	00	01	01	02	02	03	03	작업일차
지하철	시승	시하	시승	시하	시승	시하	시승	...	시하	시승	시하	시승	시하	시승	시하	시승	시하	
0	202111	1	782	14	12451	2166	8849	6242	14133	...	8790	47	1776	0	0	0	0	20211203
1	202111	1	141	1	2570	907	3387	4074	5711	...	2974	8	2017	0	2	0	0	20211203
2	202111	1	630	11	8985	7058	12028	40803	36492	...	12360	101	1257	1	3	2	2	20211203
3	202111	1	30	0	2006	4859	2980	19785	6504	...	2196	65	309	0	0	0	0	20211203
4	202111	1	327	8	8031	2831	7820	10278	16712	...	7668	39	1222	0	0	0	0	20211203
...
48372	201501	1	0	0	177	56	642	403	1292	...	633	3	222	0	0	0	0	20151223
48373	201501	1	1	0	292	4	415	46	537	...	383	3	157	0	0	0	0	20151223
48374	201501	1	14	1	4929	424	8539	2084	22160	...	6501	86	1617	0	0	0	0	20151223
48375	201501	1	0	0	161	50	368	149	1227	...	336	8	85	0	0	0	0	20151223
48376	201501	1	436	3	10929	3067	23317	10430	65043	...	35702	767	9314	0	0	0	0	20151223

48377 rows x 52 columns

```

1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 from matplotlib import font_manager, rc
5 font_path = './malgun-ttf'
6 font_name = font_manager.FontProperties(fname = font_path).get_name()
7 rc('font', family=font_name)
8
9 # 승차인원 열만 가져오기
10 colup = [2]
11 for i in range(1,25):
12     colup.append(2*i+1)
13 colup = list(map(int, colup))
14
15 # 하차인원 열만 가져오기
16 coldown = []
17 for i in range(1,26):
18     coldown.append(2*i)
19 coldown = list(map(int, coldown))
20
21 # 강남역 11월 승차 인원 평균
22 df_up = pd.read_csv('서울시 지하철 인원 정보.csv', encoding = 'euc-kr',
23                     skiprows=[1,2,3,4,5,6,7,8,9,10], index_col = '지하철역', nrows = 50, usecols = colup)
24 df_up.columns = ['04시-05시', '05시-06시', '06시-07시', '07시-08시', '08시-09시',
25                 '09시-10시', '10시-11시', '11시-12시', '12시-13시', '13시-14시',
26                 '14시-15시', '15시-16시', '16시-17시', '17시-18시', '18시-19시',
27                 '19시-20시', '20시-21시', '21시-22시', '22시-23시', '23시-24시',
28                 '00시-01시', '01시-02시', '02시-03시', '03시-04시']
29
30 # 강남역 11월 하차 인원 평균
31 df_down = pd.read_csv('서울시 지하철 인원 정보.csv', encoding = 'euc-kr',
32                       skiprows=[1,2,3,4,5,6,7,8,9,10], index_col = '지하철역', nrows = 50, usecols = coldown)
33 df_down.columns = ['04시-05시', '05시-06시', '06시-07시', '07시-08시', '08시-09시',
34                   '09시-10시', '10시-11시', '11시-12시', '12시-13시', '13시-14시',
35                   '14시-15시', '15시-16시', '16시-17시', '17시-18시', '18시-19시',
36                   '19시-20시', '20시-21시', '21시-22시', '22시-23시', '23시-24시',
37                   '00시-01시', '01시-02시', '02시-03시', '03시-04시']
38
39 df_total = df_up.iloc[0] + df_down.iloc[0] # 강남역 승하차 인원 합계
40
41 # 강남역 승하차, 총 인원의 값만 가진 데이터프레임 생성
42 new_df = pd.DataFrame([df_up.iloc[0].values, df_down.iloc[0].values, df_total.values],
43                       index=['승차인원', '하차인원', '총인원'], columns = df_up.columns)
44

```

```

41 # 강남역 승하차, 총 인원의 값만 가진 데이터프레임 생성
42 new_df = pd.DataFrame([df_up.iloc[0].values, df_down.iloc[0].values, df_total.values],
43                       index=['승차인원', '하차인원', '총인원'], columns = df_up.columns)
44
45 plt.style.use('ggplot')
46
47 # 각 시간대별로 승차인원과 하차인원을 비교해 보여주기 위해 승차인원과 하차인원 행만 가져옴.
48 new_df_updown = new_df.loc[['승차인원', '하차인원']]
49 new_df_updown = new_df_updown.transpose()
50 ax1 = new_df_updown.plot(kind='bar', figsize=(25,15), width=0.7, color=['orange', 'green'])
51
52 # 총 인원수를 같은 막대에 보여주기 위해 2축 그래프 생성
53 ax2 = ax1.twinx()
54 ax2.plot(new_df.columns, new_df.iloc[2].values, ls='--', marker='o', markersize=10,
55         color='red', label = '총 인원 수') # 총 인원수 그래프 그리기
56
57 plt.title('강남역 시간대별 지하철 이용 인원수', fontsize=25)
58
59 ax1.set_xlabel('시간', size = 20)
60 ax1.set_ylabel('(승하차)인원수', size = 20)
61 ax2.set_ylabel('총 인원수', size = 20)
62 ax1.legend(loc='best', fontsize=20)
63 plt.yticks(size=20)
64 plt.legend(loc='upper left', fontsize=20)
65
66 plt.show()

```

	04	05	06	07	08	09	10	11	12	13	...	18	19	20	21	22	23	00	01	02
시	시	시	시	시	시	시	시	시	시	시	...	시	시	시	시	시	시	시	시	시
승차인원	219	8681	17040	36970	64022	50557	49286	65524	81700	96001	...	332029	202805	154586	184097	180028	71371	87	1	0
하차인원	4	10474	48420	134376	310431	318239	147977	105735	105018	121180	...	205453	117181	55556	42434	36659	24139	637	1	0
총인원	223	19155	65460	171346	374453	368796	197263	171259	186718	217181	...	537482	319986	210142	226531	216687	95510	724	2	0

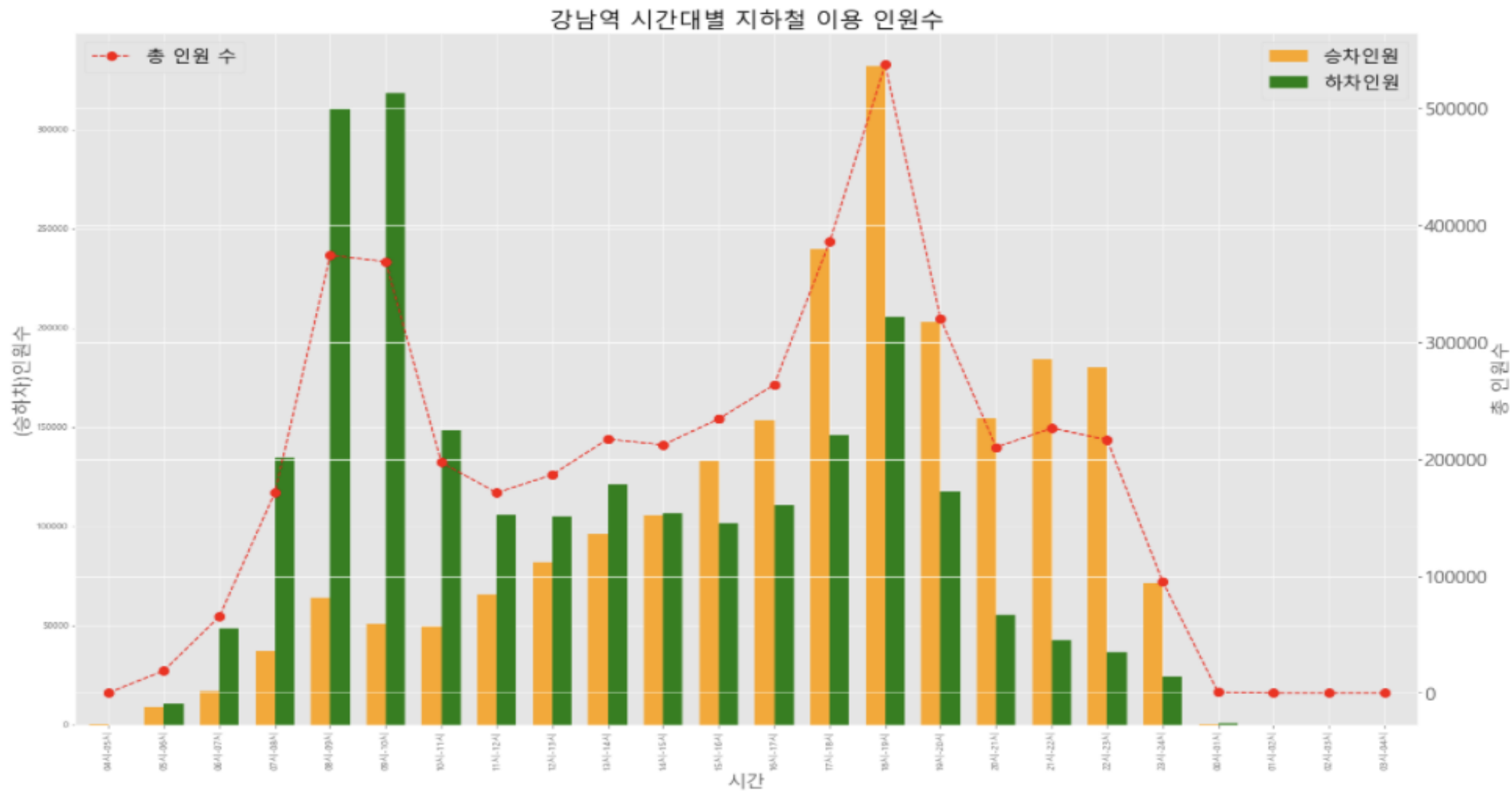
3 rows x 24 columns

시간대별 승차, 하차 인원 데이터

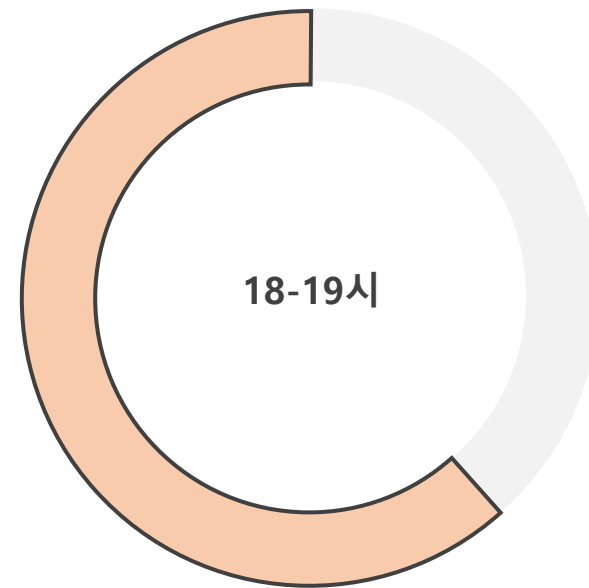
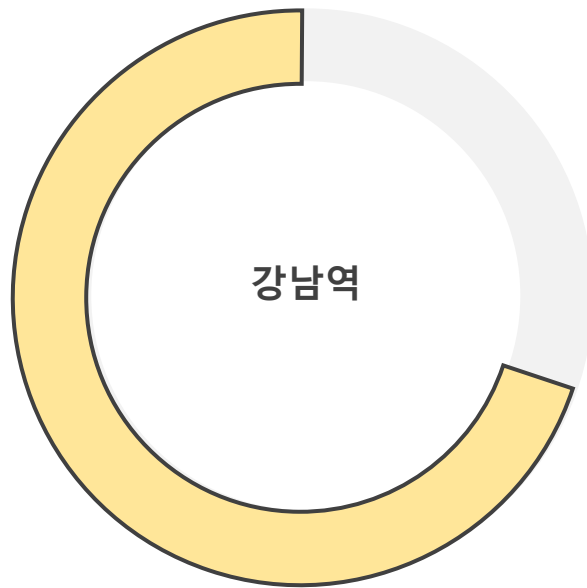
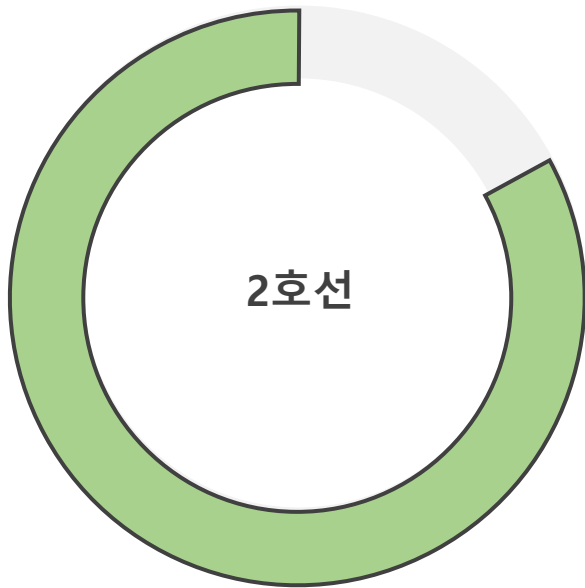
새로운 데이터프레임 ->

분석 과정

☑ 강남역 시간대별 지하철 이용수



결론



2021년 11월 평균 기준)

다른 광고 매체에 비해 지하철은 유동인구가 많고 적은 비용으로 최대 효과를 낼 수 있어 최적의 광고 장소 중 하나로 손꼽힌다. 특히 스크린 도어는 넓은 지면을 활용해 다량의 정보를 전달하면서 지하철 내 광고 중에서도 승객의 움직임이 적어 최소 3분간은 강제 노출이 가능해 광고 효과가 높다.

앞서 분석 결과에서도 보았듯이 2호선 강남역 18-19시에 가장 많은 인원이 지하철을 이용하여 강남역에 있는 지하철 광고가 가장 많은 사람들에게 노출됨을 알 수 있다. 유동 인구가 많아 광고 노출도가 높지만, 그만큼 광고 비용이 다른 노선, 역에 비해 높다. 하지만 광고 비용에 비해 광고로 얻어 들일 수 있는 수익, 마케팅 효과가 다른 노선, 역에서 노출된 광고로 얻는 것보다 더 크기 때문에 기업들은 유동인구가 많은 자리를 선점하기 위해 경쟁이 치열하다.



감사합니다.



END