

---

Apache Spark를 활용하여

ASAC 김정우

지하철  
혼잡도를  
활용한 최적의  
출근 시간 및  
지역선정

---

# CONTENTS



Data Source

Preprocessing

Insight

Conclusion

---

\_\_\_\_\_

## 기본

- TMAP API

- SDK

- ## 지오패코딩

- › 교통정보

- ### ▶ 다중 경유

- ## > RoadAPI

- > 유가정보

- ## TMAP TMS 소개

- 권역정보

- ▶ 배송지정.

- ### 배차결과요청

- TMAP 대중교통 소개

SK 그룹의 다양한 API와 Data를 활용하여 새로운 서비스를 개발할 수 있습니다.

SK그룹의 다양한 API와 데이터를 제공하는 SK open API에 오신 것을 환영합니다.  
SK open API에서는 TMAP, META, SafeCaster, NUGU, TMAP TMS, GIRAF, SuperNova, 11st, Broadband 등 API 뿐만 아니라, SK의 빅데이터에 기반한 리포토와 분석 서비스까지 제공하고 있습니다.

## 분석



T 리포트



## API & Data

## API & Data

환영합니다!

### 주요 상품

교통/물류주거

쇼핑

여가/오락여행금융/경제통신/모바일보안

---

# DATA TARGET



서울 3대 업무지구인 CBD(종로구·중구 일대), GBD(강남구·서초구 일대), YBD(영등포구 일대)를 비롯해 서울디지털국가산업단지, 마곡&염창지구 등 대규모 업무지구 및 산업단지를 빠르게 이동할 수 있는 지역에 주거지가 위치하면 출퇴근 시 느끼는 대중교통 혼잡이나 장거리 이동에 대한 부담으로부터 자유로울 수 있다.

---

# PIPE\_LINE

- 1. call api -> json
- 2. json -> DataFrame
- 3. DataFrame -> csv
- 4. csv -> sparkDataFrame
- 5. preprocessing (up\_down, congest, time(by minute))



---

# RAW DATA

- 공덕역 6호선, MON, 07시

```
'contents': {'subwayLine': '6호선',
'stationName': '공덕역',
'stationCode': '626',
'stat': [{ 'startStationCode': '610',
'startStationName': '응암역',
'endStationCode': '626',
'endStationName': '공덕역',
'prevStationCode': '625',
'prevStationName': '대흥역',
'updnLine': 1,
'directAt': 0,
'data': [{ 'dow': 'MON', 'hh': '07', 'mm': '00', 'congestionTrain': 0},
{ 'dow': 'MON', 'hh': '07', 'mm': '10', 'congestionTrain': 0},
{ 'dow': 'MON', 'hh': '07', 'mm': '20', 'congestionTrain': 0},
{ 'dow': 'MON', 'hh': '07', 'mm': '30', 'congestionTrain': 0},
{ 'dow': 'MON', 'hh': '07', 'mm': '40', 'congestionTrain': 0},
{ 'dow': 'MON', 'hh': '07', 'mm': '50', 'congestionTrain': 0}]}],
{'startStationCode': '610',
'startStationName': '응암역',
'endStationCode': '631',
'endStationName': '한강진역',
'prevStationCode': '625',
'prevStationName': '대흥역',
'updnLine': 1,
'directAt': 0,
'data': [{ 'dow': 'MON', 'hh': '07', 'mm': '00', 'congestionTrain': 0},
{ 'dow': 'MON', 'hh': '07', 'mm': '10', 'congestionTrain': 0},
{ 'dow': 'MON', 'hh': '07', 'mm': '20', 'congestionTrain': 0},
{ 'dow': 'MON', 'hh': '07', 'mm': '30', 'congestionTrain': 0},
```

---

# RAW DATA

- 공덕역 6호선, MON, 07시

```
{'status': {'code': '00', 'message': 'success', 'totalCount': 1},  
'contents': {'subwayLine': '6호선',  
'stationName': '공덕역',  
'stationCode': '626',  
'stat': [{'startStationCode': '610',  
'startStationName': '응암역',  
'endStationCode': '626',  
'endStationName': '공덕역',  
'prevStationCode': '625',  
'prevStationName': '대흥역',  
'updnLine': 1,  
'directAt': 0,  
'data': [{'dow': 'MON',  
'hh': '07',  
'mm': '00',  
'congestionCar': [0, 0, 0, 0, 0, 0, 0, 0, 0]},  
{'dow': 'MON',  
'hh': '07',  
'mm': '10',  
'congestionCar': [0, 0, 0, 0, 0, 0, 0, 0, 0]},  
{'dow': 'MON',  
'hh': '07',  
'mm': '20',  
'congestionCar': [0, 0, 0, 0, 0, 0, 0, 0, 0]},  
{'dow': 'MON',  
'hh': '07',  
'mm': '30',  
'congestionCar': [0, 0, 0, 0, 0, 0, 0, 0, 0]},  
{'dow': 'MON',  
'hh': '07',  
'mm': '40',  
'congestionCar': [0, 0, 0, 0, 0, 0, 0, 0, 0]},  
{'dow': 'MON',  
'hh': '07',  
'mm': '50',  
'congestionCar': [0, 0, 0, 0, 0, 0, 0, 0, 0]}]}
```

---

# PREPROCESSING

- DataFrame의 value가 list안의 list 형태
- DataFrame의 value가 list안의 list 안의 list 형태

	startStationCode	startStationName	endStationCode	endStationName	prevStationCode	prevStationName	updnLine	directAt	data
0	610	응암역	626	공덕역	625	대흥역	1	0	[{'dow': 'MON', 'hh': '07', 'mm': '00', 'conge...
1	610	응암역	631	한강진역	625	대흥역	1	0	[{'dow': 'MON', 'hh': '07', 'mm': '00', 'conge...
2	610	응암역	647	봉화산역	625	대흥역	1	0	[{'dow': 'MON', 'hh': '07', 'mm': '00', 'conge...
3	610	응암역	648	신내역	625	대흥역	1	0	[{'dow': 'MON', 'hh': '07', 'mm': '00', 'conge...
4	613	독바위역	647	봉화산역	625	대흥역	1	0	[{'dow': 'MON', 'hh': '07', 'mm': '00', 'conge...
5	616	새절역	647	봉화산역	625	대흥역	1	0	[{'dow': 'MON', 'hh': '07', 'mm': '00', 'conge...
6	616	새절역	648	신내역	625	대흥역	1	0	[{'dow': 'MON', 'hh': '07', 'mm': '00', 'conge...
7	631	한강진역	610	응암역	627	효창공원앞역	0	0	[{'dow': 'MON', 'hh': '07', 'mm': '00', 'conge...
8	639	안암역	610	응암역	627	효창공원앞역	0	0	[{'dow': 'MON', 'hh': '07', 'mm': '00', 'conge...
9	642	상월곡역	610	응암역	627	효창공원앞역	0	0	[{'dow': 'MON', 'hh': '07', 'mm': '00', 'conge...
10	647	봉화산역	610	응암역	627	효창공원앞역	0	0	[{'dow': 'MON', 'hh': '07', 'mm': '00', 'conge...
11	647	봉화산역	616	새절역	627	효창공원앞역	0	0	[{'dow': 'MON', 'hh': '07', 'mm': '00', 'conge...
12	648	신내역	610	응암역	627	효창공원앞역	0	0	[{'dow': 'MON', 'hh': '07', 'mm': '00', 'conge...
13	648	신내역	616	새절역	627	효창공원앞역	0	0	[{'dow': 'MON', 'hh': '07', 'mm': '00', 'conge...
14	648	신내역	626	공덕역	627	효창공원앞역	0	0	[{'dow': 'MON', 'hh': '07', 'mm': '00', 'conge...

---



---

# PREPROCESSING

	dow	hh	mm	congestionTrain	up_down	subwayLine	stationName	time
0	MON	7	0	38	0	5호선	공덕역	1900-01-01 07:00:00
1	MON	7	10	51	0	5호선	공덕역	1900-01-01 07:10:00
2	MON	7	20	64	0	5호선	공덕역	1900-01-01 07:20:00
3	MON	7	30	64	0	5호선	공덕역	1900-01-01 07:30:00
4	MON	7	40	69	0	5호선	공덕역	1900-01-01 07:40:00
...	...	...	...	...	...	...	...	...
709	FRI	20	10	18	1	2호선	을지로3가역	1900-01-01 20:10:00
710	FRI	20	20	21	1	2호선	을지로3가역	1900-01-01 20:20:00
711	FRI	20	30	19	1	2호선	을지로3가역	1900-01-01 20:30:00
712	FRI	20	40	23	1	2호선	을지로3가역	1900-01-01 20:40:00
713	FRI	20	50	18	1	2호선	을지로3가역	1900-01-01 20:50:00

12540 rows × 8 columns

# PREPROCESSING

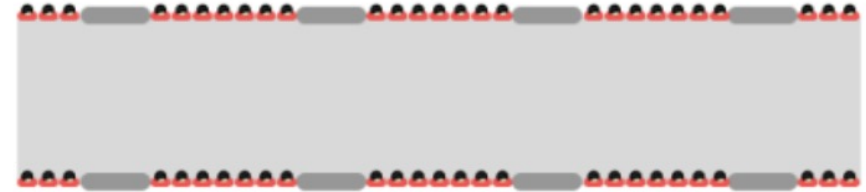
	dow	hh	mm	congestionCar	up_down	congestion_0	congestion_1	congestion_2	congestion_3	congestion_4	congestion_5	congestion_6	congestion_7	congestion_8	congestion_9	subwayLine	stationName	time
0	MON	7	0	[34, 34, 38, 52, 41, 35, 32, 42, 23, 28]	0	34	34	38	52	41	35	32	42	23	28	2호선	선릉역	1900-01-01 07:00:00
1	MON	7	10	[42, 48, 55, 62, 50, 49, 44, 32, 38]	0	42	48	55	62	50	49	49	44	32	38	2호선	선릉역	1900-01-01 07:10:00
2	MON	7	20	[52, 56, 50, 64, 55, 55, 52, 61, 46, 39]	0	52	56	50	64	55	55	52	61	46	39	2호선	선릉역	1900-01-01 07:20:00
3	MON	7	30	[66, 74, 79, 75, 86, 72, 86, 79, 71, 63]	0	66	74	79	75	86	72	86	79	71	63	2호선	선릉역	1900-01-01 07:30:00
4	MON	7	40	[53, 71, 71, 80, 65, 68, 70, 75, 66, 51]	0	53	71	71	80	65	68	70	75	66	51	2호선	선릉역	1900-01-01 07:40:00
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
4573	FRI	20	10	[15, 20, 18, 22, 21, 22, 18, 26, 29, 19]	1	15	20	18	22	21	22	18	26	29	19	2호선	을지로3가역	1900-01-01 20:10:00
4574	FRI	20	20	[16, 24, 21, 24, 22, 28, 22, 23, 27, 31]	1	16	24	21	24	22	28	22	23	27	31	2호선	을지로3가역	1900-01-01 20:20:00
4575	FRI	20	30	[19, 22, 20, 22, 22, 20, 19, 23, 26, 27]	1	19	22	20	22	22	20	19	23	26	27	2호선	을지로3가역	1900-01-01 20:30:00
4576	FRI	20	40	[21, 29, 30, 23, 27, 25, 25, 26, 29, 25]	1	21	29	30	23	27	25	25	26	29	25	2호선	을지로3가역	1900-01-01 20:40:00
4577	FRI	20	50	[18, 23, 20, 21, 22, 23, 20, 23, 26, 24]	1	18	23	20	21	22	23	20	23	26	24	2호선	을지로3가역	1900-01-01 20:50:00
26544 rows x 18 columns																		

# DATA INFO

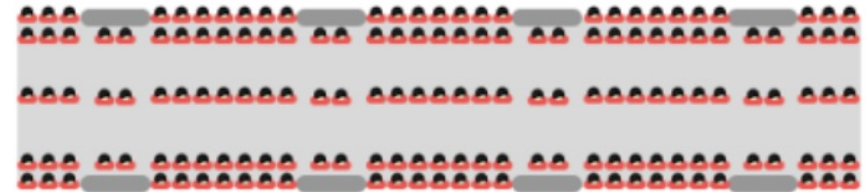
## 지하철 혼잡도 기준

지하철 승객 수	열차 내 혼잡도 환산 수치
54명	34%
94명	59%
160명	100%
240명	150%
368명	230%

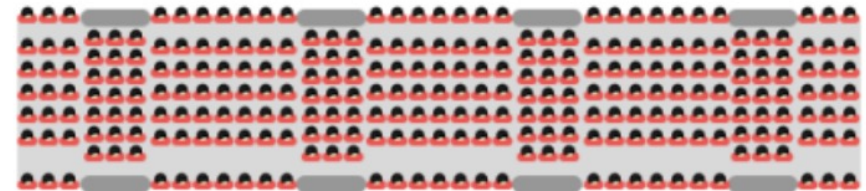
**34%** 좌석에 모두 앉은 상태



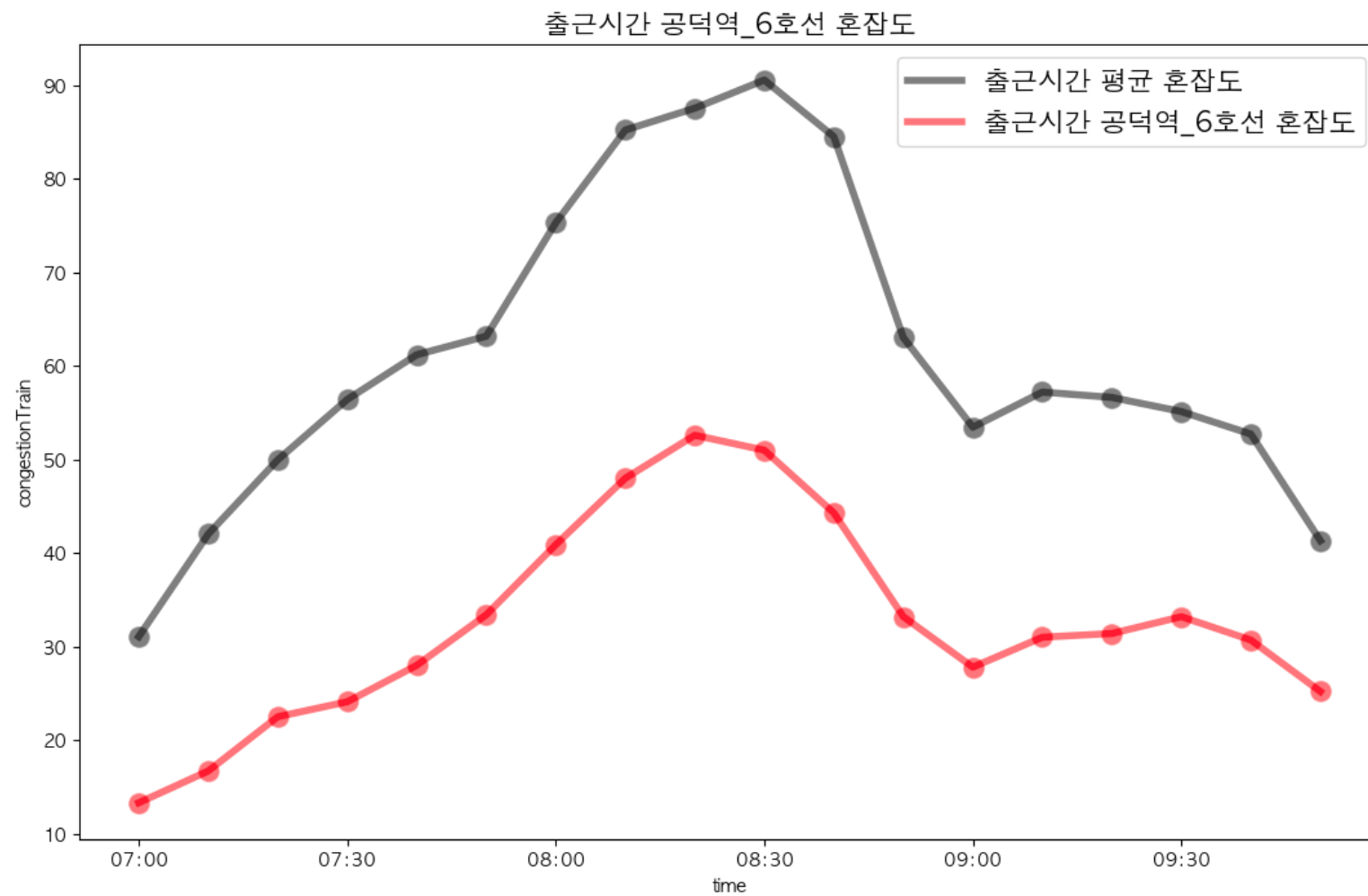
**100%** 통로에 3줄, 각 출입문에 2명씩 서 있는 상태



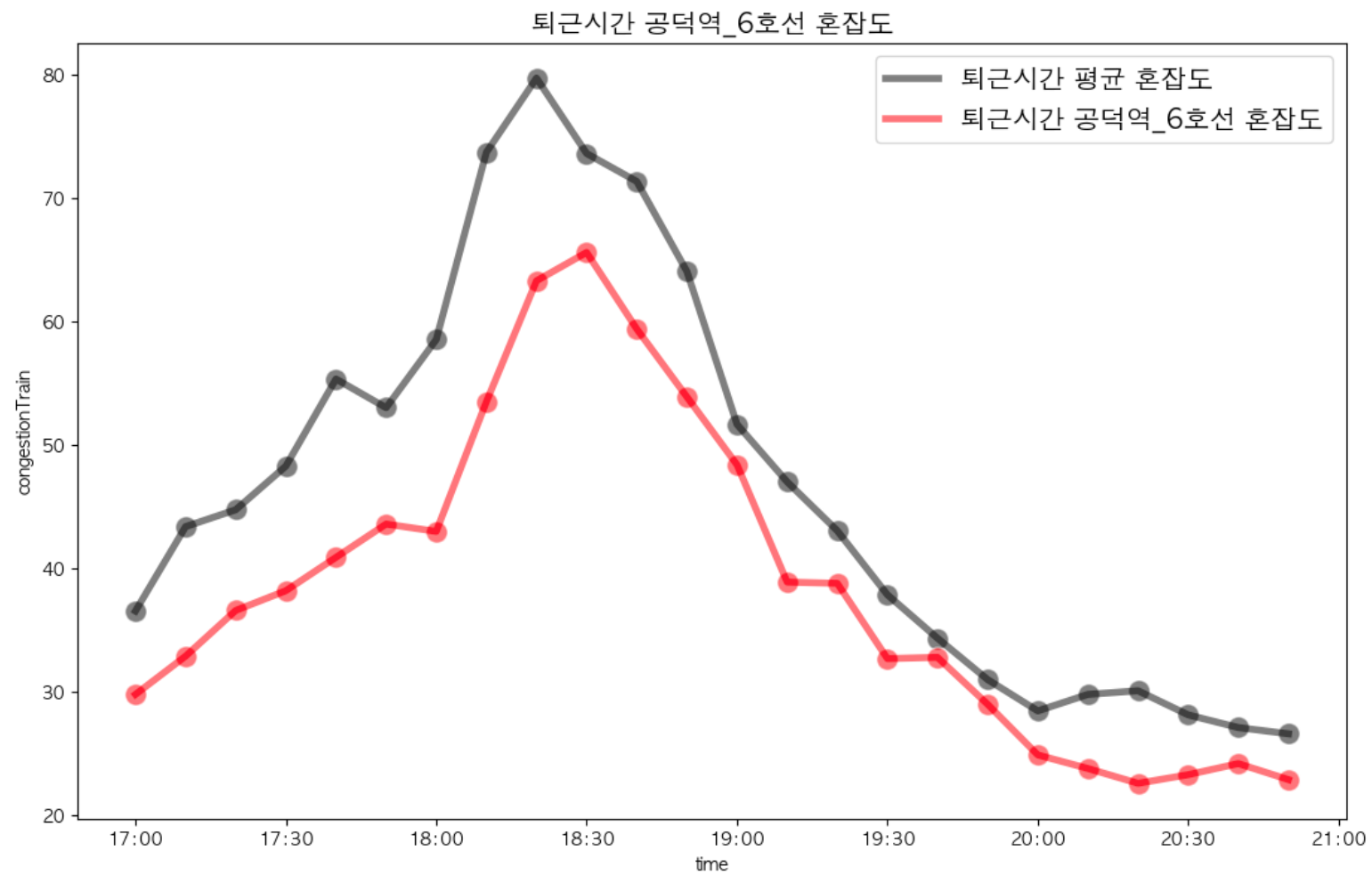
**230%** 통로에 5줄, 각 출입문에 30~40명씩 서 있는 상태

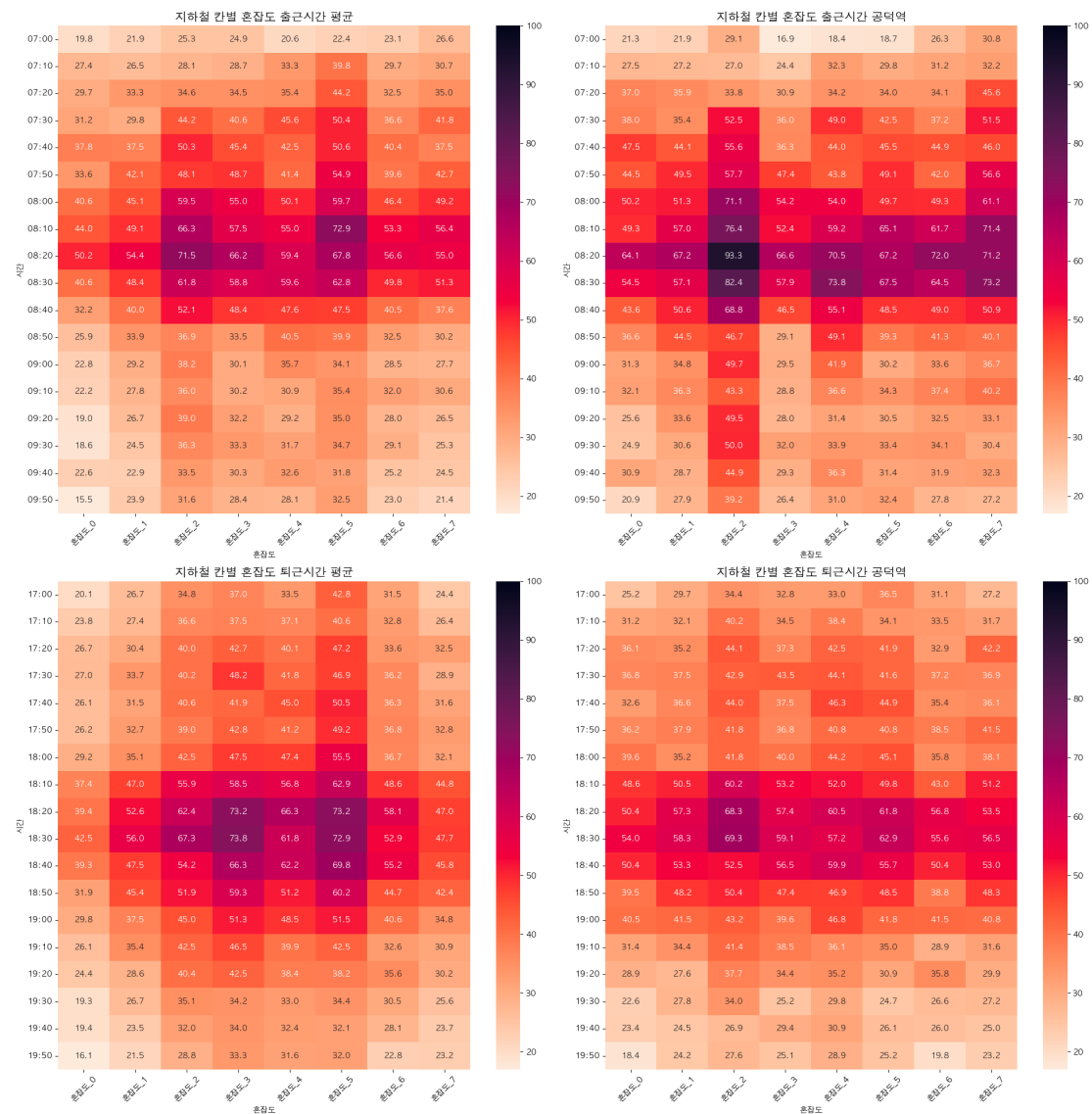


# INSIGHT

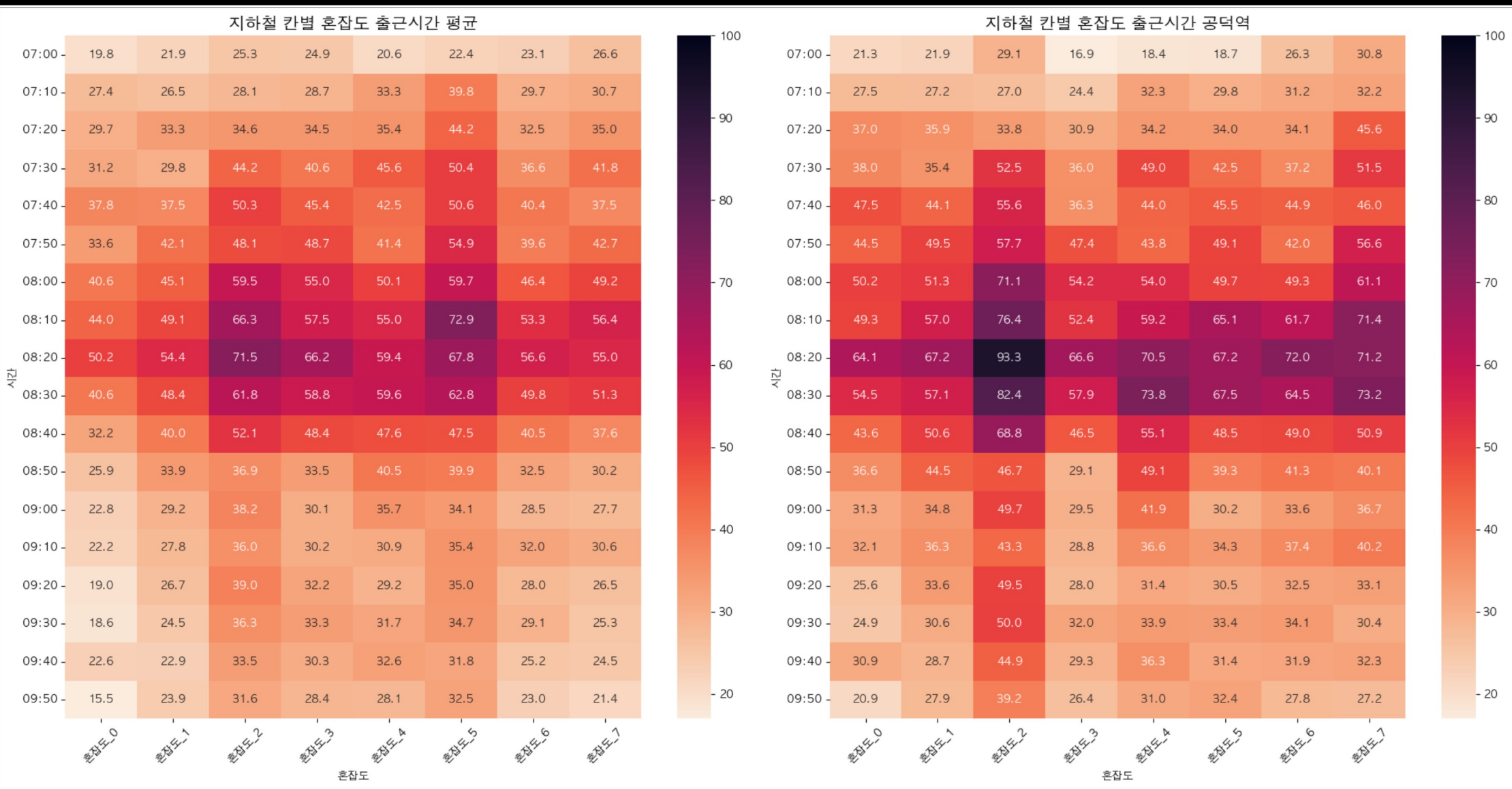


# INSIGHT

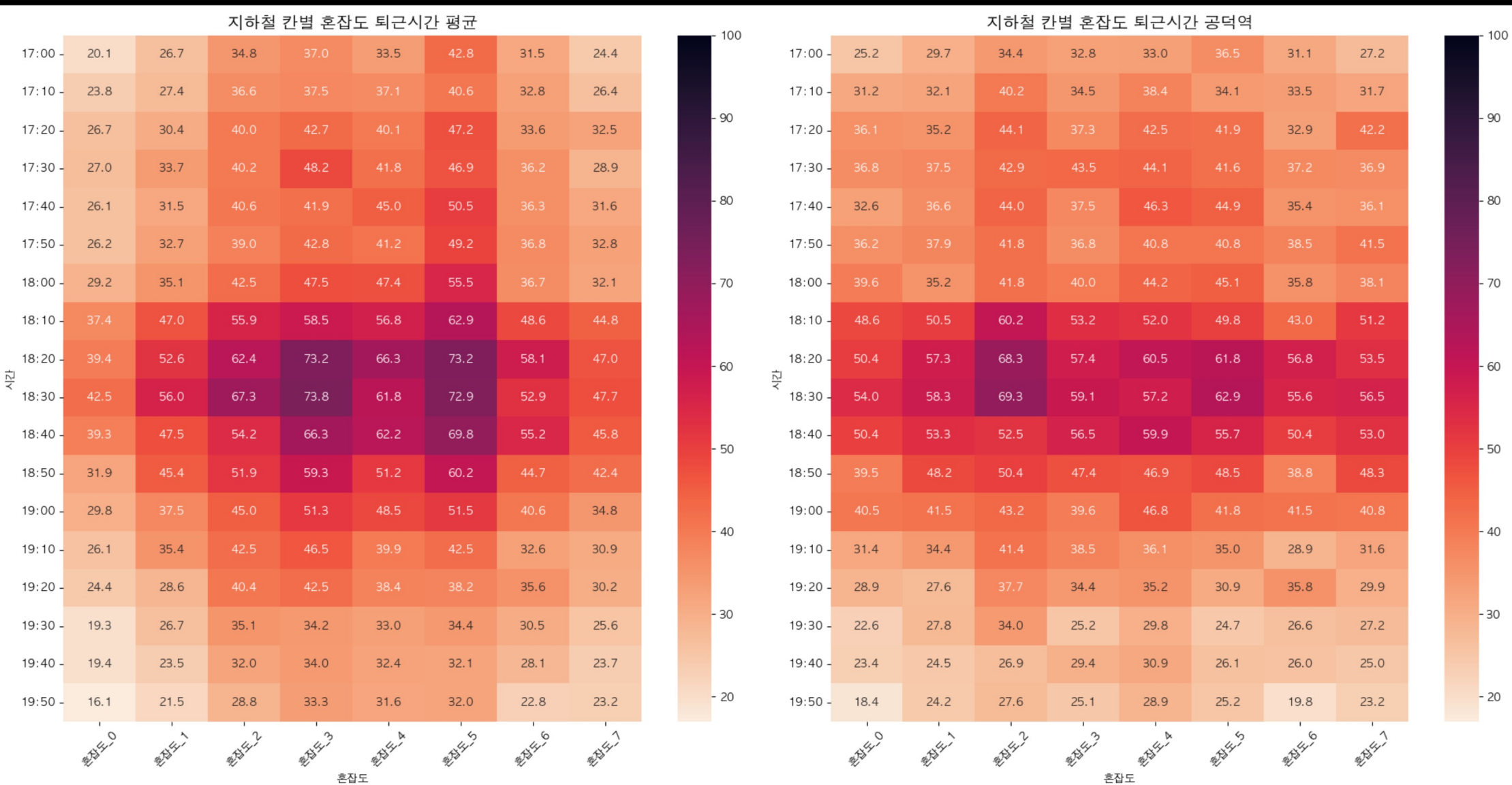




# INSIGHT



# INSIGHT





---

# CONCLUSION

```
.
├── main_pipeline.ipynb # pipeline by ipynb
├── pipe_line_py.py # pipeline by py
├── README.md
├── ## main_codes ##
│   ├── show_congest_sectional_graph.py # 칸별 혼잡도 데이터 전처리 및 그래프 생성
│   ├── show_congest_graph.py # 역별 혼잡도 데이터 전처리 및 그래프 생성
│   └── read_data.py # 칸별, 역별 혼잡도 데이터 읽어오기
├── 1. codes_get_data
│   ├── get_congest.py # 역별 혼잡도 데이터 불러오기 api 불러와서 csv로 저장
│   └── get_congest_subway.py # 칸별 혼잡도 데이터 불러오기 api 불러와서 csv로 저장
├── 2. prerprocessing
│   ├── preprocessing_main.py # 역별 혼잡도 전처리 코드
│   └── preprocessing_congest.py # 칸별 혼잡도 전처리 코드
├── data
│   └── StationCode_mapper.csv # 역명에 대한 역명 코드 매퍼
├── private
│   ├── _private_.py # api secrete key
│   └── private.md
├── ppt
│   ├── Presentation1.pptx # 발표 자료
│   └── ~$Presentation1.pptx
├── utils
│   └── utils.py
└── 7 directories, 18 files
```

---

시연

---

감사합니다.

