

인공지능 기초 탐구활동

공공 데이터로 AI 모델 만들기 서울 지하철 승하차 승객수 분석을 통한 데이터 처리 전 과정(수집→전처리→학습→해석) 해보기

31209 소현우

목차

01

탐구동기

왜 탐구주제로 선택한이유

02

탐구과정(1)

하기전 재료및데이터 수집:공공데이터포털에서 서울특별시지하철 승하차 승객수 공공데이터 다운로드

03

탐구과정(2)

데이터 전처리:중복 데이터 제거,
이상치 처리

04

탐구과정(3)

AI 모델 학습

시계열 모델을 이용한 미래 승객 수 예측

05

탐구과정 (4)

결과 해석 및 시각화

모델의 정확도 확인, 결과 시각화

06

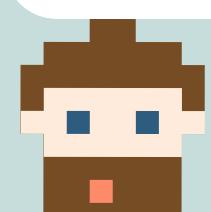
심화발전방향

문제 해결을 위한 시도와 이탐구 활동를 하면서 느낀점 그리고 앞으로의 발전방향

07

참고자료

참고책및강의들



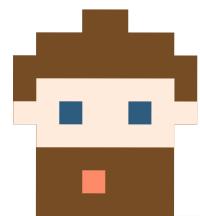
도움말

시계열 모델:시간에 따라 순차적으로 발생하는 데이터를 분석하고 예측하는데 사용되는 통계적 모델입니다. 예를 들어, 주식 가격, 기온 변화 등 있습니다.

결과 시각화:복잡한 데이터나 분석 결과를 그래프, 차트, 등으로 변환해서 쉽게 이해하고 해석할 수 있게 만듭니다.

01 | 탐구 동기

kaggle



구글클라우드(GCP): 서버, 네트워크, 데이터베이스, AI, 빅데이터, IoT 같은 기술을 개발자나 기업이 사용할 수 있도록 제공합니다.

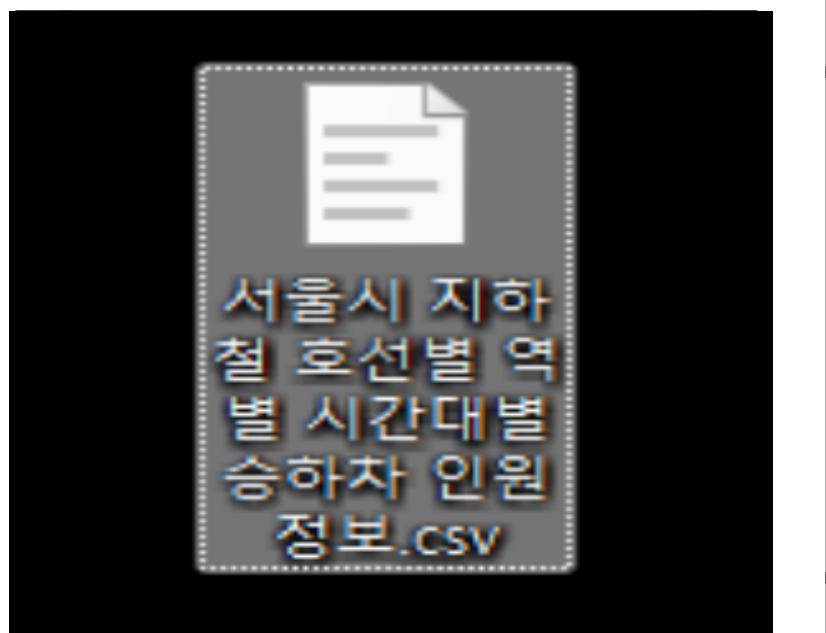
Kaggle(캐글): 전 세계 사람들이 데이터 분석, 머신러닝, 인공지능(AI) 프로젝트를 올리고, 서로의 코드를 보고 배우고 경쟁할 수 있는 데이터 과학 플랫폼입니다.

데이터 과학: 데이터를 수집하고, 정리하고, 저장하고, 쓸 수 있게 만들어주는 모든 기술과 시스템을 다루는 분야입니다.

● 탐구동기

구글 클라우드 플랫폼 관련 책을 읽고 데이터 과학과 인공지능(AI)을 결합한 탐구 활동을 찾고 있던 중 Kaggle(캐글)과 공공데이터포털 제공하는 서울 지하철 승하차 승객수 공공데이터와 무료 실행 환경을 발견하고 이를 활용한 AI 모델 개발을 탐구 주제로 정했고 그리고 서울특별시 지하철 승하차 승객수를 예측하는 모델을 만들며 데이터 처리 전 과정(수집→전처리→학습→해석)을 직접 체험하고자 하였습니다.

02 | 탐구 준비물



공공데이터포털에서
받은 파일

● 탐구 준비물

1. 서울특별시 지하철 승하차 승객수.CSV
2. kaggle(캐글)
3. 파이썬

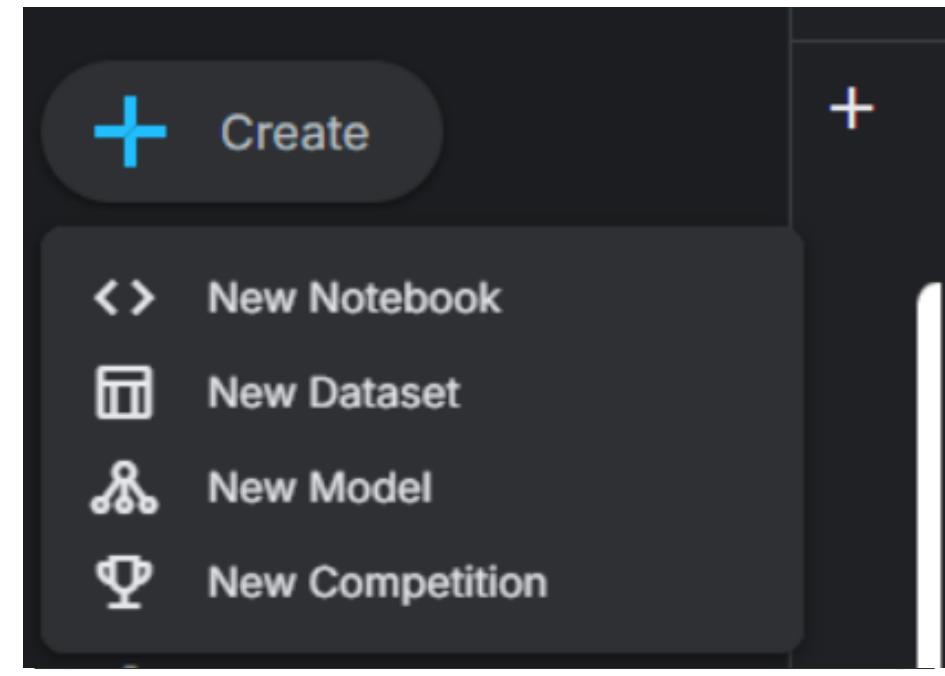
03 | 탐구과정 (1) - 데이터 수집



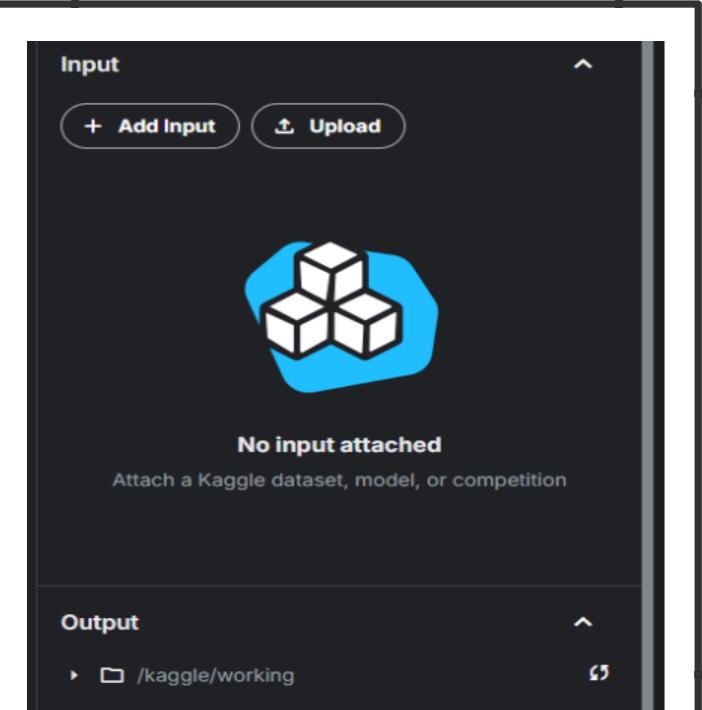
데이터 수집

1. 공공데이터포털 구글에 쳐서 들어갑니다
2. 검색창에 서울특별시_지하철 승하차 승객수라고 치고 들어갑니다
3. 다운로드를 눌러주시면 됩니다
(**주의**: 파일 이름을 영어로 바꿔주는게 좋습니다 안그러면 나중에 Kaggle(캐글)에 파일 업로드할때 파일이름이 안떠서 파일 경로 설정 코딩할때 많이 귀찮아집니다)

04 | 탐구과정 (2) - 데이터 전처리 준비



저기있는 Create 눌러주시고 첫 번째 버튼 클릭을 눌러주세요



그리고 Upload 누르기



이상치 처리: 다른 데이터들에 비해 너무 크거나, 너무 작거나, 튀는 값입니다 예를 들어 지하철 역에서 하루 평균 승객이 1만 명인데 어느 날 갑자기 100만 명이 기록됐다면 그건 시스템 오류로 생긴 이상치입니다

CSV파일: 데이터를 쉼표(,)로 구분해서 저장한 텍스트 파일입니다 엑셀처럼 표 형태의 데이터를 아주 가볍게 저장할 수 있는 형식입니다

데이터 전처리 준비

데이터 전처리는 집 안을 청소하고 물건을 제자리에 놓는 과정과 비슷합니다 예를 들어 중복 데이터 제거, 이상치 처리 같은 작업이 이에 해당합니다 이제 데이터 전처리를 하려면 Kaggle(캐글)에 다운로드한 CSV파일을 업로드 하면 됩니다

05 | 탐구과정 (2) - 데이터 전처리 시작

```
# 컬럼명 확인 및 수정 (한글 깨짐이나 불필요한 공백 제거)
df.columns = ['날짜', '호선명', '승차총승객수', '하차총승객수', '총이용객']

# 결측치 빈 값 제거 (분석에서 오류 발생하지 않게 데이터 정리)
df = df.dropna()

# '날짜' 컬럼이 있으면 문자열을 날짜(datetime)로 변환
if '날짜' in df.columns:
    df['날짜'] = pd.to_datetime(df['날짜'], errors='coerce')
```

데이터 전처리 파일 부분코드



데이터 전처리: 분석이나 모델링에 사용할 데이터를 깨끗하고 쓸 수 있는 상태로 만드는 과정입니다. 쉽게 말하면 데이터를 청소하고 정리하는 작업이에요

컬럼명: 데이터 표에서 각 열의 이름이에요

결측값: 데이터에서 값이 비어 있는 부분을 말해요. 예를 들어 표나 데이터에 값이 빠지거나 기록이 안 된 경우예요

데이터 전처리 시작

데이터 전처리 시작하기 위해서 일단 파이썬으로 코드를 짜야해야합니다. 그 후 데이터를 불러오고 **컬럼명**을 수정하거나 **결측값**을 처리하는 등의 작업을 진행합니다.

06 | 탐구과정 (3) - AI 모델 학습

```
# 날짜별 총 이용객수 합계 및 시각화
daily = df.groupby('날짜')['총이용객'].sum()
daily.plot(figsize=(12, 5), title="날짜별 총 이용객 수")
plt.xlabel("날짜")
plt.ylabel("총 이용객")
plt.grid(True)
plt.tight_layout()
# 이미지를 /kaggle/working/ 폴더에 저장
plt.savefig(os.path.join(save_path, 'daily_total_passenger.png'), bbox_inches='tight')
plt.close()

# ARIMA 모델을 위한 데이터 준비
daily = daily.sort_index()
daily = daily.asfreq('D')
daily = daily.fillna(method='ffill')

# ARIMA 모델 훈련 및 예측
model = ARIMA(daily, order=(5, 1, 0))
model_fit = model.fit()

# 향후 30일 예측
forecast = model_fit.forecast(steps=30)

# 예측 결과 시각화
plt.plot(daily.index[-30:], daily[-30:], label='실제 값')
plt.plot(pd.date_range(daily.index[-1], periods=31, freq='D')[1:], forecast, label='예측 값', color='red', linestyle='dashed')
plt.legend()
plt.title("시계열 예측: 총 이용객 수")
plt.xlabel("날짜")
plt.ylabel("총 이용객 수")
plt.tight_layout()
# 이미지를 /kaggle/working/ 폴더에 저장
plt.savefig(os.path.join(save_path, 'forecast_passenger.png'), bbox_inches='tight')
plt.close()

# 예측 결과를 CSV 파일로 저장
forecast_df = pd.DataFrame({
    '날짜': pd.date_range(daily.index[-1], periods=31, freq='D')[1:],
    '예측 총이용객수': forecast
})
forecast_df.to_csv(os.path.join(save_path, 'forecast_result.csv'), index=False)
```

AI 모델 학습 파이썬 코드

● AI 모델 학습

이제 AI 모델 학습을 해야 합니다 저는 **시계열 모델**로 AI모델을 만들고 **미래 예측 서울 지하철 승하차 승객수** 분석을 할겁니다 이걸 파이썬 코드로 만들겁니다 근데 AI모델 만들다가 AI 쪽 버그수정이 힘들어서 이건 **챗 GPT**의 도움을 좀 받았습니다 양해부탁드립니다

07 | 탐구과정 (4) - 결과해석 및 시각화

```
# 예측 결과 시각화
plt.plot(daily.index[-30:], daily[-30:], label='실제 값')
plt.plot(pd.date_range(daily.index[-1], periods=31, freq='D')[1:], forecast, label='예측 값', color='red', linestyle='dashed')
plt.legend()
plt.title("시계열 예측: 총 이용객 수")
plt.xlabel("날짜")
plt.ylabel("총 이용객 수")
plt.tight_layout()
plt.savefig(os.path.join(save_path, 'forecast_passenger.png'), bbox_inches='tight')
plt.close()

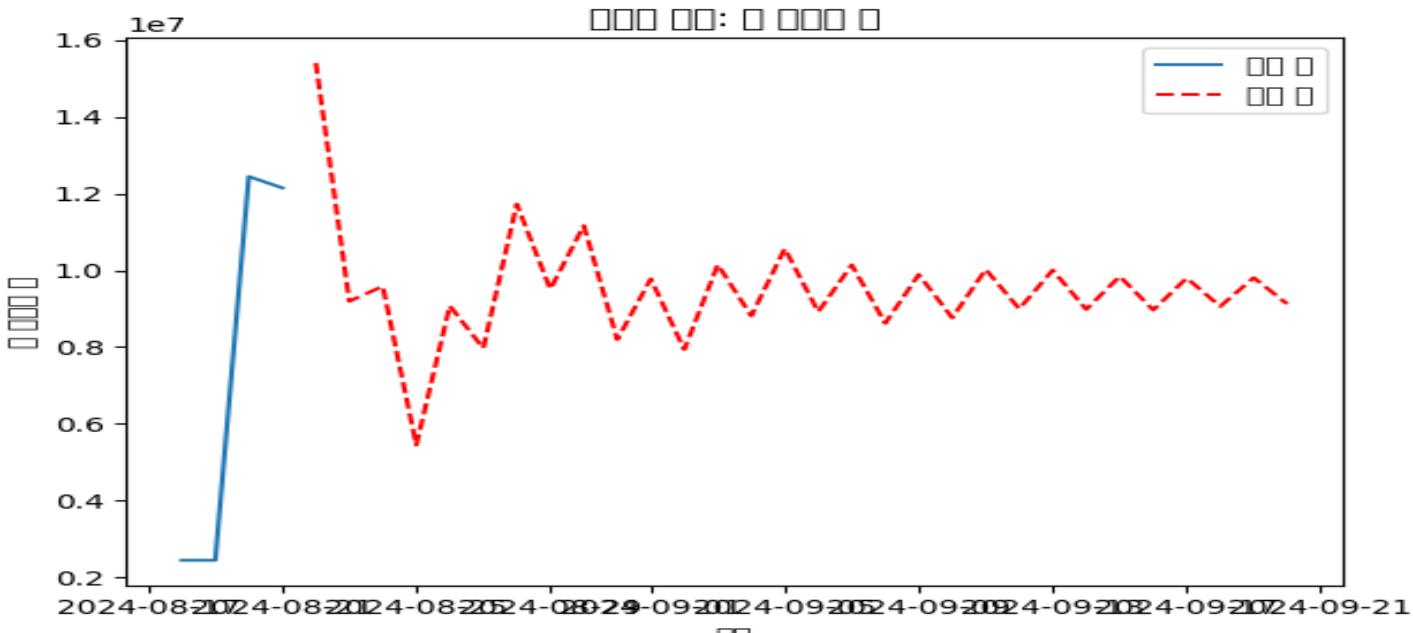
# 예측 결과를 CSV 파일로 저장
forecast_df = pd.DataFrame({
    '날짜': pd.date_range(daily.index[-1], periods=31, freq='D')[1:],
    '예측 총이용객수': forecast
})
forecast_df.to_csv(os.path.join(save_path, 'forecast_result.csv'), index=False)
```

예측 결과 시각화 코드

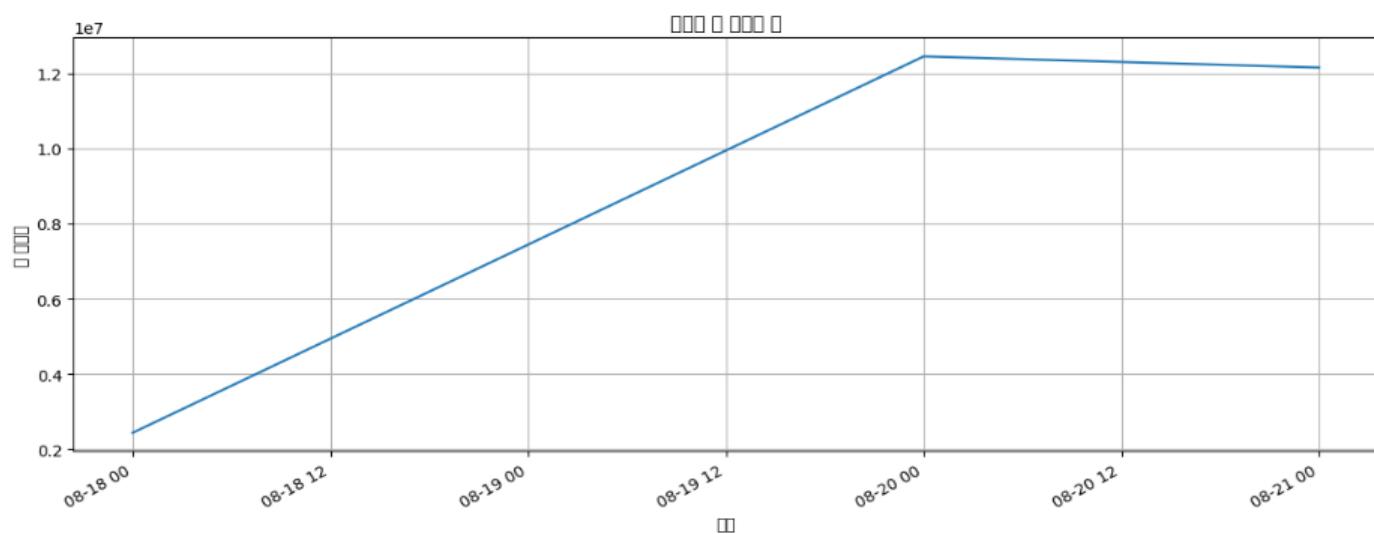
● 결과해석 및 시각화

이제 AI 모델 학습한걸 서울 지하철 승하차 승객수 분석 및 시계열 모델으로 미래 서울 지하철 승객수도 그래프로 표현할겁니다 이제 실행을 해서 나온결과 파일을 열어보면 됩니다

08 | 탐구과정 (4) - 결과해석 및 시각화



미래 예측 서울 지하철 승하차 승객수 그래프



서울 지하철 승하차 승객수 분석 그래프

● 결과해석 및 시각화

이제 나온 결과와 그래프를 보면서 데이터를 분석해봅시다

1번째 그래프는 파란선은 하루전체 승차수이고 빨간 점선은 일별 하차 인원입니다 보면 파란선은 데이터 누락을 당해서 선이 끊어졌고 빨간선은 하차 인원으로 일정 주기마다 감소 하며 이 시점은 주말인 가능성이 높습니다

2번째 그래프는 출근시간인 8시 18분 부터 9시 19분까지 지속적으로 증가하고 그이후에 후반 구간 9시 20분부터 그래프가 지속적으로 유지됩니다 이렇게 데이터공학과 AI를 결합하면 이러한 데이터를 더 잘 이해하고 예측할 수 있어 다양한 문제를 해결하는 데 도움을 줄 수 있습니다

09 | 심화발전방향 - 문제 해결을 위한 시도

```
FileNotFoundError: [Errno 2] No such file or directory: '/path/to/파일을열어_지어야할 숨겨주.csv'
/usr/local/lib/python3.11/dist-packages/pandas/io/parsers/readers.py in _read(filepath_or_buffer, **kwargs)
    619     # Create the parser.
-> 620     parser = TextFileReader(filepath_or_buffer, **kwargs)
    621 
    622     if chunksize or iterator:
    623         kwds.update(kwds_defaults)

/usr/local/lib/python3.11/dist-packages/pandas/io/parsers/readers.py in __init__(self, f, engine, **kwargs)
    1019         self._handles = IOHandles()
-> 1020         self._engine = self._make_engine(f, self._engine)
    1021 
    1022     def close(self) -> None:
    1023         ...

/usr/local/lib/python3.11/dist-packages/pandas/io/parsers/readers.py in _make_engine(self, f, engine)
    1008         if "b" not in mode:
    1009             mode += "b"
-> 1010         self._handles = get_handle(
    1011             f,
    1012             mode,
    1013             ...

/usr/local/lib/python3.11/dist-packages/pandas/_libs/common.py in get_handle(path_or_buf, mode, encoding, compression, memory_map, is_text, errors, storage_options)
    91             if iorargs.encoding and "b" not in iorargs.mode:
    92                 # Encoding
-> 93                 handle = open(
    94                     handle,
    95                     iorargs.mode,
    96                     ...

FileNotFoundError: [Errno 2] No such file or directory: '/path/to/파일을열어_지어야할 숨겨주.csv'
```

(1) 파이썬 파일 경로 오류코드

```
/tmp/ipykernel_31/867676755.py:43: UserWarning: Glyph 45216 (\N{HANGUL SYLLABLE NAL}) missing from current font.
  plt.tight_layout()
/tmp/ipykernel_31/867676755.py:43: UserWarning: Glyph 51676 (\N{HANGUL SYLLABLE JJA}) missing from current font.
  plt.tight_layout()
/tmp/ipykernel_31/867676755.py:43: UserWarning: Glyph 52509 (\N{HANGUL SYLLABLE CONG}) missing from current font.
  plt.tight_layout()
/tmp/ipykernel_31/867676755.py:43: UserWarning: Glyph 51060 (\N{HANGUL SYLLABLE T}) missing from current font.
  plt.tight_layout()
/tmp/ipykernel_31/867676755.py:43: UserWarning: Glyph 50857 (\N{HANGUL SYLLABLE YONG}) missing from current font.
  plt.tight_layout()
/tmp/ipykernel_31/867676755.py:43: UserWarning: Glyph 44061 (\N{HANGUL SYLLABLE GAEG}) missing from current font.
  plt.tight_layout()
/tmp/ipykernel_31/867676755.py:43: UserWarning: Glyph 48324 (\N{HANGUL SYLLABLE BYEOL}) missing from current font.
  plt.tight_layout()
/tmp/ipykernel_31/867676755.py:43: UserWarning: Glyph 49688 (\N{HANGUL SYLLABLE SU}) missing from current font.
  plt.tight_layout()
/usr/local/lib/python3.11/dist-packages/IPython/core/pylabtools.py:151: UserWarning: Glyph 52509 (\N{HANGUL SYLLABLE CONG}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
/usr/local/lib/python3.11/dist-packages/IPython/core/pylabtools.py:151: UserWarning: Glyph 51060 (\N{HANGUL SYLLABLE T}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
/usr/local/lib/python3.11/dist-packages/IPython/core/pylabtools.py:151: UserWarning: Glyph 50857 (\N{HANGUL SYLLABLE YONG}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
/usr/local/lib/python3.11/dist-packages/IPython/core/pylabtools.py:151: UserWarning: Glyph 44061 (\N{HANGUL SYLLABLE GAEG}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
/usr/local/lib/python3.11/dist-packages/IPython/core/pylabtools.py:151: UserWarning: Glyph 45216 (\N{HANGUL SYLLABLE NAL}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
/usr/local/lib/python3.11/dist-packages/IPython/core/pylabtools.py:151: UserWarning: Glyph 51676 (\N{HANGUL SYLLABLE JJA}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
/usr/local/lib/python3.11/dist-packages/IPython/core/pylabtools.py:151: UserWarning: Glyph 49324 (\N{HANGUL SYLLABLE BYEOL}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
/usr/local/lib/python3.11/dist-packages/IPython/core/pylabtools.py:151: UserWarning: Glyph 49588 (\N{HANGUL SYLLABLE SU}) missing from current font.
```

(2) 한국어 폰트 문제 에러 코드

문제 해결을 위한 시도

1) 파일경로오류

문제점: 파일이름을 한국어로 해서 발생한 문제입니다 이건 5 번째 페이지에서 설명했던거랑 같은 증상입니다

해결: 이건 kaggle(캐글)에서 한국어 지원을 안하기 때문에
다운한 CSV파일 이름을 영어로 바꿔주면 해결되는 간단한 문
제입니다

2):한국어 폰트 문제

문제점:kaggle(캐글)에선 한국어 폰트를 지원하지 않습니다

해결: 그래서 kaggle(캐글)에서 기본제공 폰트인 DejaVu

Sans(데자뷰 산스)를 사용하니 저런 오류가 안뜨습니다 솔직히 더 간단한 해결방법을 한글 폰트를 설치하는것이지만 다운하려고 하니 인터넷 연결이 안돼있다고 설치가 안돼서 저 방법을 사용했습니다

10 | 심화발전방향 - 문제 해결을 위한 시도

	SEOUL SUBWAY.CSV					
1						
2	2024-08-21, 38 1.8 1.0, 522162, 520400, 1047562					
3	2024-08-21, 48 1.0, 515752, 527409, 1043161					
4	2024-08-21, 68 1.0, 336194, 331413, 667687					
5	2024-08-21, 18 1.0, 160216, 159023, 319239					
6	2024-08-21, 28 1.0, 1417030, 1446635, 2863665					
7	2024-08-21, 18 1.0, 235840, 229645, 465485					
8	2024-08-21, 18 1.0, 11728, 12318, 24046					
9	2024-08-21, 625, 559, 1184, 107699, 103584, 211283					
10	2024-08-21, 18 1.0, 31107, 30757, 61864					
11	2024-08-21, 13527, 13130, 26657, 12240, 11663, 23903					
12	2024-08-21, 18 1.0, 9152, 9158, 18310					
13	2024-08-21, 18 1.0, 19758, 17631, 37389					
14	2024-08-21, 18 1.0, 3221, 2641, 5862					
15	2024-08-21, 78 1.0, 602476, 595913, 1198389					
16	2024-08-21, 18 1.0, 138649, 151756, 290405					
17	2024-08-21, 18 1.0, 59186, 54885, 114071					
18	2024-08-21, 18 1.0, 47104, 45422, 92526					
19	2024-08-21, 18 1.0, 30103, 28272, 58375					
20	2024-08-21, 18 1.0, 4181, 4067, 8248					
21	2024-08-21, 18 1.0, 27867, 26994, 54861					
22	2024-08-21, 18 1.0, 17683, 15869, 33552					
23	2024-08-21, 18 1.0, 43653, 42163, 85816					
24	2024-08-21, 18 1.0, 23163, 22769, 45932					
25	2024-08-21, 18 1.0, 7295, 7044, 14339					
26	2024-08-21, 18 1.0, 68334, 61859, 130193					
27	2024-08-21, 18 1.0, 9512, 10945, 20457					
28	2024-08-21, 18 1.0, 676691, 674965, 1351656					
29	2024-08-21, 18 1.0, 105459, 100404, 205863					
30	2024-08-21, 18 1.0, 46266, 43420, 89686					
31	2024-08-21, 18 1.0, 26738, 25943, 52681					
32	2024-08-21, 18 1.0, 297559, 302204, 599763					
33	2024-08-21, 18 1.0, 116426, 115581, 232007					
34	2024-08-21, 18 1.0, 54433, 52981, 107414					
35	2024-08-21, 18 1.0, 38708, 44086, 82794					
36	2024-08-21, 18 1.0, 209635, 210578, 420213					
37	2024-08-21, 18 1.0, 4847, 4847, 9694					
38	2024-08-21, 18 1.0, 15835, 13980, 29815					
39	2024-08-21, 18 1.0, 1376, 1517, 2893					
40	2024-08-20, 98 1.0, 118626, 118064, 236690					
41	2024-08-20, 98 1.0, 71682, 63535, 135217					
42	2024-08-20, 98 1.0, 15645, 7414, 15059					
43	2024-08-20, 98 1.0, 61339, 57735, 119074					
44	2024-08-20, 98 1.0, 55425, 54474, 109899					
45	2024-08-20, 98 1.0, 17113, 15007, 32120					
46	2024-08-20, 98 1.0, 9816, 11101, 20917					
47	2024-08-20, 98 1.0, 342742, 338960, 681702					
48	2024-08-20, 98 1.0, 108539, 103453, 211992					
49	2024-08-20, 98 1.0, 31544, 31875, 63419					
50	2024-08-20, 98 1.0, 111128, 107096, 218224					
51	2024-08-20, 98 1.0, 27383, 26770, 54153					
52	2024-08-20, 98 1.0, 20283, 18557, 38840					
53	2024-08-20, 98 1.0, 690572, 687264, 1377836					
54	2024-08-20, 98 1.0, 305422, 309410, 614832					
55	2024-08-20, 98 1.0, 30451, 28980, 59431					
56	2024-08-20, 98 1.0, 13073, 12496, 25569					
57	2024-08-20, 98 1.0, 3450, 2881, 6331					
58	2024-08-20, 98 1.0, 1453109, 1477494, 2930603					
59	2024-08-20, 98 1.0, 141153, 154710, 295863					

데이터 파일이 깨져있는 문제 사진

● 문제 해결을 위한 시도

(3):데이터 파일에 누락

문제점:공공데이터포털 사이트에서 다운한 파일이 깨져있는 문제

해결법(?):이건 공공데이터포털 문제인지 제컴에 문제인지 모르겠

어서 그래서 살릴수있는 데이터는 살려놨습니다

그래서 그래프 해석할때 그래프가 잘려 있습니다

11 | 심화발전방향 - 탐구활동을 하면서 느낀점

```
import matplotlib.pyplot as plt
import pandas as pd
import warnings
from statsmodels.tsa.arima.model import ARIMA
import matplotlib
import os

# 경고 메시지 숨기기
warnings.filterwarnings("ignore")

# 한글 폰트를 DejaVu Sans로 설정
plt.rcParams['font.family'] = 'DejaVu Sans'

# 빅데이터 설정을 Agg로 변경
matplotlib.use('Agg')

# 저장할 경로 설정
save_path = '/kaggle/working/'

# 기존 파일 삭제
files_to_delete = [
    'line_2_station_passenger.png',
    'daily_total_passenger.png',
    'forecast_passenger.png',
    'forecast_result.csv'
]

for file_name in files_to_delete:
    file_path = os.path.join(save_path, file_name)
    if os.path.exists(file_path):
        os.remove(file_path)

# CSV 파일 읽기
df = pd.read_csv("./kaggle/input/seoul-subway/Seoul subway.csv", encoding='cp949')
df.columns = ['날짜', '호선명', '승차출승객수', '하차출승객수', '총이용객']

# 컬럼의 타입을 확인
print("컬럼명 목록:", df.columns.tolist())
print(df.head())

# '날짜' 컬럼을 datetime 형식으로 변환
df['날짜'] = pd.to_datetime(df['날짜'], errors='coerce')
df = df.dropna()

# 2호선 승하차 인원 함께 시각화
if set(['호선명', '역명', '승차출승객수', '하차출승객수']).issubset(df.columns):
    line_2 = df[df['호선명'] == '2호선']
    line_2_grouped = line_2.groupby('역명')[['승차출승객수', '하차출승객수']].sum().sort_values(by='승차출승객수', ascending=False)
    line_2_grouped.plot(kind='bar', figsize=(12, 6), title='2호선 역별 승하차 인원 합계')
    plt.xlabel("역명")
    plt.ylabel("인원 수")
    plt.tight_layout()
    plt.savefig(os.path.join(save_path, 'line_2_station_passenger.png'), bbox_inches='tight')
    plt.close()

# 날짜별 총 이용객수 계산 및 시각화
daily = df.groupby('날짜')[['총이용객']].sum()
daily.plot(figsize=(12, 6), title="날짜별 총 이용객 수")
plt.xlabel("날짜")
plt.ylabel("총 이용객")
plt.grid(True)
plt.tight_layout()
plt.savefig(os.path.join(save_path, 'daily_total_passenger.png'), bbox_inches='tight')
plt.close()

# ARIMA 모델 훈련 및 예측
model = ARIMA(daily, order=(5, 1, 0))
model_fit = model.fit()

# 향후 30일 예측
forecast = model_fit.forecast(steps=30)

# 예측 결과 시각화
plt.plot(daily.index[-30:], daily[-30:], label='실제 값')
plt.plot(pd.date_range(daily.index[-1], periods=31, freq='D')[1:], forecast, label='예측 값', color='red', linestyle='dashed')
plt.legend()
plt.title("시계열 예측: 총 이용객 수")
plt.xlabel("날짜")
plt.ylabel("총 이용객 수")
plt.tight_layout()
plt.savefig(os.path.join(save_path, 'forecast_passenger.png'), bbox_inches='tight')
plt.close()

# 예측 결과 CSV 파일로 저장
forecast_df = pd.DataFrame({
    '날짜': pd.date_range(daily.index[-1], periods=31, freq='D')[1:],
    '예측_총이용객수': forecast
})
forecast_df.to_csv(os.path.join(save_path, 'forecast_result.csv'), index=False)
```

● 탐구활동을 하면서 느낀점

솔직히 처음엔 많이 어렵고 힘들었습니다 제가 구글 클라우드 데이터과학이랑 데이터 엔지니어 97가지 조언이라는 책이랑 데이터 공학강의를 보고 겨우 이걸 이해했을정도로 많이 힘들었고 파이썬 코드짤때 버그도 많이나고 AI학습 코드짤때 버그가 많이 나서 힘들었지만 그래도 해내고 나니 뿌듯하고 즐거웠습니다 처음은 많이 어렵지만 점점하다보니 재밌고 제가 데이터 공학이랑 인공지능에 관심이 많아서 포기하지 않고 끝까지 했던것 같습니다 2 사진은 이 탐구의 전체 파이썬 코드입니다 혹시 모르시는 코드 있으실까봐 전체 코드에 해석하기 쉽게 주석처리로 코드마다 해석을 넣었습니다

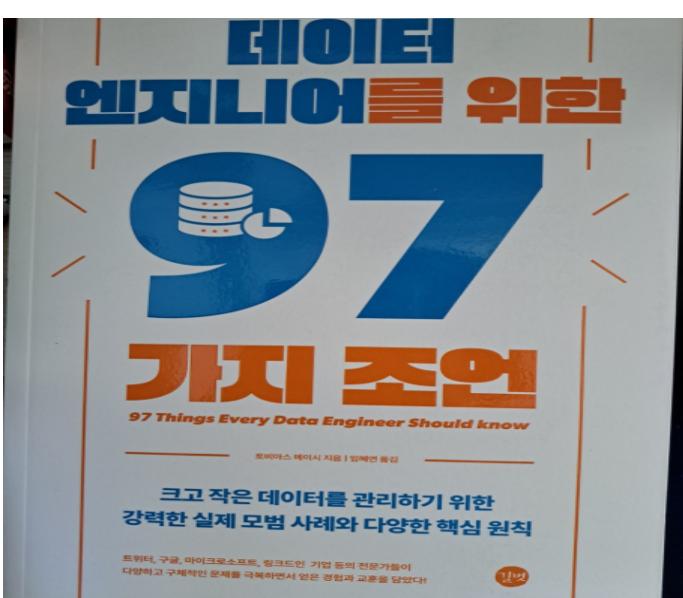
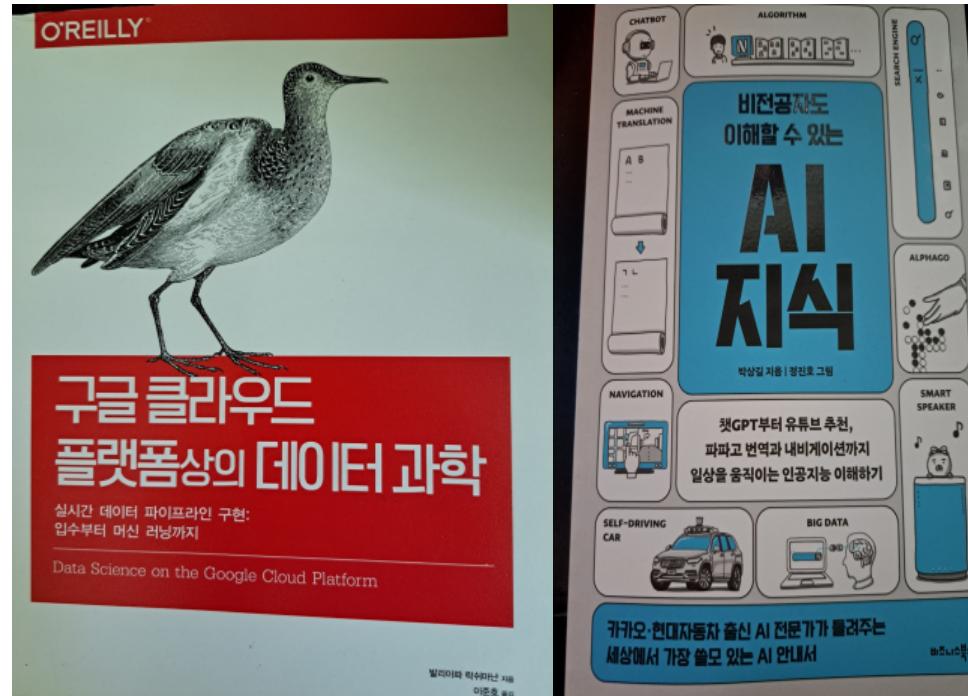
12 | 심화발전방향 - 앞으로의 발전방향



● 앞으로의 발전방향

1. 제가 대학가서 데이터 공학 전공을 하면 데이터 공학이랑 인공지능을 더 공부해서 이렇게 공공의 데이터로 나와있는 거 말고 제가 만든 데이터로 시간열 모델을 만들어 보고 싶습니다
2. 시간이 된다면 그래프 한글깨짐을 고치겠습니다 지금 이 걸 해결하려면 Kaggle(캐글)의 인터넷을 고쳐야하는데 Kaggle(캐글)의 인터넷 방식을 몰라서 못고치고 있습니다 그래야 그래프 한글폰트가 깨지지 않고 제대로 나올수 있습니다
3. 구글 클라우드로 데이터 처리 전과정을 만들겁니다

13 | 참고자료



● 참고자료

참고한 책들

1. 구글 클라우드 플랫폼상의 데이터 과학(발리아파 락쉬마난 지음)

(도와준점:데이터공학 실기와 기초를 알려주는데 도움준책)

2. 데이터 엔지니어를 위한 97가지 조언(토비아스 메이시 지음)

(도와준점:데이터 공학의 모범사례와 데이터 공학의 다양한 노하우와 아이디어를 찾는데 도움을 준 책)

도움을 준 유튜브 강의들

<https://www.youtube.com/watch?v=rEbBxxChJJQ&list=PLjh1hImDSDkdTBTGLbFXR3z0tdtpXlvla>(채널:DIP 대구 빅데이터활용 센터)

3. 비전공자도 이해할 수 있는 AI 지식(박상길 지음)

(도와준점:인공지능의 기초지식과 인공지능 종류 와 인공지능의 많은관심을 가지게 한책)

챗 GPT

(도와준점:AI 시계열 모델 코드 버그 수정을 도와주었다)

시계열 모델링크 (채널:데이터 스테이션) <https://www.youtube.com/watch?v=0ErbtP9HQA>

저의 탐구활동을 들어주셔서 감사합니다

탐구기간: 2025년 4월 28일~2025년 5월 2일

31209 소현우