기상데이터 전처리

# 2009년~2020년 전국 관측소별 기상데이터 불러오기

import pandas as pd

df1 = pd.read\_csv('dataset/weather09.csv',index\_col=0)

df2 = pd.read\_csv('dataset/weather1012.csv',index\_col=0)

df3 = pd.read\_csv('dataset/weather1314.csv',index\_col=0)

df4 = pd.read\_csv('dataset/weather15.csv',index\_col=0)

df5 = pd.read\_csv('dataset/weather1620.csv',index\_col=0)

# 데이터프레임 합치기

df = pd.concat([df1,df2,df3,df4,df5])

df.head()

# 최고기온, 평균습도, 평균지면온도, 하루총강수량 컬럼만 사용

# 피벗테이블로 바꾸어 전국 평균 계산하기

df1 = df[['tm','maxTa','avgRhm','avgTs','sumRn']]

df1.columns = map(str.lower, df1.columns)

df1['date'] = pd.to\_datetime(df1['tm'])

df1.set\_index('date',inplace=True)

df1 = df1.drop('tm',axis=1)

# 피벗테이블에서 평균 계산하기 위해 공백을 NaN값으로 바꾸고 데이터타입이 float으로 바뀜

import numpy as np

df1['maxta'] = df1['maxta'].replace('',np.nan)

df1['avgrhm'] = df1['avgrhm'].replace('',np.nan)

df1['avgts'] = df1['avgts'].replace('',np.nan)

df1['sumrn'] = df1['sumrn'].replace('',np.nan)

df1.reset\_index()

pvt = pd.pivot\_table(df1,index='date',aggfunc='mean')

# 7,8월 데이터만 사용

pvt1 = pvt.loc['2009-07':'2009-08']

pvt2 = pvt.loc['2010-07':'2010-08']

pvt3 = pvt.loc['2011-07':'2011-08']

pvt4 = pvt.loc['2012-07':'2012-08']

pvt5 = pvt.loc['2013-07':'2013-08']

pvt6 = pvt.loc['2014-07':'2014-08']

pvt7 = pvt.loc['2015-07':'2015-08']

pvt8 = pvt.loc['2016-07':'2016-08']

pvt9 = pvt.loc['2017-07':'2017-08']

pvt10 = pvt.loc['2018-07':'2018-08']

pvt11 = pvt.loc['2019-07':'2019-08']

pvt12 = pvt.loc['2020-07':'2020-08']

pvt = pd.concat([pvt1,pvt2,pvt3,pvt4,pvt5,pvt6,pvt7,pvt8,pvt9,pvt10,pvt11,pvt12])

# 2009-2019년 데이터와 2010-2020년 데이터로 나눠서 2010-2020년의 전년 동월 평균 대비 증감율 출력하기

d9 = pvt['2009':'2019']

d10 = pvt['2010':'2020']

d9.reset\_index(inplace=True)

d10.reset\_index(inplace=True)

# m(월), y(년) 기준으로 그룹화하여 연도별 7,8월 평균 계산

d9['m'] = d9.date.dt.month

d9['y'] = d9.date.dt.year

d10['m'] = d10.date.dt.month

d10['y'] = d10.date.dt.year

grouped = d9.groupby(['y','m']).mean()

d9\_mean = pd.DataFrame(grouped)

d9\_mean.reset\_index(inplace=True)

# 당해 년도 데이터와 전년 동월 평균 데이터를 합치기

# on에 y,m 순서 바꾸면 데이터 달라짐

d10 = d10.merge(d9\_mean, on=['y','m'])

# 년도가 다르기때문에 인덱스가 같은 상태에서 merge 먼저 하고 다시 date를 인덱스로 만들어 줌

d10.set\_index('date',inplace=True)

d10 # \_x: 2010~2020년 일 평균 데이터, \_y: 전년(2009~2019년) 동월 평균 데이터

# \_diff: 2010~2020년의 전년 동월 평균 대비 증감량, \_pct: 2010~2020년의 전년 동월 평균 대비 증감율

d10['avgrhm\_diff'] = d10['avgrhm\_x']-d10['avgrhm\_y']

d10['avgts\_diff'] = d10['avgts\_x']-d10['avgts\_y']

d10['maxta\_diff'] = d10 ['maxta\_x']-d10['maxta\_y']

d10['sumrn\_diff'] = d10['sumrn\_x']-d10['sumrn\_y']

d10['avgrhm\_pct'] = (d10['avgrhm\_x']-d10['avgrhm\_y'])/d10['avgrhm\_y']

d10['avgts\_pct'] = (d10['avgts\_x']-d10['avgts\_y'])/d10['avgts\_y']

d10['maxta\_pct'] = (d10['maxta\_x']-d10['maxta\_y'])/d10['maxta\_y']

d10['sumrn\_pct'] = (d10['sumrn\_x']-d10['sumrn\_y'])/d10['sumrn\_y']

d10 = d10.round(2)

##### - 2010~2020년의 전년 동월 평균 대비 증감량

d\_diff = d10[['avgrhm\_diff', 'avgts\_diff', 'maxta\_diff', 'sumrn\_diff']]

##### - 2010~2020년의 전년 동월 평균 대비 증감율

d\_pct = d10[['avgrhm\_pct','avgts\_pct','maxta\_pct','sumrn\_pct']]

전력데이터 전처리

# 최대전력 데이터 모으기

import pandas as pd

df = pd.DataFrame()

for i in range(1,25): #2009년7월,8월~2020년7월8월

df1 = pd.read\_excel(f'data/최대전력/{i}.xls')

df1 = df1[::-1]

df2 = df.append(df1)

df = df2

# 중간에 날자가아닌 다른값이 들어간, null값 삭제.

df = df[df['일시'].notnull()]

#최대전력만 가져오기

df = df[['일시','최대전력(만kW)']]

df = df.rename({'일시':'date','최대전력(만kW)':'금년'},axis =1)

df = df.reset\_index()

df = df.drop('index',axis=1)

date = pd.DataFrame()

for i in [2009,2010,2011,2012,2013,2014,2015,2016,2017,2018,2019,2020]:

d = pd.date\_range(start = f'{i}-07-01', end = f'{i}-08-31', freq = 'D')

d = pd.DataFrame(d)

date1 = date.append(d)

date = date1

date = date.reset\_index()

date = date.drop('index',axis=1)

df['date'] = date

df = df.set\_index('date')

#점유율 자료 가져오기

use = pd.DataFrame()

for i in range(1,13):

use\_read = pd.read\_excel(f'data/점유율/점유율/{i}.xlsx')

use\_read = use\_read.iloc[28:30] #7,8월 점유율 추출

use1 = use.append(use\_read)

use = use1

use = use[['Unnamed: 2']]

li = []

for i in [2009,2010,2011,2012,2013,2014,2015,2016,2017,2018,2019,2020]:

i7 = f'{i}-07-01'

i8 = f'{i}-08-01'

li.append(i7)

li.append(i8)

use['date'] = li

use.date = use.date.astype('datetime64')

use = use.set\_index('date')

use = use.rename(columns = {'Unnamed: 2' : '가정용점유율(%)'})

#당일 전력사용량과 점유율 합치기

all\_data = df.join(use, how='left')

all\_data = all\_data.fillna( method = 'ffill')

all\_data['금년'] = all\_data['금년'].str.replace(',','')

all\_data['금년'] = all\_data['금년'].astype('float')

all\_data['가정용점유율(%)'] = all\_data['가정용점유율(%)'].astype('float')

all\_data['가정용전력'] = all\_data['금년']\*all\_data['가정용점유율(%)']/100

- 데이터저장

all\_data.to\_excel('data/가정전력.xlsx')

# 가정용 전력 사용 증감률 계산하기(전년 동월 평균 대비)

import pandas as pd

df = pd.read\_csv('dataset/가정전력.csv', parse\_dates = ['date'], index\_col='date')

# 2009~2019 7월 8월 월평균 전력 사용량

mean\_list7 = []

mean\_list8 = []

for i in [2009,2010,2011,2012,2013,2014,2015,2016,2017,2018,2019]:

mean\_list7.append(df.loc[f'{i}-7']['가정용전력'].mean())

mean\_list8.append(df.loc[f'{i}-8']['가정용전력'].mean())

# 2009년~2020년 7월 8월 일자별 전력 사용량

e\_list7 = []

e\_list8 = []

for i in [2010,2011,2012,2013,2014,2015,2016,2017,2018,2019,2020]:

e\_list7.append(df.loc[f'{i}-7']['가정용전력'].values)

e\_list8.append(df.loc[f'{i}-8']['가정용전력'].values)

# 전년 7~8월 평균 대비 금년 7~8월의 일자별 전력 사용 증감량

li7 = []

li8 = []

for j in range(0,11):

for i in range(0,31):

a = (e\_list7[j][i]-mean\_list7[j])/mean\_list7[j]\*100

li7.append(a)

b = (e\_list8[j][i]-mean\_list8[j])/mean\_list8[j]\*100

li8.append(b)

# 계산한 증감률을 데이터프레임으로 만들기

pi1 = df['2010':].copy()

pi1 = pi1.reset\_index()

pi1['month'] = pi1.date.dt.month

pi\_m = pd.pivot\_table(pi1, index = ['month','date'])

pi\_m['mean'] = li7+li8

df\_mean = pi\_m.reset\_index().set\_index('date').drop(['month','가정용전력','가정용점유율','금년'],axis=1)

df\_mean = df\_mean.sort\_index() # date 정렬

- 데이터저장

df\_mean.to\_csv('dataset/증감률.csv')

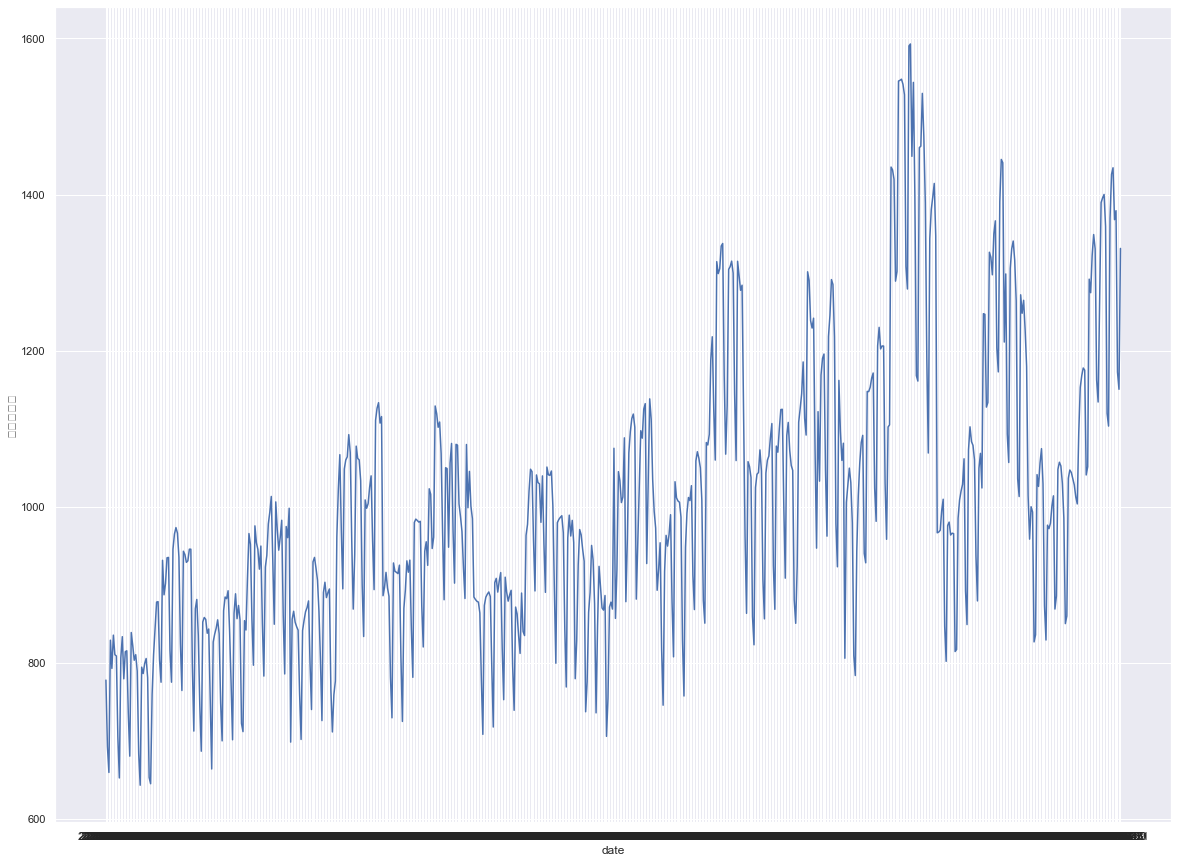
전력데이터 그래프

import seaborn as sns

import matplotlib.pyplot as plt

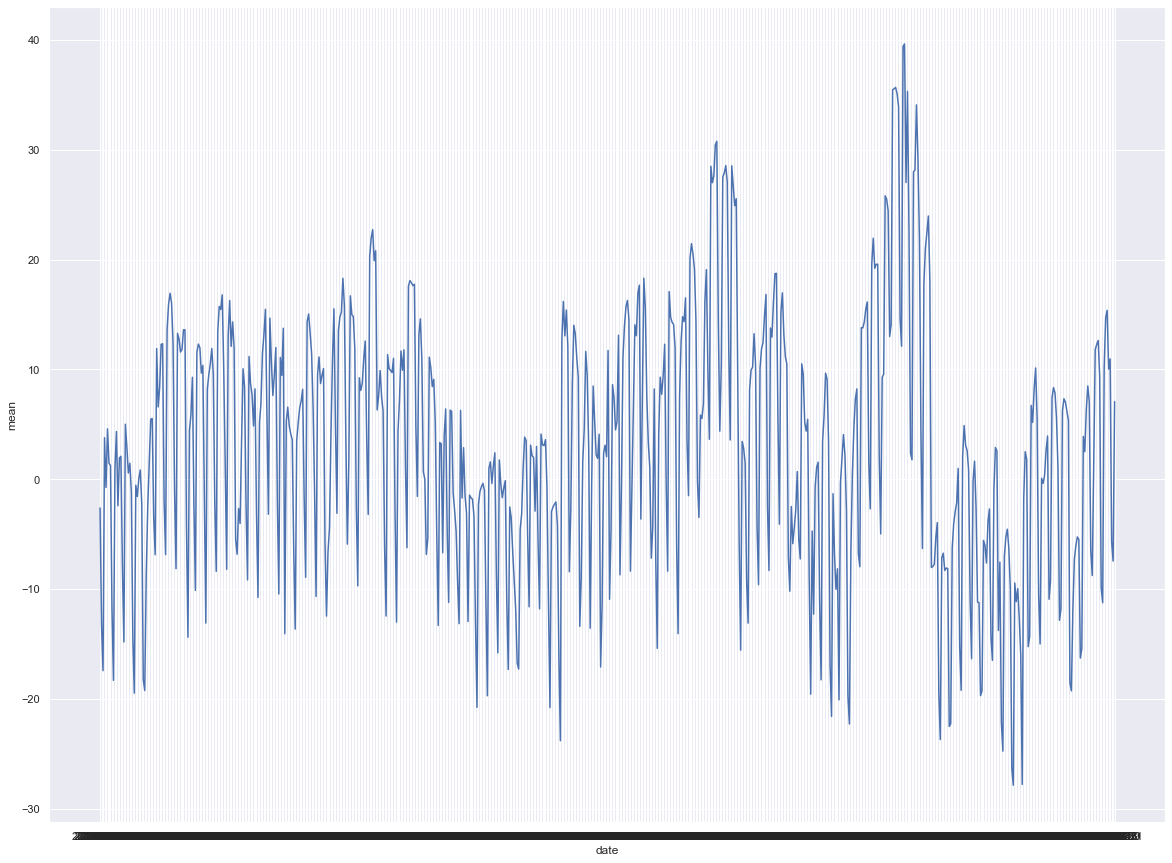
sns.set(rc={'figure.figsize':(20,15)})

sns.lineplot(x='date',y='가정용전력',data=df)



sns.set(rc={'figure.figsize':(20,15)})

sns.lineplot(x='date',y='mean',data=df)



df[df['mean']>=20] #증가율 20% 이상 데이터 출력