미세먼지 데이터 전처리

In [1]:

```
import pandas as pd
import numpy as np
```

2016년 1분기 미세먼지 데이터 확인

In [2]:

```
df = pd.read_csv('2016/2016년 1분기.csv', encoding = 'cp949')
```

In [3]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 698880 entries, 0 to 698879
Data columns (total 11 columns):
 #   Column  Non-Null Count   Dtype  
---  ------  --------------   -----  
 0   지역      698880 non-null  object 
 1   측정소코드   698880 non-null  int64  
 2   측정소명    698880 non-null  object 
 3   측정일시    698880 non-null  int64  
 4   SO2     672372 non-null  float64
 5   CO      672071 non-null  float64
 6   O3      676046 non-null  float64
 7   NO2     680038 non-null  float64
 8   PM10    681157 non-null  float64
 9   PM25    305801 non-null  float64
 10  주소      698880 non-null  object 
dtypes: float64(6), int64(2), object(3)
memory usage: 58.7+ MB
```

데이터 전처리

In [4]:

```
df['시도'] = df['지역'].apply(lambda x:x[:2])
df['측정일시'] = df['측정일시'].astype('str')
df['일시'] = df['측정일시'].apply(lambda x:x[:8])
df = df.drop(['지역','측정소코드','측정소명','주소','측정일시'], axis = 1)
df = df.replace(0, np.NaN)
```

기하평균 함수 적용

In [5]:

```
def geo_mean(iterable):
    a = np.array(iterable)
    a = a[~np.isnan(a)]
    a = np.log(a)
    return np.exp(a.sum()/len(a))
```

In [6]:

```
df = df.groupby(['시도','일시']).agg(geo_mean)
```

```
c:\app\python37\lib\site-packages\ipykernel_launcher.py:5: RuntimeWarning: invalid value
encountered in double_scalars
```

```
encountered in double_scalars
  """
```

In [7]:

```
df.info()
df.head()
```

```
<class 'pandas.core.frame.DataFrame'>
MultiIndex: 1547 entries, ('강원', '20160101') to ('충북', '20160331')
Data columns (total 6 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   SO2     1547 non-null   float64
 1   CO      1547 non-null   float64
 2   O3      1547 non-null   float64
 3   NO2     1547 non-null   float64
 4   PM10    1547 non-null   float64
 5   PM25    1456 non-null   float64
dtypes: float64(6)
memory usage: 76.5+ KB
```

Out[7]:

		SO2	CO	O3	NO2	PM10	PM25
시도	일시						
강원	20160101	0.003358	0.717610	0.012746	0.016153	50.964868	42.369581
	20160102	0.003589	0.617421	0.017016	0.013722	46.711979	37.520386
	20160103	0.004811	0.882757	0.013789	0.016651	64.950720	53.703037
	20160104	0.003965	0.650780	0.026051	0.014416	61.007831	46.295302
	20160105	0.002832	0.435089	0.020151	0.011488	24.346586	15.479883

각 년도 미세먼지 데이터 전처리 및 병합

In [8]:

```
temp = pd.DataFrame(columns=['시도','일시','SO2','CO','O3','NO2','PM10','PM25'])

for i in [2016,2017,2018]:

    for j in [1,2,3,4]:

        # 데이터 불러오기
        df = pd.read_csv(f'{i}/{i}년 {j}분기.csv', encoding = 'cp949')

        # 시도 컬럼 생성
        df['시도'] = df['지역'].apply(lambda x:x[:2])

        # 측정일시 칼럼 시간 -> 일 단위로 변경
        df['측정일시'] = df['측정일시'].astype('str')
        df['일시'] = df['측정일시'].apply(lambda x:x[:8])

        # 불필요한 칼럼 제거
        df = df.drop(['지역','측정소코드','측정소명','주소','측정일시'], axis = 1)

        # 0인 값 결측치로 처리
        df = df.replace(0, np.NaN)

        # 일 평균 데이터로 변환
        # 기하평균 적용
        df = df.groupby(['시도','일시']).agg(geo_mean).reset_index()

        # 일시 칼럼 타입 변경
        df['일시'] = pd.to_datetime(df['일시'], format='%Y-%m-%d')

        temp = temp.append(df, ignore_index = True)

        pass
```

```
    pass

temp.to_csv('dust.csv', encoding = 'utf8', index = False)
```

```
c:\app\python37\lib\site-packages\ipykernel_launcher.py:5: RuntimeWarning: invalid value
encountered in double_scalars
  """
```

결측치 확인 및 처리

In [9]:

```
df = pd.read_csv('dust.csv')
```

결측치 확인

In [10]:

```
print(df[df['SO2'].isnull()==True])
print(df[df['CO'].isnull()==True])
print(df[df['O3'].isnull()==True])
print(df[df['NO2'].isnull()==True])
print(df[df['PM10'].isnull()==True])
print(df[df['PM25'].isnull()==True])
```

```
Empty DataFrame
Columns: [시도, 일시, SO2, CO, O3, NO2, PM10, PM25]
Index: []
Empty DataFrame
Columns: [시도, 일시, SO2, CO, O3, NO2, PM10, PM25]
Index: []
Empty DataFrame
Columns: [시도, 일시, SO2, CO, O3, NO2, PM10, PM25]
Index: []
Empty DataFrame
Columns: [시도, 일시, SO2, CO, O3, NO2, PM10, PM25]
Index: []
Empty DataFrame
Columns: [시도, 일시, SO2, CO, O3, NO2, PM10, PM25]
Index: []
       시도          일시       SO2        CO        O3       NO2        PM10  PM25
819    세종  2016-01-01  0.003247  1.038007  0.010429  0.024375  59.129622   NaN
820    세종  2016-01-02  0.003851  0.920347  0.012311  0.025407  50.912560   NaN
821    세종  2016-01-03  0.003356  1.006733  0.011697  0.023761  65.312262   NaN
822    세종  2016-01-04  0.005736  0.861132  0.027744  0.021746  57.256727   NaN
823    세종  2016-01-05  0.005067  0.652506  0.017043  0.024010  31.917832   NaN
...    ..         ...       ...       ...       ...       ...        ...   ...
2392   세종  2016-04-27  0.001594  0.402582  0.037483  0.014116  39.137141   NaN
2393   세종  2016-04-28  0.001864  0.348411  0.033638  0.014576  31.160152   NaN
2394   세종  2016-04-29  0.002174  0.413231  0.035948  0.014328  52.114254   NaN
2395   세종  2016-04-30  0.002731  0.465045  0.050157  0.011600  56.072313   NaN
16845  제주  2018-08-23  0.001067  0.138722  0.023250  0.001788   6.000000   NaN

[122 rows x 8 columns]
```

세종 초미세먼지 결측치 처리

In [11]:

```
# 2016년 1분기 및 2분기(1~4월) 데이터 불러오기

dust_1Q = pd.read_csv('2016/2016년 1분기.csv', encoding = 'cp949')
dust_2Q = pd.read_csv('2016/2016년 2분기.csv', encoding = 'cp949')

dust =pd.concat([dust_1Q,dust_2Q],ignore_index= True)
```

In [12]:

In [12]:

```
# 시도와 시군구 칼럼 추가

dust['측정일시'] = dust['측정일시'].astype('str')
dust['일시'] = pd.to_datetime(dust['측정일시'].apply(lambda x:x[:8]), format='%Y-%m-%d')
dust['시도'] = dust['지역'].agg(lambda x:x[:2])
dust['시군구'] = dust['지역'].agg(lambda x:x[3:5])
```

In [13]:

```
dust[(dust['시도']== '대전') | (dust['시군구'] == '청주')]
dust = dust.replace(0, np.NaN)
PM25_sejong = dust[['일시','PM25']].groupby('일시').agg(geo_mean).reset_index()
PM25_sejong['시도'] = '세종'
```

```
c:\app\python37\lib\site-packages\ipykernel_launcher.py:5: RuntimeWarning: invalid value
encountered in double_scalars
  """
```

In [14]:

```
PM25_sejong['일시'] = PM25_sejong['일시'].astype('str')
```

In [15]:

```
df = pd.merge(df, PM25_sejong, how='left', on=['시도','일시'])
df['PM25'] = df['PM25_x'].replace(np.NaN, df['PM25_y'])
df = df.drop(['PM25_x','PM25_y'], axis = 1)
```

In [16]:

```
df[df['PM25'].isnull() == True]
```

Out[16]:

	시도	일시	SO2	CO	O3	NO2	PM10	PM25
16845	제주	2018-08-23	0.001067	0.138722	0.02325	0.001788	6.0	NaN

제주 초미세먼지 결측치 처리

In [17]:

```
jeju = df[(df['시도'] =='제주')]
```

In [18]:

```
geo_mean(jeju[jeju['일시'].str.startswith('2018-08')]['PM25'])
```

Out[18]:

```
11.339777817683828
```

In [19]:

```
df = df.fillna(11.339777817683828)
```

In [20]:

```
df[df['PM25'].isnull() == True]
```

Out[20]:

	시도	일시	SO2	CO	O3	NO2	PM10	PM25

전처리 된 데이터 저장

In [21]:

```
df.to_csv('dust_Fixed.csv')
```