

```
In [1]: import pandas as pd
import numpy as np
import os, sys
from datetime import datetime
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
from matplotlib.font_manager import FontProperties
myfont = FontProperties(fname='C:\Windows\Fonts\msjh.ttc', size=14)
sns.set(font=myfont.get_name())
```

```
In [2]: #讀檔
datas = os.listdir('data')
df = pd.DataFrame()
for i, name in enumerate(datas):
    df = pd.concat([df, pd.read_csv('data/'+name)], ignore_index=True)
```

```
In [3]: #臺中市
df_tc = df[df['SiteName'].str.contains("臺中市")]
df_tc['timestamp'] = pd.to_datetime(df_tc['timestamp'], format='%Y-%m-%d %H:%M:%S')

#臺北市
df_tp = df[df['SiteName'].str[0:2]=="市立"]
df_tp['timestamp'] = pd.to_datetime(df_tp['timestamp'], format='%Y-%m-%d %H:%M:%S')

# 臺中市時平均PM 2.5濃度
df_tc = df_tc.groupby(by = [df_tc['timestamp'].dt.day, df_tc['timestamp'].dt.hour])
mean_tc = df_tc.mean()
mean_tc.index.names = ['dath', 'hour']
mean_tc.columns = ['PM25_TC']

#臺北市逐時平均PM 2.5濃度
df_tp = df_tp.groupby(by = [df_tp['timestamp'].dt.day, df_tp['timestamp'].dt.hour])
mean_tp = df_tp.mean()
mean_tp.index.names = ['dath', 'hour']
mean_tp.columns = ['PM25_TP']

#合併
mean = pd.merge(mean_tc, mean_tp, on=['dath', 'hour'])
index = []
for i in range(1,13):
    for j in range(24):
        index.append('{i}th {j}:00'.format(i=i,j=j))
mean.index = index
mean.index.name = 'time'
mean
```

```
1th 0:00  9.677046  25.141860
1th 1:00  10.247142  24.093677
1th 2:00  10.378924  23.230068
1th 3:00  9.704525  23.479218
1th 4:00  9.669014  22.864583
...
12th 19:00  19.646087  2.452282
12th 20:00  20.528662  5.880266
12th 21:00  20.993433  2.509569
12th 22:00  19.911189  5.567130
12th 23:00  18.556846  4.457500
```

288 rows × 2 columns

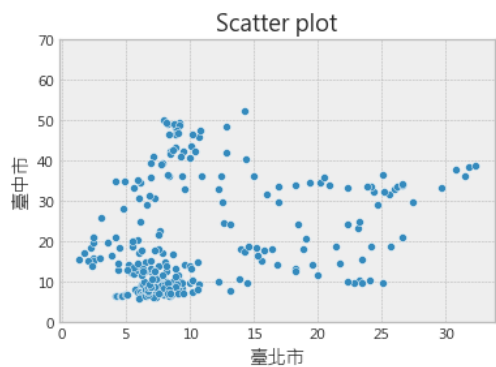
```
In [4]: mean.index.name = 'time'
mean
```

```
Out[4]:
```

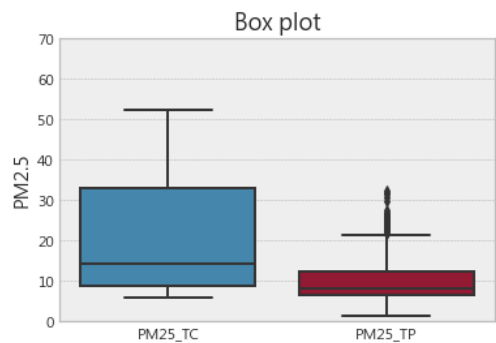
	PM25_TC	PM25_TP
time		
1th 0:00	9.677046	25.141860
1th 1:00	10.247142	24.093677
1th 2:00	10.378924	23.230068
1th 3:00	9.704525	23.479218
1th 4:00	9.669014	22.864583
...
12th 19:00	19.646087	2.452282
12th 20:00	20.528662	5.880266
12th 21:00	20.993433	2.509569
12th 22:00	19.911189	5.567130
12th 23:00	18.556846	4.457500

288 rows × 2 columns

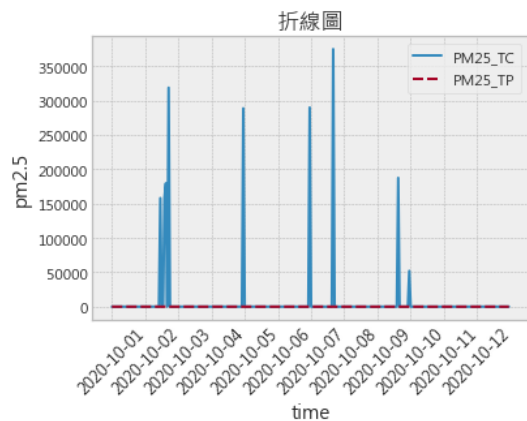
```
In [5]: plt.figure()
plt.style.use('bmh')
sns.scatterplot(data = mean, x = 'PM25_TP', y = 'PM25_TC')
plt.ylim([0,70])
plt.xlabel('臺北市')
plt.ylabel('臺中市')
plt.title('Scatter plot')
plt.show()
```



```
In [6]: plt.figure()
plt.style.use('bmh')
sns.boxplot(data = mean)
plt.ylim([0,70])
plt.ylabel('PM2.5')
plt.title('Box plot')
plt.show()
```



```
In [7]: #沒調整 yLim的折線圖
fig ,ax = plt.subplots()
sns.lineplot(data = mean)
ax.set_xticks(['{i}th 0:00'.format(i=i) for i in range(1,13)])
ax.set_xticklabels(['2020-10-%02d' % i for i in range(1,13)],rotation=45, fontsize=12)
plt.ylabel("pm2.5")
plt.title("折線圖")
plt.show()
```



```
In [8]: #有調整 yLim的折線圖
fig ,ax = plt.subplots()
xlab = ['{i}th 0:00'.format(i=i) for i in range(1,13)]
sns.lineplot(data = mean)
ax.set_xticks(['{i}th 0:00'.format(i=i) for i in range(1,13)])
ax.set_xticklabels(['2020-10-%02d' % i for i in range(1,13)],rotation=45, fontsize=12)
plt.ylim([0,70])
plt.ylabel("pm2.5")
plt.title("折線圖")
plt.show()
```

