

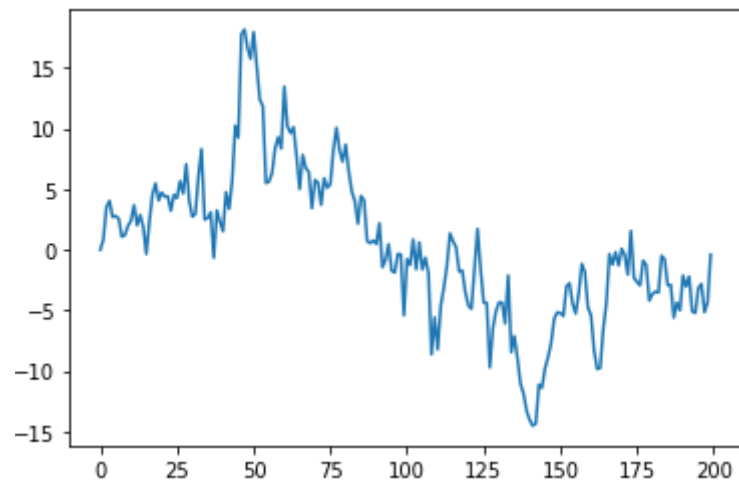
時間序列 HW7

0853411 劉書維

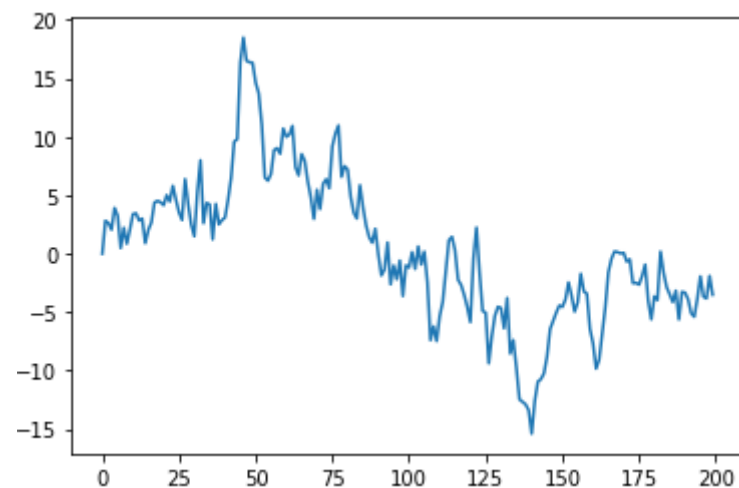
1. Generate a local-trend model :

我運用 Python 語言來模擬資料，利用 y 和 μ 的 list 來記錄資料。所以產

生出來的 y 如下圖：



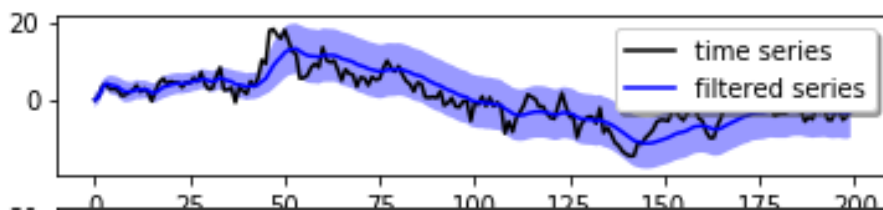
μ 如下圖：



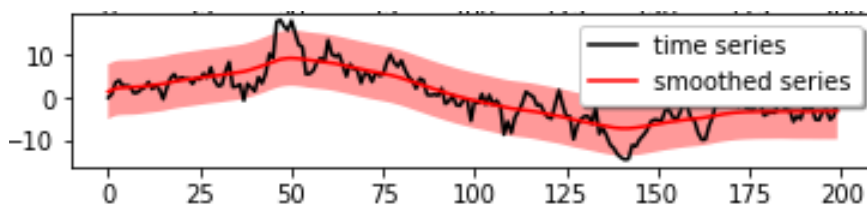
2. Perform Kalman filter and smoothing on it. Plot your results.

利用 Python 中的套件 `pydlm` 即可自動做出針對觀察到的 y 做出預測的

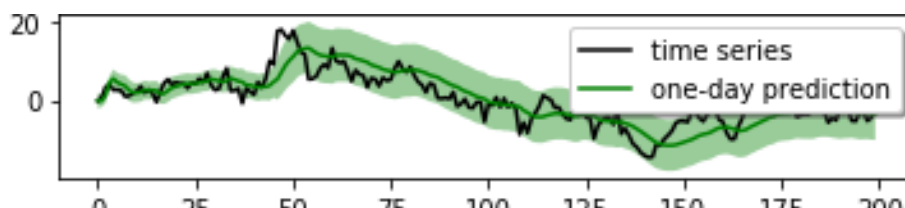
Kalman filter 和 smoothing，並畫出結果。所以 Kalman filter 如下圖：



Smoothing 結果如下：



另外也可以做提前一天的預測：



完整程式碼如下：

```
import numpy as np

# Generate a local-trend model
y = []
mu = []
start_point = 0

y.append(start_point)
mu.append(start_point)

for i in range(200):
    if i == 0:
        temp = start_point
```

```
else:
    data = temp + np.random.normal(0,1)
    y.append(data)
    next_temp = temp + np.random.normal(0, 2)
    mu.append(next_temp)
    temp = next_temp
```

```
#Perform Kalman filter and smoothing on it.
```

```
#Plot your results
```

```
from pydlm import dlm, trend, seasonality
```

```
linear_trend = trend(degree=1, discount=0.95, name='linear_trend', w=10)
```

```
simple_dlm = dlm(y)+ linear_trend
```

```
simple_dlm.fit()
```

```
simple_dlm.turnOff('data points')
```

```
simple_dlm.plot()
```