# 模組統計與機器學習初探

1.

$$\rho(X,Y) = \frac{cov(X,Y)}{\sqrt{var(X)\,var(Y)}}$$

Hint: $var(aX+bY) = a^2 var(X) + b^2 var(Y) + 2ab\, cov(X,Y)$

令 $b=1$

$$Var(aX+Y) = a^2 var(X) + var(Y) + 2a\, cov(X,Y)$$

$$\frac{d}{da} var(aX+Y) = \frac{d}{da}\left(a^2 var(X) + var(Y) + 2a\, cov(X,Y)\right)$$

$$\Rightarrow 2a\, var(X) + 2cov(X,Y) = 0$$

$$a = \frac{-cov(X,Y)}{var(X)}$$

代回 $a$

$$\left(-\frac{cov(X,Y)}{var(X)}\right)^2 var(X) + var(Y) + 2\left(\frac{-cov(X,Y)}{var(X)}\right)(cov(X,Y))$$

$$\Rightarrow \frac{cov^2(X,Y)}{var(X)} + var(Y) + \frac{-2cov^2(X,Y)}{var(X)}$$

$$\Rightarrow \frac{-cov^2(X,Y)}{var(X)} + var(Y) \geq 0$$

$$\Rightarrow var(X)\cdot var(Y) \geq cov^2(X,Y)$$

$$\Rightarrow \frac{cov^2(X,Y)}{var(X)\cdot var(Y)} \leq 1 \quad \#$$

2.

| $(x,y)$ | $x_i-\mu_X$ | $y_i-\mu_Y$ | | | |
|---|---|---|---|---|---|
| (1, 4) | -3.2 | -5.2 | 16.64 | 10.24 | 27.04 |
| (2, 5) | -2.2 | -4.2 | 9.24 | 4.84 | 17.64 |
| (5, 10) | 0.8 | 0.8 | 0.64 | 0.64 | 0.64 |
| (6, 12) | 1.8 | 2.8 | 5.04 | 3.24 | 7.84 |
| (7, 15) | 2.8 | 5.8 | 16.24 | 7.84 | 33.64 |

$\mu_X = 4.2$ $\mu_Y = 9.2$

$cov(X,Y) = 42.8$ $var(X) = 26.8$ $var(Y) = 86.8$

$$\rho = \frac{cov(X,Y)}{\sqrt{var(X)}\sqrt{var(Y)}} = \frac{42.8}{\sqrt{26.8}\sqrt{86.4}} = 0.9901 \quad \#$$

```
import numpy as np
data = [1,4,2,5,5,10,6,12,7,15]
data = np.array(data)
data = data.reshape(5,2)

##function
def pearsons_correlation_coeddicient_and_covariance(data):
    meanx = 0
    meany = 0
    varx = 0
    vary = 0
    covxy = 0
    for i in range(data.shape[0]):
        meanx += float(data[i][0])
        meany += float(data[i][1])
    meanx = meanx/data.shape[0]
    meany = meany/data.shape[0]
    for i in range(data.shape[0]):
        varx +=    (float(data[i][0])-meanx)**2
        vary +=    (float(data[i][1])-meany)**2
    for i in range(data.shape[0]):
        covxy += (float(data[i][0])-meanx)*(float(data[i][1])-meany)
    pearsons_correlations = covxy/((varx**0.5) * (vary**0.5))
    return meanx,meany,varx,vary,covxy,pearsons_correlations

##pearsons_correlations
meanx,meany,varx,vary,covxy,pearsons_correlations =
pearsons_correlation_coeddicient_and_covariance(data)
print(pearsons_correlations)
####output :0.0.9910615723046898


##target data
import numpy as np
np.random.seed(1)
n=100
noise = (np.random.rand(n)-0.5)*2
x = np.random.normal(0,1,n)
```

```
y = 5 * x    +    10 * noise

##data in list
datalist = []
for i in range(n):
        datalist.append(float(x[i]))
        datalist.append(float(y[i]))
datalist = np.array(datalist)
datalist = datalist.reshape(n,2)

##pearsons_correlations
meanx,meany,varx,vary,covxy,pearsons_correlations =
pearsons_correlation_coeddicient_and_covariance(datalist)
print(pearsons_correlations)
####output :0.5530828259342412
```