D13 pandas 統計函式使用教學

簡報閱讀 範例與作業 問題討論 學習心得(完成)

重要知識點
統計函式
統計函式：平均值mean()
統計函式：加總sum()，個數count()

重要知識點

- 應用統計函式
- 自定義的行或列函式應用

統計函式

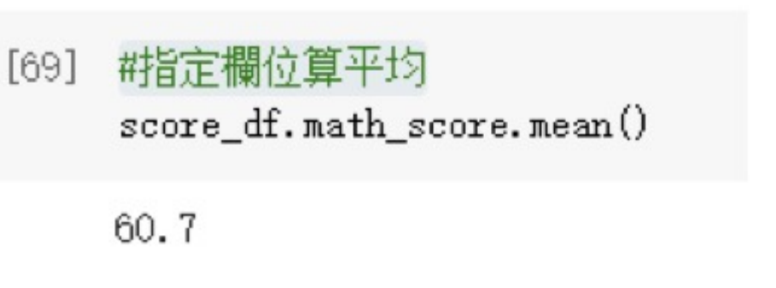
在生活中常聽到以下情況

1. 台灣平均薪資為 XXX
2. 今年指考最高分為 XXX
3. 今年台大最低入取分數為 XXX
4. 6 個標準差的良率

因為數據很多的情況下時常使用敘述統計量來描述數據的分佈與統計量，在資料分析中常拿來對資料做初步的了解。接下來我們以 pandas 的 DataFrame 資料來做統計函式的介紹。

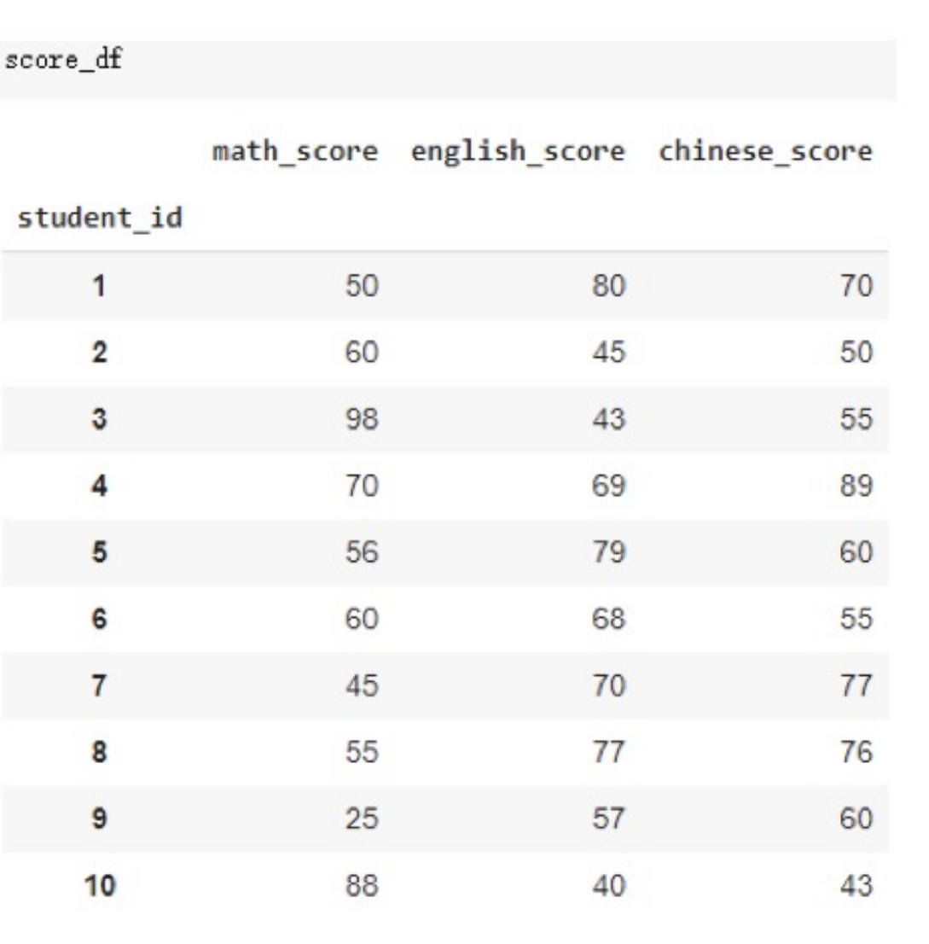
統計函式：平均值mean()

今天都以班上學生國文、英文、數學分數的資料(右表)為例子介紹各個統計函數。
首先是最常使用到的平均值 mean()，pandas 可針對指定欄位算平均值，如果沒指定會對全部欄位算平均值。



```
[69] #指定欄位算平均
     score_df.math_score.mean()

60.7
```

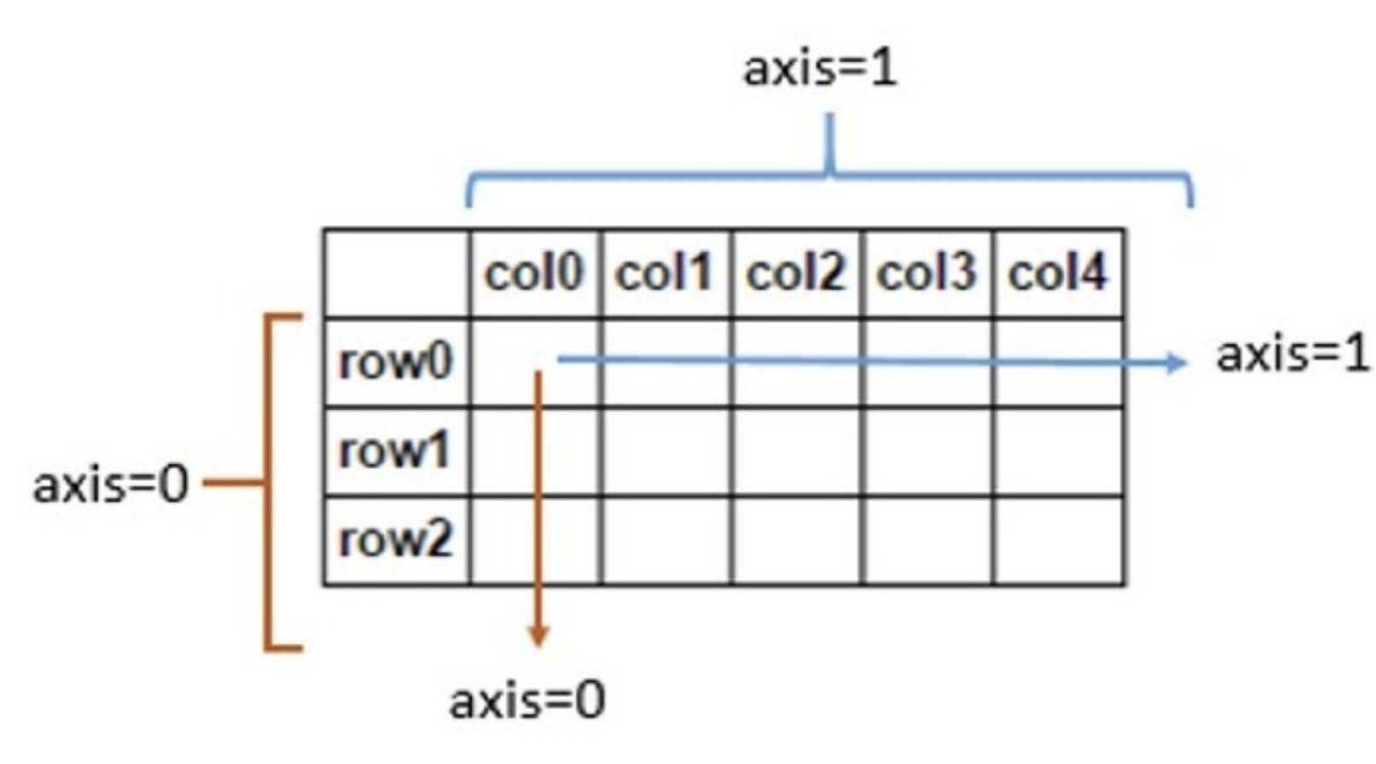
```
[70] #全欄位算平均
     score_df.mean()

math_score        60.7
english_score     62.8
chinese_score     63.5
dtype: float64
```

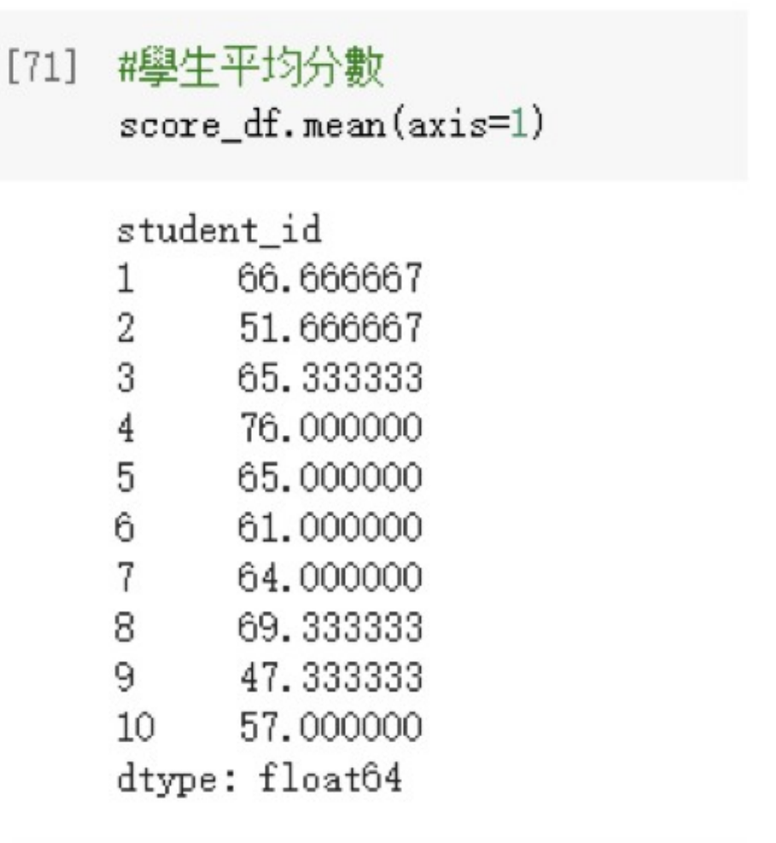


```
score_df
```

student_id	math_score	english_score	chinese_score
1	50	80	70
2	60	45	50
3	98	43	55
4	70	69	89
5	56	79	60
6	60	68	55
7	45	70	77
8	55	77	76
9	25	57	60
10	88	40	43

如果今天想要算每個學生的總平均分數怎麼辦？
Pandas 統計函式中有參數 axis=0 為行運算，axis=1 為列運算，此參數適用在之後介紹的統計函式。



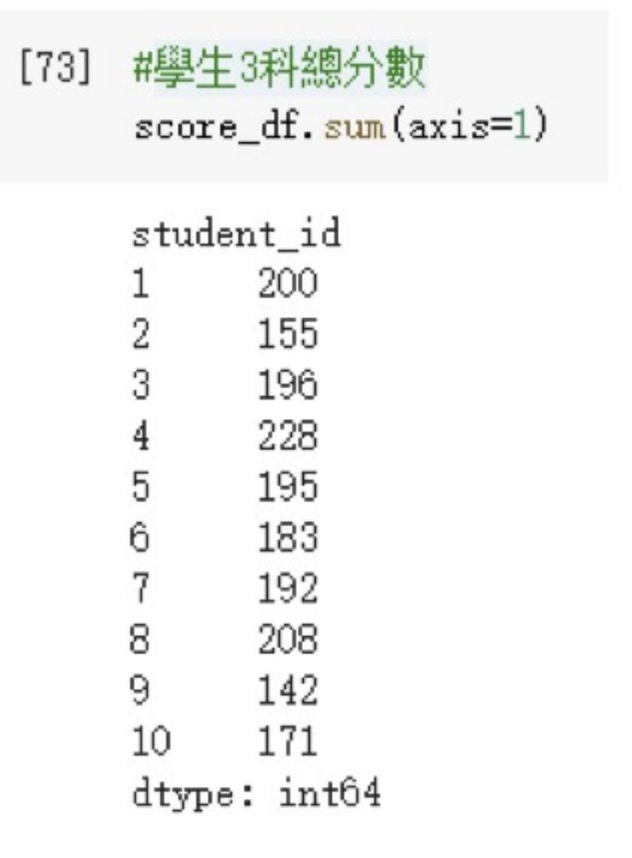


```
[71] #學生平均分數
     score_df.mean(axis=1)

student_id
1     66.666667
2     51.666667
3     65.333333
4     76.000000
5     65.000000
6     61.000000
7     64.000000
8     69.333333
9     47.333333
10    57.000000
dtype: float64
```

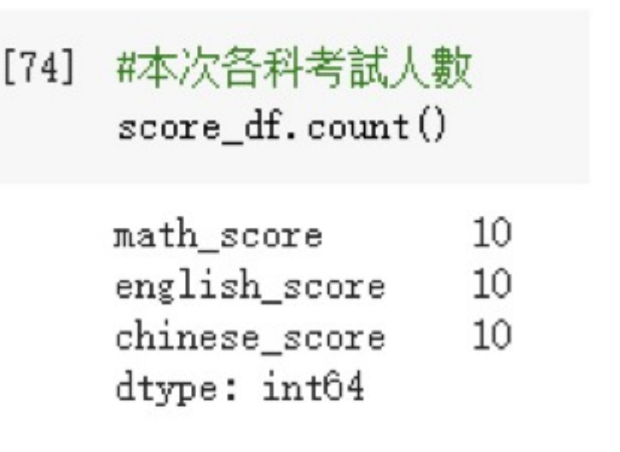
統計函式：加總sum()，個數count()

加總：計算總和，時常用在計算家庭開銷
個數：計算個數，時常用在出遊時的點名

以下利用加總算出學生 3 科總分，利用各數計算出應考人數



```
[73] #學生3科總分數
     score_df.sum(axis=1)

student_id
1     200
2     155
3     196
4     228
5     195
6     183
7     192
8     208
9     142
10    171
dtype: int64
```

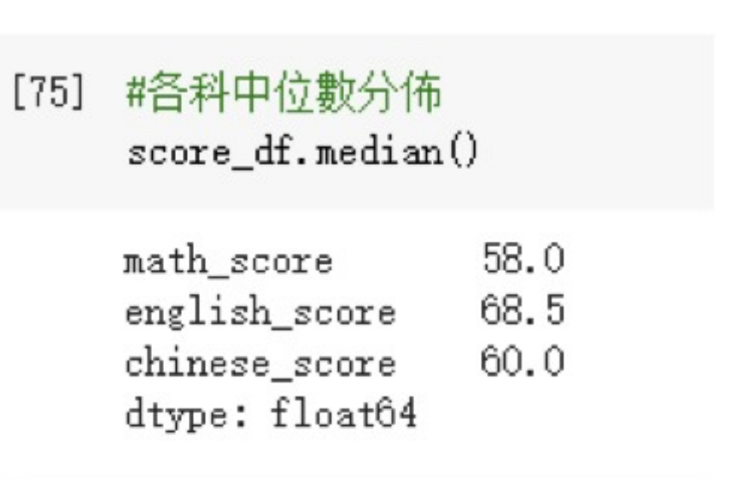


```
[74] #本次各科考試人數
     score_df.count()

math_score       10
english_score    10
chinese_score    10
dtype: int64
```

統計函式：中位數median()

中位數通常使用在有否贏過 50% 的數據，假如薪資中為數為 4 萬，超過 4 萬即為贏過 50% 的人，反之亦然。
中位數：通過把所有觀察值高低排序後找出正中間的一個作為中位數。 如果觀察值有偶數個，則中位數不唯一，通常取最中間的兩個數值的平均數作爲中位數。

以利用中位數算出各科中位數，如果今天數學考了 60 分超過了中位數的 58 分，我就可以說我數學贏過了全班一半的同學。

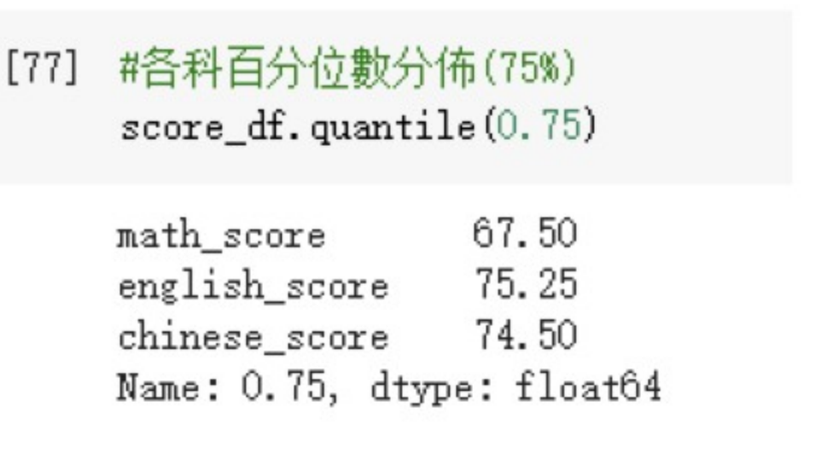


```
[75] #各科中位數分佈
     score_df.median()

math_score       58.0
english_score    68.5
chinese_score    60.0
dtype: float64
```

統計函式：百分位數quantile()

百分位數使用在觀察數據百分比，最常運用到的是升學分數的百分位數。
百分位數：將一組數據從小到大排序，並計算相應的累計百分位，則某一百分位所對應數據的值就稱為這一百分位的百分位數。如果百分位數設定在 50% 即為中位數。

以下計算 75% 的百分位數，如果我今天國文分數為 75分，我可以說我的國文贏過班上 75% 的同學



```
[77] #各科百分位數分佈(75%)
     score_df.quantile(0.75)

math_score       67.50
english_score    75.25
chinese_score    74.50
Name: 0.75, dtype: float64
```

統計函式：最大值max()、最小值min()

最大最小值時常拿觀察極端值，也可以檢視資料的資料最小與最大分佈。
其中最小值常常拿來當通過門檻，例如：大學入取分數最低幾分。

以下計算全班各科最高與最低分：

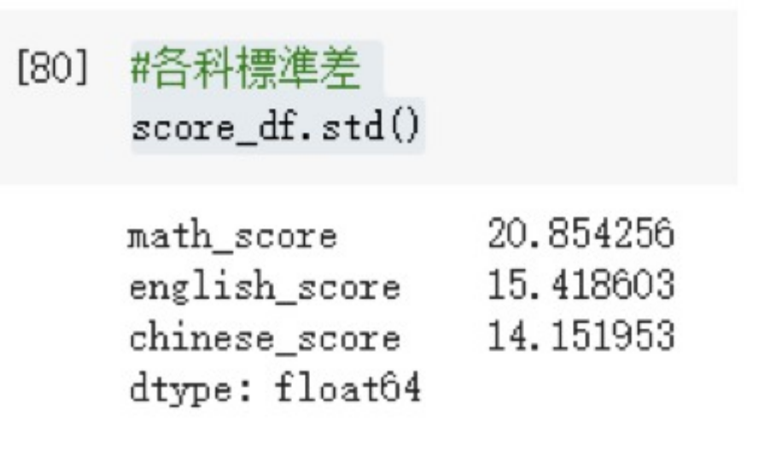
```
[78] #各科最大值
     score_df.max()

math_score       98
english_score    80
chinese_score    89
dtype: int64
```

```
[79] #各科最小值
     score_df.min()

math_score       25
english_score    40
chinese_score    43
dtype: int64
```

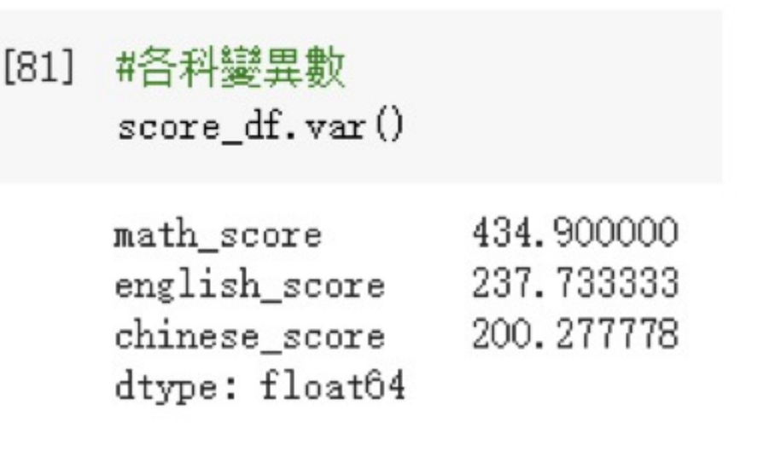
統計函式：標準差std()，變異數var()

標準差：在機率統計中最常使用作為測量一組數值的離散程度之用。一個較大的標準差，代表大部分的數值和其平均值之間差異較大；一個較小的標準差，代表這些數值較接近平均值。
變異數：為標準差平方

以下計算出標準差，可以發現國文分數標準差比數學分數標準差來的小，所以國文的分散程度比較小，也可以說國文分數較為集中。



```
[80] #各科標準差
     score_df.std()

math_score       20.854256
english_score    15.418603
chinese_score    14.151953
dtype: float64
```

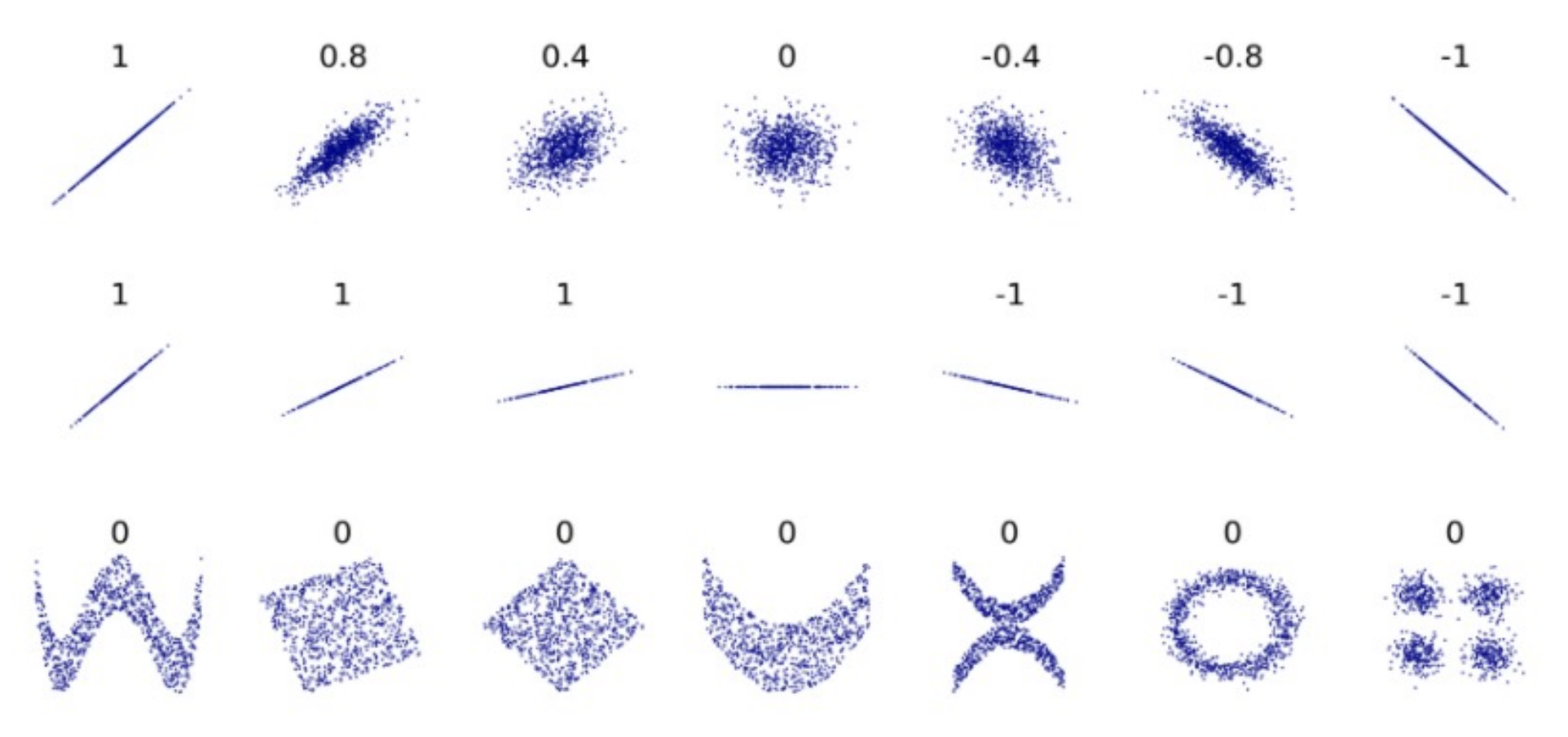


```
[81] #各科變異數
     score_df.var()

math_score       434.900000
english_score    237.733333
chinese_score    200.277778
dtype: float64
```

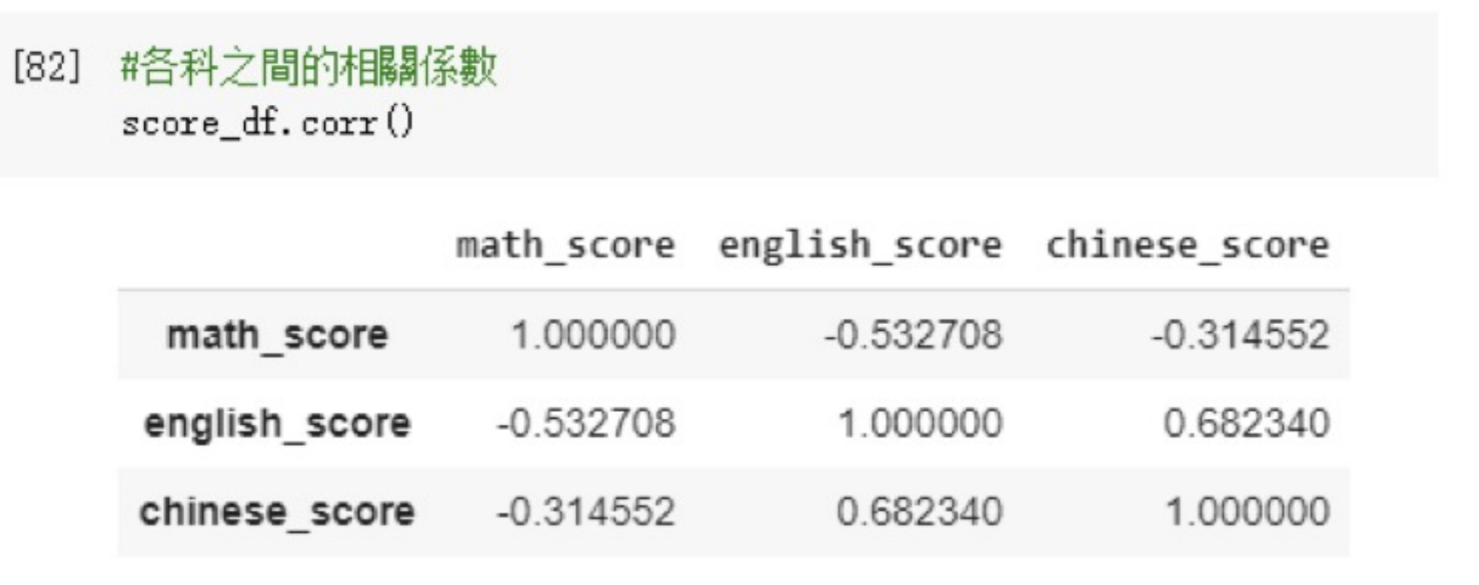
統計函式：相關係數corr()

相關係數：皮爾遜積矩相關係數（Pearson product-moment correlation coefficient）用於度量兩個變數X和Y之間的相關程度（線性相依）。在自然科學領域中，該係數廣泛用於度量兩個變數之間的線性相依程度。相關係數的值介於 −1 與 +1 之間，即 −1≤r≤+1。其性質如下：

1. 當 r>0 時，表示兩變數正相關，r<0 時，兩變數為負相關，r=0 時，表示兩變數間無線性相關關係。
2. 一般可按三級劃分：|r|<0.4 為低度線性相關；0.4≤|r|<0.7 為顯著性相關；0.7≤|r|<1為高度線性相關。



可以發現說英文相對數學相關係數為 -0.53，可以解釋說英文跟數學有負的高度線性相關，可以說明此班學生數學越高分英文越低分，另外國文相對英文相關係數為 0.68 為正向高度相關，說明此班學生英文越高分國文越高分。



```
[82] #各科之間的相關係數
     score_df.corr()
```

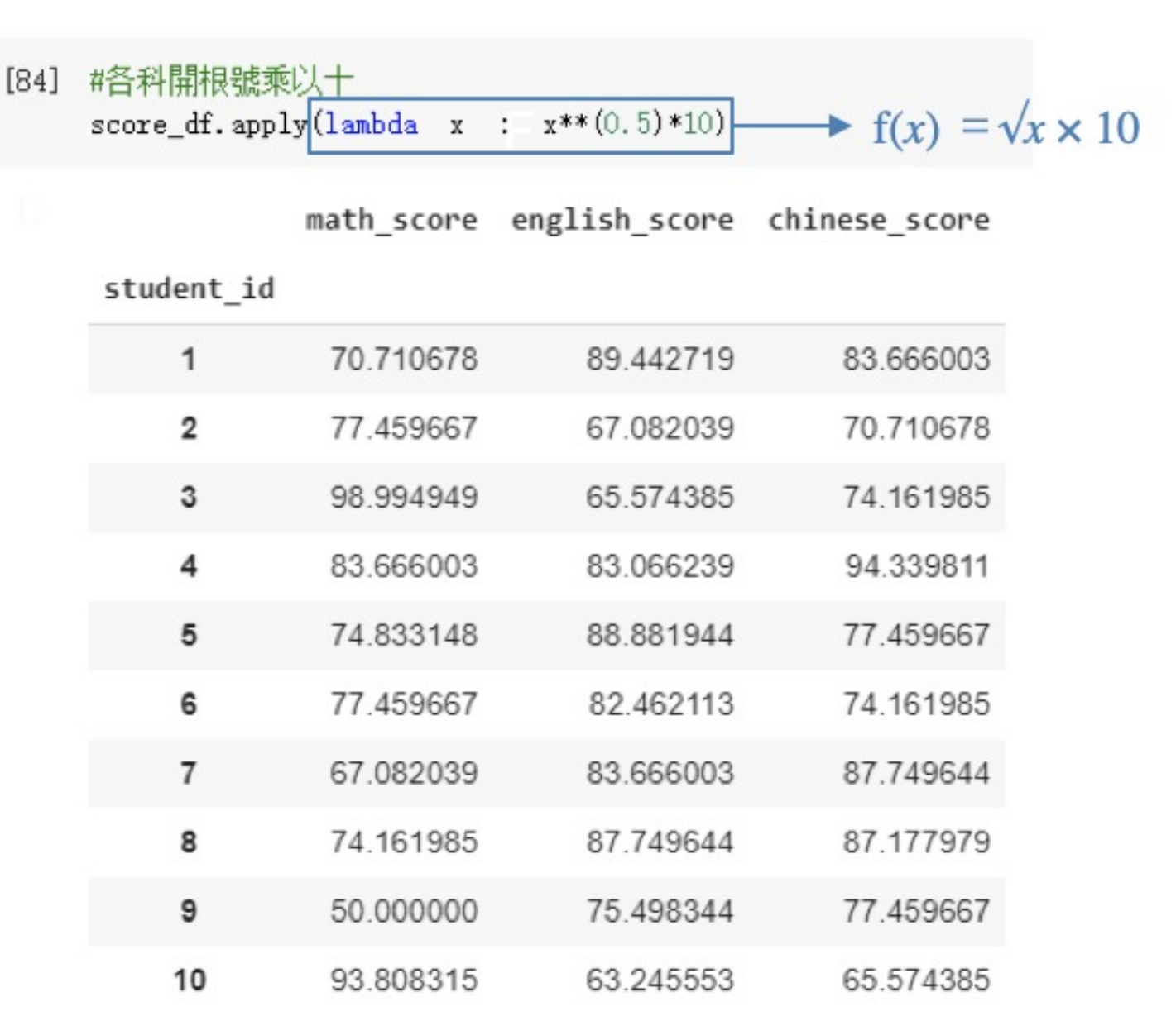
	math_score	english_score	chinese_score
math_score	1.000000	-0.532708	-0.314552
english_score	-0.532708	1.000000	0.682340
chinese_score	-0.314552	0.682340	1.000000

自訂義的行或列函式應用 apply()

你有時候可能儲覺得說前面的統計函式不足以表達資料的特性，此時你可以使用 apply 做自定義的函式。

像是學校最常使用的加分方式為開根號乘以十，例如：我考 49 分加分過後 √49 × 10 ＝ 70，這種方程式沒辦法在統計函式中算出來，需要藉由 apply 中 lambda 的函式達成。

其中 lambda x 相當於數學式中的 f(x) ＝√x × 10

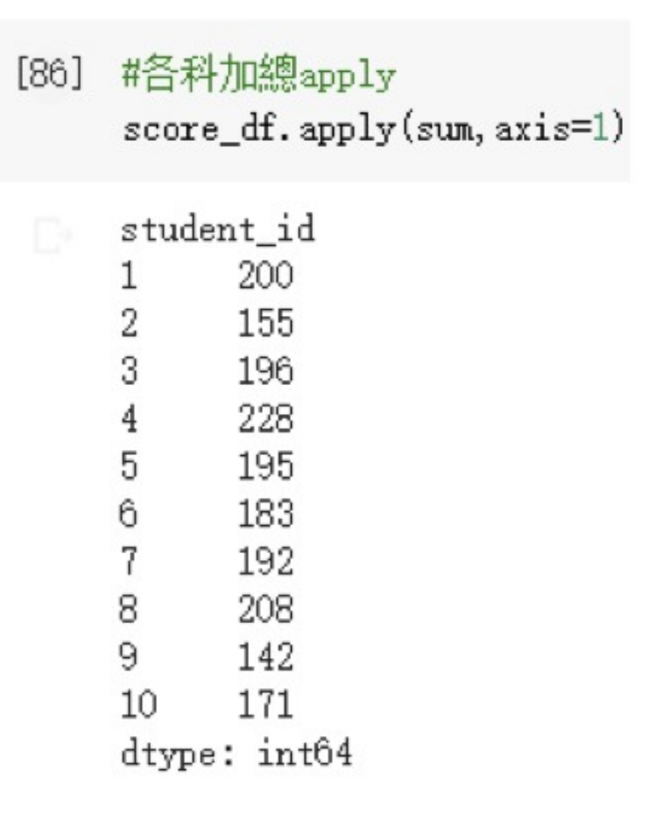


```
[84] #各科開根號乘以十
     score_df.apply(lambda  x  :  x**(0.5)*10)
```

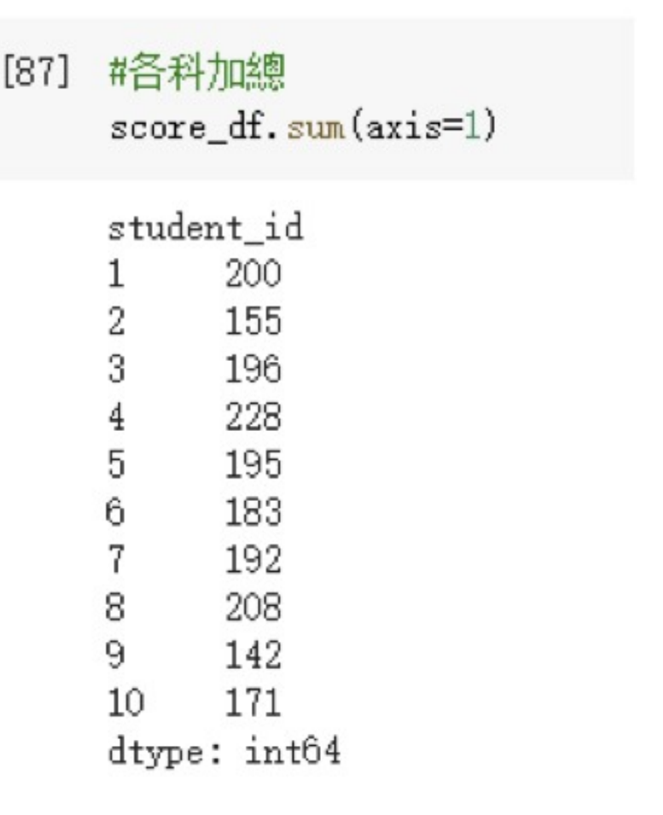
$f(x) = \sqrt{x} \times 10$

student_id	math_score	english_score	chinese_score
1	70.710678	89.442719	83.666003
2	77.459667	67.082039	70.710678
3	98.994949	65.574385	74.161985
4	83.666003	83.066239	94.339811
5	74.833148	88.881944	77.459667
6	77.459667	82.462113	74.161985
7	67.082039	83.666003	87.749644
8	74.161985	87.749644	87.177979
9	50.000000	75.498344	77.459667
10	93.808315	63.245553	65.574385

apply 也適用先前統計函式，可以用下列程式碼看出兩個計算邏輯是等價的。



```
[86] #各科加總apply
     score_df.apply(sum,axis=1)

student_id
1     200
2     155
3     196
4     228
5     195
6     183
7     192
8     208
9     142
10    171
dtype: int64
```



```
[87] #各科加總
     score_df.sum(axis=1)

student_id
1     200
2     155
3     196
4     228
5     195
6     183
7     192
8     208
9     142
10    171
dtype: int64
```

參考資料

Pandas 描述性統計
網站：程式教程網

有很多方法用來集體計算`DataFrame`的描述性統計信息和其他相關操作。 其中大多數是`sum()`，`mean()`等聚合函數，但其中一些，如`sumsum()`，產生一個相同大小的對象。 一般來說，這些方法採用軸參數，就像`ndarray.{sum，std，...}`，但軸可以通過名稱或整數來指定：

- *數據幀(DataFrame)* - 「index」(axis=0，默認)，columns(axis=1)

下面創建一個數據幀(DataFrame)，並使用此對象進行演示本章中所有操作。

示例

```
import pandas as pd
import numpy as np

#Create a Dictionary of series
d = {'Name':pd.Series(['Tom','James','Ricky','Vin','Steve','Minsu','Jack',
   'Lee','David','Gasper','Betina','Andres']),
   'Age':pd.Series([25,26,25,23,30,29,23,34,40,30,51,46]),
   'Rating':pd.Series([4.23,3.24,3.98,2.56,3.20,4.6,3.8,3.78,2.98,4.80,4.10,3

#Create a DataFrame
df = pd.DataFrame(d)
print df
```

Pandas 函數應用
網站：程式教程網

表格函數應用

可以通過將函數和適當數量的參數作爲管道參數來執行自定義操作。 因此，對整個`DataFrame`執行操作。

例如，爲`DataFrame`中的所有元素相加一個值`2`。

adder 函數

`adder`函數將兩個數值作爲參數相加並返回總和。

```
def adder(ele1,ele2):
    return ele1+ele2
```

現在將使用自定義函數對DataFrame進行操作。

```
df = pd.DataFrame(np.random.randn(5,3),columns=['col1','col2','col3'])
df.pipe(adder,2)
```

下一步：閱讀範例與完成作業