

商管程式設計（107-2）

作業五

作業設計：盧信銘
國立台灣大學資管系

截止時間：2019 年 4 月 15 日 21 點

測資助教：陳潔智

作業繳交請至 PDOGS (<http://pdogs.ntu.im/judge/>)。為各題上傳 Python 3.5 原始碼 (以複製貼上原始碼的方式上傳)。作業自己做。嚴禁抄襲。不接受紙本繳交，不接受遲交。請以英文或中文作答。

除了課本的內容外，Python 線上文件也很有用: <https://docs.python.org/3/>。好的程式設計師會把線上文件摸熟。

本次作業可以在各題使用的 library: math, sys, string。除上述可用之 library，禁用其他 library。違反禁用規定者，當題不予計分。如果你的結果正確，但沒有依照題目規定的方式實作，亦不予計分。

第一題

(30 points) 本題將練習一種時間數列的預測方法。時間數列是一串有先後次序的觀察值，大部分的狀況各觀察值是在固定的時間間格取值。很多時候時間數列有自我相關，也就是下一個觀察值會跟前幾個有關。這個特性可以用來建構預測時間數列的方法。簡單的說，我們對下一期的預測就是歷史資料的加權平均。而這個權重會隨則越遠離現在而遞減。

考慮一個時間數列 y_1, y_2, \dots, y_T ，其中下標代表時間（第幾期，由第一期開始）， T 代表現在的時間。我們對下一期的預測是：

$$\hat{y}_{T+1} = z_T = \alpha y_T + (1 - \alpha)z_{T-1}$$

其中 z_T 為最新的指數平滑結果， α 為指數平滑參數，需介於 $[0, 1]$ 之間。也就是說，我們不斷的利用 $z_t = \alpha y_t + (1 - \alpha)z_{t-1}$ 計算下一期的 z 值，然後用最新的結果當成預測值。在第一季的時候一律將 z_0 設為 0。

請寫一個名叫 ewp 的函數計算無母數迴歸的結果。你的程式應套用下面的範本：

```
def ewp(alpha, ylist):  
    #develop your function here.
```

```

#=====
alpha = input()
ystr = input()
yall = ystr.split(',')

out1 = ewp(alpha, yall)
if(out1 == None):
    print("ERROR")
else:
    print("%0.6f" % out1)

```

這個程式用 `input()` 分別獲得參數 `alpha` 與一個逗點分隔的時間數列資料，將資料依照逗點切割後，傳入 `ewp()`。`ewp()` 的第一個參數是處理好的時間數列資料。所有資料皆須為實數，數值大小需在正負十億之間。

在以下狀況時，`ewp()` 直接輸出 `None`。這時主程式會輸出 **ERROR**：

- 時間數列長度小於五。
- `alpha` 小於 0。
- `alpha` 大於 1。

如果沒有上述問題，則 `ewp()` 依照題目給的公式計算下一期的預測值，然後回傳。主程式會印出結果。輸出的部分利用了字串格式化的功能，會將結果輸出至小數六位，四捨五入，不足補零。例如，如果 `ans1 = 0.112`，則這個指令：`print("%0.6f" % ans1)` 會印出 `0.112000`。

範例輸入

```

0.6
1,2,3,4,5,6

```

範例輸出

```

5.336064

```

範例輸入

```

0.2
4,5,6,7

```

範例輸出

```

ERROR

```

範例輸入

```
0.9
100,110, 130, 150, 190, 210, 240, 150
```

範例輸出

```
158.675778
```

評分原則

依據程式運算的正確性給分。PDOGS 會直譯並執行你的程式、輸入測試資料，並檢查輸出答案的正確性。一筆測試資料佔 2 分。

第二題

(30 points) 隨機變數的生成是在許多情境下會遇到的問題。舉例而言，如果我們在設計一個猜拳遊戲，可能需要程式自動依照某個剪刀、石頭、布的機率分布，產生遊戲一方出示的選擇。要能做到這件事，我們需要可能結果的機率。比如在猜拳遊戲中，我們可以指定剪刀、石頭、布出現的機率為 0.2, 0.3, 0.5。依照這個機率，我們把 0 至 1 之間切成三段，分別是 $[0, 0.2]$, $(0.2, 0.5]$, 與 $(0.5, 1]$ 。這三個區段分別對應到剪刀、石頭、與布的選擇。當然，這個線段的長度剛好是它們被選到的機率。接下來如果我們可以由 0 至 1 的均勻分布 (Uniform Distribution) 中隨機抽出一個變數，並檢查這個變數是落在哪個區間內，然後傳回命中區間所對應到的選擇，這個流程就會讓回傳的結果服從我們要的機率分布。比如說如果我們由均勻分布抽出 0.4141234，則是落到 $(0.2, 0.5]$ 這個區間。因此對應到石頭。

依照前面所描述的方法，我們要設計一個名為 `gen_random()` 的函數，其輸入的參數名稱與其意義為：

- **problist**: 一個 list，代表各種可能發生的機率。比如有三種可能，則長度為 3，各元素為 0 至 1 之間的實數，且所有元素加起來為 1。如果輸入的數值不合乎要求，則函數傳回 `None`。所有元素加起來為 1 這條條件允許小量誤差。如果加起來的值離 1 小於 10^{-6} ，則當作是符合要求。如果輸入的機率符合誤差條件，則自動將最後一個類別的右邊界延伸至 1。
- **runif**: 一個介於 0 到 1 之間的數字。通常是由均勻分布 (Uniform Distribution) 中抽出。如果輸入的數值不合乎要求，則函數傳回 `None`。

`gen_random()` 會依照傳入值，決定輸出為何。為了簡化問題，我們規定輸出為整數，介於 0 至 $K-1$ 之間，其中 K 是 **problist** 的長度。

你的程式應套用下面的範本：

```
def gen_random(problist, runif):
    #develop your function here.

prob = input()
ru = float(input())

prob = prob.split(',')
for i in range(len(prob)):
    prob[i] = float(prob[i])

out1 = gen_random(prob, ru)
if(out1 == None):
    print("DATA_ERROR")
else:
    print(out1)
```

這個程式用 `input()` 由使用者獲得一個逗點分隔的機率資料，然後再由另一個 `input()` 獲得一個隨機變數。輸入轉成浮點數，傳入 `gen_random()`。

下面是本題的範例輸出入。

(a)

範例輸入

```
0.2,0.3,0.5
0.4141234
```

範例輸出

```
1
```

範例輸入

```
0.4,0.4
0.188
```

範例輸出

```
DATA_ERROR
```

範例輸入

```
0.2, 0.3, 0.3, 0.2
0.77777
```

範例輸出

```
2
```

評分原則：依據程式運算的正確性給分，一筆測試資料佔 2 分。

第三題

(40 points) 邏輯迴歸 (Logistic Regression) 是一個常用的資料分析手法。小花準備使用下面的模型分析 PBC 的期中考成績與最後是否能順利取得課程學分的關係：

$$p(Y = 1|X_1 = a_1, X_2 = a_2) = \frac{1}{1 + e^{-(c_0 + c_1 \cdot a_1 + c_2 \cdot a_2)}}$$

其中 Y 為 1 代表某個學生順利取得學分，而 Y 為 0 代表這個學生沒有取得學分。 X_1 與 X_2 為這個學生的第一次與第二次期中考成績。 c_0, c_1, c_2 為迴歸係數，皆為實數。迴歸係數可以將資料餵入統計估計的函數得到。我們在這個作業將探討給定 c_0, c_1, c_2 的值時，對取得學分這件事的意義。

第一小題

設計一個名為 `logic()` 的函數，這個函數將會依照輸入的值，計算某學生可以順利取得課程學分的機率。這個函數的輸入參數名稱與其意義為 (請依此順序)：

- `x`: 一個 list，長度為 2，為 a_1, a_2 的值。
- `coef`: 一個 list，長度為 3，為 c_0, c_1, c_2 的值。

你的程式應使用 `input()` 由使用者獲得一個逗點分隔的 a_1, a_2 ，並由另一個 `input()` 獲得逗點分隔的 c_0, c_1, c_2 。經過適當的處理之後，傳入 `logic()`，並印出輸出的機率值，輸出值應列印至小數六位，四捨五入，不足補零。

範例輸入

```
60, 70
-15, 0.1, 0.15
```

範例輸出

```
0.817574
```

範例輸入

```
70, 80
-18, 0.15, 0.1
```

範例輸出

0.622459

範例輸入

20, 60 -18, 0.15, 0.1

範例輸出

0.000123

第二小題

我們在探討這類的問題時，也關心邊際貢獻度的問題。也就是說，給定一組期中考成績，如果其中的一個分數有微小的變化，對取得學分機率的影響。針對這類的問題，我們可以透過數值方式計算一階微分的值。具體而言，給定一組期中成績 $[x_1, x_2]$ 以及係數 $[c_0, c_1, c_2]$ ，我們還需要指定一個計算數值微分的寬度 w 。針對第一次期中考的成績 x_1 ，我們可以計算在 $[x_1 + \frac{w}{2}, x_2]$ 與 $[x_1 - \frac{w}{2}, x_2]$ 的機率值，令這兩個機率值分別為 p_2 與 p_1 ，則在 $[x_1, x_2]$ 這個點， x_1 的邊際貢獻為： $\frac{p_2 - p_1}{w}$ 。類似的方法也可以用來計算在 $[x_1, x_2]$ 時， x_2 的邊際貢獻。本題中 x_1, x_2, c_0, c_1, c_2 均為實數。

設計一個名為 `logic_margin()` 的函數，這個函數將會依照輸入的值，計算兩個期中考成績對取得學分機率的邊際貢獻。這個函數的輸入參數名稱與其意義為 (請依此順序)：

- `x`: 一個 list，長度為 2，為 a_1, a_2 的值。
- `coef`: 一個 list，長度為 3，為 c_0, c_1, c_2 的值。
- `width`: 浮點數。

這個函數應輸出一個長度為 2 的 list，其內容為 x_1 與 x_2 的邊際貢獻。

你的程式應使用 `input()` 由使用者獲得一個逗點分隔的 a_1, a_2 ，由另一個 `input()` 獲得逗點分隔的 c_0, c_1, c_2 ，再由另一個 `input()` 獲得 w 。經過適當的處理之後，傳入 `logic_margin()`，並印出兩個邊際貢獻值，每個值一行。輸出值應列印至小數六位，四捨五入，不足補零。

範例輸入

```
65, 70
-16, 0.1, 0.15
1
```

範例輸出

```
0.019660
0.029487
```

範例輸入

```
45, 60
-16, 0.1, 0.15
1
```

範例輸出

```
0.007012
0.010521
```

範例輸入

```
65, 65
-16, 0.1, 0.15
0.1
```

範例輸出

```
0.024613
0.036920
```

評分原則

兩個小題各為 20 分。依據程式運算的正確性給分。PDOGS 會直譯並執行你的程式、輸入測試資料，並檢查輸出答案的正確性。一筆測試資料佔 2 分。