Informatique

3

Modules - Bibliothèques

TD 3-3
Calcul statistique

Exercice 1: Moyenne glissante – Courbes de tendance

Pour cet exercice, récupérer par copier-coller les deux listes « Liste_Temps » et « Liste_Poids » dans le fichier « 3-3 - TD - Calcul statistique - Données.py » en lien ici. Ces listes sont issues des mesures d'une balance connectée Withings.

Préliminaires

Question 1: Créer une fonction Affiche(fig,X,Y) qui affiche sur la figure numéro fig la courbe Y en fonction de X

Question 2: Tracer la courbe de poids en fonction du temps

Si L est une liste de flottants composée de N termes on rappelle les formules donnant sa moyenne *m* et sa variance *V* :

$$\mathrm{m} = \frac{1}{N}\sum_{i=0}^{N-1} L[i]$$

$$V = \frac{1}{N}\sum_{i=0}^{N-1} (L[i] - m)^2$$

Question 3: Ecrire une fonction Moyenne(L) qui, étant donnée une liste de nombres L, renvoie sa moyenne

```
>>> Moyenne([1, 3, 5, 6])
3.75
```

Question 4: Ecrire une fonction Variance(L) qui, étant donnée une liste de nombres L, renvoie sa variance

```
>>> Variance([1,3,5,6])
3.6875
```

Lissage

On cherche à lisser la courbe de poids obtenue ci-dessus.
Soit L une liste de nombres et n un entier supérieur ou égal à 1. Le principe du lissage consiste à remplacer chaque valeur de L par la moyenne de la valeur en question et de n valeurs de part et d'autre de celle-ci (i.e. n valeurs à droite et n valeurs à gauche). Toutefois, si une valeur se trouve sur un bord, on ne prendra que les valeurs qui sont aussi dans la liste. Par exemple, pour L= [3, 2, 3, 4, 4, 4, 3, 2, 3, 1] et n = 2, le premier 4 se remplacé par la moyenne de (2,3,4,4,4), le premier 3 sera remplacé par la moyenne de (3,2,3) et le dernier 3 sera remplacé par la moyenne de (3,2,3,1).

Question 5: Ecrire une fonction Extraction(L,i,n) qui, étant donnée une liste L, un indice i (compris entre 0 et len(L)-1) et un entier n supérieur ou égal à 1, renvoie la liste extraite de L correspondant aux valeurs dont il faudra faire la moyenne lors du lissage

Vérifier :

```
>>> Extraction([3,2,3,4,4,4,3,2,3,1],3,2)
[2, 3, 4, 4, 4]

>>> Extraction([3,2,3,4,4,4,3,2,3,1],0,2)
[3, 2, 3]

>>> Extraction([3,2,3,4,4,4,3,2,3,1],8,2)
[3, 2, 3, 1]
```

Question 6: Ecrire une fonction Lissage(L,n) qui, étant donnée une liste L et un entier n renvoie la liste lissée à partir de L avec n valeurs de part et d'autres de chaque éléments

Remarques :

- On veillera à ne pas modifier la liste L donnée en entrée
- On essaiera d'obtenir une complexité de cette fonction en O(1)

Question 7: Rajouter sur le graphique précédent la courbe de poids lissée avec par exemple n=10

Covariance

Si X et Y sont deux listes de flottants composées chacune de N termes, on appelle covariance de X et Y le nombre :

$$cov(X,Y) = \left(\frac{1}{N}\sum_{i=0}^{N-1} X[i]Y[i]\right) - m_x m_y$$

où m_x et m_y sont les moyennes respectives de X et Y.

Question 8: Ecrire une fonction Covariance(X,Y) qui, étant données deux listes X et Y contenant le même nombre d'éléments, renvoie la covariance de X et Y.

Vérifier :

```
>>> Covariance([1,2,3,5,9],[2,8,5,3,4])
-0.8000000000000007
```

Courbe de tendance

Nous allons maintenant déterminer une approximation de la droite qui passe au plus près par les dernières mesures effectuées afin d'en déduire le délai restant, en jours, pour atteindre un poids cible. Dans un premier temps, un peu de cours sur les approximations de droites aux moindres carrés. On souhaite déterminer l'équation de la droite qui passe au plus près, au sens des moindres carrés, d'un ensemble de points.

Soit la droite d'équation : $y = ax + b$

On souhaite minimiser la somme des carrés des distances entre N points connus - dont les abscisses et les ordonnées sont stockées dans deux listes X et Y - et leurs projetés sur la droite parallèlement à l'axe (Oy). Si on note S cette somme, on a :

$$S(a,b) = \sum_{i=0}^{N-1}\big(y_i - (ax_i + b)\big)^2 = \sum_{i=0}^{N-1}(y_i - ax_i - b)^2$$

Le minimum de S sera atteint lorsque les deux dérivées partielles de S sont nulles. Il faut donc résoudre le système ci-dessous :

$$\begin{cases} \dfrac{\partial S(a,b)}{\partial a} = 0 \\ \dfrac{\partial S(a,b)}{\partial b} = 0 \end{cases} \Leftrightarrow \begin{cases} -2\displaystyle\sum_{i=0}^{N-1} x_i(y_i - ax_i - b) = 0 \\ -2\displaystyle\sum_{i=0}^{N-1} (y_i - ax_i - b) = 0 \end{cases}$$

En notant m_x et m_y les valeurs moyennes des listes X et Y, les solutions a et b du système sont :

$$\begin{cases} a = \dfrac{\left[\frac{1}{N}\sum_{i=0}^{N-1}(x_i y_i)\right] - m_x m_y}{\left[\frac{1}{N}\sum_{i=0}^{N-1}(x_i{}^2)\right] - m_x{}^2} = \dfrac{cov(X,Y)}{V(X)} \\ b = m_y - a m_x \end{cases}$$

Remarque : on peut montrer que $\left[\frac{1}{N}\sum_{i=0}^{N-1}(x_i{}^2)\right] - m_x{}^2 = \left[\frac{1}{N}\sum_{i=0}^{N-1}(x_i - m_x)^2\right]$

Question 9: Ecrire une fonction Droite(LX,LY) qui renvoie le couple (a,b) des coefficients de la droite qui passe au mieux au sens des moindres carrés par les points dont les abscisses sont dans la liste LX et les ordonnées dans la liste LY

Pour la suite et pour simplifier légèrement le code, on suppose que la tendance permet de se rapprocher de l'objectif, ce qui est le cas des données fournies (perte de poids en cours avec un objectif de 60 kg en dessous du poids actuel).

Question 10: Ecrire une fonction Objectif(a, b, cible) qui, étant donnés trois nombres a, b et cible, renvoie l'abscisse du point d'intersection de la droite d'équation $y = ax + b$ avec la droite horizontale d'équation $y = cible$

Question 11: Déterminer à quel instant le poids de 60kg sera atteint si l'on prend en compte seulement les 10 dernières mesures

Question 12: Ecrire une fonction Affiche_Droite(fig,a,b,x1,x2) qui trace la droite d'équation $y = ax + b$ sur l'intervalle [x1,x2] sur la figure fig

Question 13: Rajouter sur la figure la droite qui permet de voir clairement la date où l'objectif de 60kg sera atteint tout en précisant dans la console le nombre de jours pour atteindre ce poids cible