Azahrieu Mohamed MP+I2

DS09 Info

Q1.

select idPATIENT from MEDICAL where etat="hernie discale";

Q2

select nom, prenom from PATIENT join Medical on PATIENT.id = MEDICAL.idPATIENT where MEDICAL.etat="spondylolisthesis";

Q3

select etat, count(idPATIENT) from MEDICAL group by etat;

Q4

avec Numpy ~~est~~ c'est plus rapide et plus competent que les listes python

Q5

N : nombre de ligne

n : nombre de colonne

on veut stocker un tableau de N ligne et

n colonne, ou chaque case contient 4 octets (32 bits)

+ un vecteur de taille N contenant des case

codé sur 1 octet (8 bits, des entiers)

donc

la memoire totale est

$\boxed{N \times n \times 4 + N = 2{,}5 \text{ Mo}}$

Q6

```
def separationParGroupe(data, etat):
    a = [ ]
    b = [ ]
    c [ ]
    for i in range(len(data)):
```

```
        if etat[i]==0:
            a.append(data[i])
        if etat[i]==1:
            b.append(data[i])
        if etat[i]==2:
            c.append(data[i])
    return a,b,c
```

Q7 Test = i != j

Q8

Q9 On pose $max(X_{norm}) = \frac{max(X)}{max(X)} = 1$

$min(X_{norm}) = 0 = min(X) - min(X)$

on peut ecrire aussi

$$max(X_{norm}) = \frac{max(X) - min(X)}{max(X) - min(X)}$$

et $min(X_{norm}) = \frac{min(X) - min(X)}{max(X) - min(X)}$

Ainsi on peut proposer $x_{norm,j} = \frac{x_j - min(X)}{max(X) - min(X)} \in$ qui est bien entre 0 et 1

Q10

```
def min_max(X):
    min = X[0]
    max = X[0]
    for i in X:
        if i < min:
            min = i
        if i > max:
            max = i
    return min, max
```

Q11

```
def distance(z, data)
```

Q12

Q13

Q14

Q15

```
def moyenne(X):
    S = 0
    for i in X:
        S = S + i
    return S/
```

```
def variance(X):
    m = moyenne(X)

    return m - m**2
```

dans ce cas

```
def variance (X):
    s=0
    m=moyenne (X)
    for i in X:
        s=S+i**2
    return S/len(X) - m**2
```