

Fontaine

Louis

## 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 etat = "spondylolisthesis" ✓

Q3) SELECT etat COUNT(idpatient) FROM MEDICAL  
GROUP BY etat  
Distinct

Q4) L'intérêt d'utiliser la bibliothèque de calcul  
numérique numpy quand les tableaux sont de  
grande taille est qu'elle est plus rapide. ✓

Q5) Pour data, chaque case contient 4 octets  
et il y a 100 000 lignes et 6 colonnes.  
et pour le tableau etat, il y a 100 000 lignes et  
1 colonne, et dans chaque case il y a 1 octet.  
D'où la quantité de mémoire nécessaire :

$$100.000 \times 6 \times 4 + 100.000 \times 1 \times 1 = 2,5 \text{ Mo}$$
 ✓



Q6)

```
def separationParGroupe (data, etat)  
    t0 = []  
    t1 = []  
    t2 = []  
    for i in range (len(data)) :  
        if etat[i] == 0  
            t0.append (data[i])  
        elif etat[i] == 1  
            t1.append (data[i])  
        elif etat[i] == 2  
            t2.append (data[i])  
    return t0, t1, t2
```

Q7)

TEST =  $i \neq j$  car on cherche à savoir si on est ou non sur la diagonale

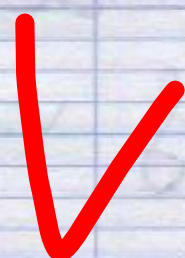
V

Q8) Le diagramme de la diagonale permet de visualiser comment sont reliées les données et vérifier si quelqu'un possède un attribut donné

V



$$Q9) x_{\text{norm}} = \frac{x_j - \min(X)}{\max(X) - \min(X)}$$



Q10)

def min-max (X)

min, max = X[0], X[0]

for i in X:

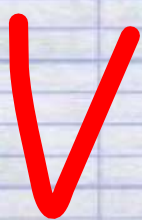
if i < min:

min = i

if i > max:

max = i

return min, max



Et la complexité ?

Q11)

def distance(z, data)

L = []

for i in range(len(data)):

D = 0

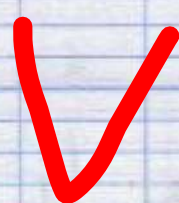
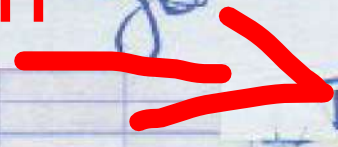
for j in range(len(data[i])):

D += (z[i] - data[i][j]) \*\* 2

L.append(sqrt(D))

return L

Eviter ce nom  
de variable





Q15/

def moyenne(x):

sc = 0

for i in x:

sc = sc + i

return sc / len(x)

def variance(x)

sc = 0

for j in x:

sc = sc + j \*\* 2

return sc / len(x) - moyenne(x) \*\* 2