

- 1/ 

```
SELECT idpatient  
FROM MEDICAL  
WHERE etat = "hernie discale" ;
```
- 2/ 

```
SELECT nom, prenom  
FROM PATIENT, MEDICAL  
WHERE etat = "spondyloarthrose" ;
```
- 3/ 

```
SELECT etat FROM MEDICAL WHERE etat = "normal"  
UNION UNION  
SELECT COUNT(*) FROM MEDICAL WHERE etat = "normal"  
UNION  
SELECT COUNT(*) FROM MEDICAL WHERE etat = "hernie discale"  
UNION  
SELECT COUNT(*) FROM MEDICAL WHERE etat = "spondyloarthrose" ;
```
- 4/ La bibliothèque Numpy permet de manipuler et de calculer des tableaux plus rapidement.  
Donc, quand les tableaux sont de grandes tailles, le temps gagné n'est pas négligeable.
- 5/ Le tableau data est codé sur 32 bits, avec  $N$  colonnes.  
Le vecteur etat est ~~codé sur~~ de taille  $N$  avec des ~~bits~~ entiers codés sur 8 bits.  
Donc la quantité de mémoire totale nécessaire est de 4 Mo.



6/ def separation Par Groupe (data, etat) :

data 0 = []

data 1 = []

data 2 = []

for i in range (len(data)) :

if data[i] = etat [0] :

data 0 . append (data[i])

if data[i] = etat [1] :

data 1 . append (data[i])

if data[i] = etat [2] :

data 2 . append (data[i])

return data 0, data 1, data 2

7/ ARGS1 : (6, 6, k)

ARGS2 : ( , , ~~mark[k]~~ marker = mark[k])

ARGS3 :

TEST :

8/ Les diagrammes présents sur la diagonale permettent de visualiser la population de patient en fonction de l'attribut désiré.

Les diagrammes hors diagonale permettent de distinguer si l'état du patient a une incidence sur l'attribut cherché ou non.

9/  $X_{normj} =$



10/ def min\_max(X):

return min(X), max(X)

11/ def distance(z, data):

D = []

for i in range(len(data)):

D.append(data[i] - z[i])

return D

12/ La partie 1 de l'algorithme va créer et trier la liste T, avec les éléments décrits dans le sujet

la partie 2

Et enfin la partie 3

13/

14/ En constatant que le taux de réussite de l'algorithme est maximal aux valeurs de  $k = 10$

Avant et après la courbe décroît

L'efficacité de cet algorithme est bonne (entre 70 et 75%) mais pas encore assez bonne pour être significative.