Mini Projet – Un nouvel apprentissage pour du clustering

Principe : distance de Hamming

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Exemple** | **Couleur** | **Noyaux** | **Flagelles** | **Membrane** |
| 1 | claire | 2 | 2 | fine |
| 2 | claire | 1 | 2 | fine |
| 3 | foncée | 2 | 2 | fine |
| 4 | claire | 2 | 1 | fine |
| 5 | claire | 2 | 2 | épaisse |
| 6 | claire | 1 | 1 | épaisse |
| 7 | foncée | 2 | 2 | épaisse |
| 8 | foncée | 1 | 1 | fine |
| 9 | foncée | 1 | 1 | épaisse |
| 10 | foncée | 2 | 1 | épaisse |

Questions :

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| 1 |  | 1 | 1 | 1 | 1 | 3 | 2 | 3 | 4 | 3 |
| 2 | 1 |  | 2 | 2 | 2 | 2 | 3 | 2 | 3 | 4 |
| 3 | 1 | 2 |  | 2 | 2 | 4 | 1 | 2 | 3 | 2 |
| 4 | 1 | 2 | 2 |  | 2 | 2 | 3 | 2 | 3 | 2 |
| 5 | 1 | 2 | 2 | 2 |  | 2 | 1 | 4 | 3 | 2 |
| 6 | 3 | 2 | 4 | 2 | 2 |  | 3 | 2 | 1 | 2 |
| 7 | 2 | 3 | 1 | 3 | 1 | 3 |  | 3 | 2 | 1 |
| 8 | 3 | 2 | 2 | 2 | 4 | 2 | 3 |  | 1 | 2 |
| 9 | 4 | 3 | 3 | 3 | 3 | 1 | 2 | 1 |  | 1 |
| 10 | 3 | 4 | 2 | 2 | 2 | 2 | 1 | 2 | 1 |  |

1. Une fois qu’on a calculé les distances de Hamming pour tous les éléments du tableau d’exemples, on prend les 2 exemples avec la plus grande distance de Hamming.

Ici la plus grande distance de Hamming est 4 :

* Entre le 1 et le 9
* Entre le 2 et le 10
* Entre le 3 et le 6
* Entre le 5 et le 8

On prend donc un de ces groupes pour commencer (ici nous prendrons le 1 et le 9 car ce sont les premiers à venir dans l’ordre du tableau).

Nous commençons donc à remplir nos 2 clusters avec l’élément 1 dans le premier et l’élément 9 dans le deuxième.

|  |  |
| --- | --- |
| **Cluster A** | **Cluster B** |
| 1 | 9 |

Pour le code : Nous remplissons un troisième cluster « temporaire » pour stocker les autres éléments en attendant de les mettre dans nos clusters A et B.

Ensuite on a le choix entre commencer par les éléments qui ont une distance de Hamming minimale (ici 1) ou maximale (ici 4).

# Classement par distance de Hamming minimal.

Ici on va placer les éléments qui ont une distance de Hamming minimal dans le même cluster.

On prend tous les éléments qui ont un distance de Hamming minimale (1 pour commencer) avec le premier élément du cluster A (donc avec 1). On obtient donc les éléments 2, 3, 4 et 5.

|  |  |
| --- | --- |
| **Cluster A** | **Cluster B** |
| 1 | 9 |
| 2 |  |
| 3 |  |
| 4 |  |
| 5 |  |

Puis, on fait la même chose avec les éléments qui ont une distance de Hamming minimale (toujours 1) avec le premier élément du cluster B (donc avec 9). On obtient donc les éléments 6, 8 et 10.

|  |  |
| --- | --- |
| **Cluster A** | **Cluster B** |
| 1 | 9 |
| 2 | 6 |
| 3 | 8 |
| 4 | 10 |
| 5 |  |

Ensuite, on continue avec le deuxième élément du cluster A (donc avec 2). On n’obtient rien donc on ne rajoute rien à nos clusters.

Puis, on continue avec le deuxième élément du cluster B (donc avec 6). On n’obtient rien donc on ne rajoute rien à nos clusters.

Enfin, on continue avec le troisième élément du cluster A (donc avec 3). On obtient l’élément 7.

|  |  |
| --- | --- |
| **Cluster A** | **Cluster B** |
| 1 | 9 |
| 2 | 6 |
| 3 | 8 |
| 4 | 10 |
| 5 |  |
| 7 |  |

Evidemment, si après avoir comparé tous les éléments de chaque cluster (A et B), il reste des éléments à placer alors on recommence les étapes en passant la distance de Hamming à 2.

# Classement par distance de Hamming maximale.

Ici on va placer les éléments qui ont une distance de Hamming maximale dans des clusters différents.

On prend tous les éléments qui ont un distance de Hamming maximale (4 pour commencer) avec le premier élément du cluster A (donc avec 1). On n’obtient rien donc on ne rajoute rien à nos clusters.

Puis, on fait la même chose avec les éléments qui ont une distance de Hamming maximale (toujours 4) avec le premier élément du cluster B (donc avec 9). On n’obtient rien donc on ne rajoute rien à nos clusters.

Ensuite, on recommence alors les étapes en passant la distance de Hamming à 3.

On prend tous les éléments qui ont un distance de Hamming de 3 avec le premier élément du cluster A (donc avec 1). On obtient donc les éléments 6, 8 et 10.

|  |  |
| --- | --- |
| **Cluster A** | **Cluster B** |
| 1 | 9 |
|  | 6 |
|  | 8 |
|  | 10 |

Puis, on prend tous les éléments qui ont un distance de Hamming de 3 avec le premier élément du cluster B (donc avec 9). On obtient donc les éléments 2, 3, 4 et 5.

|  |  |
| --- | --- |
| **Cluster A** | **Cluster B** |
| 1 | 9 |
| 2 | 6 |
| 3 | 8 |
| 4 | 10 |
| 5 |  |

Enfin, on continue avec le deuxième élément du cluster A (donc avec 2). On obtient l’élément 7.

|  |  |
| --- | --- |
| **Cluster A** | **Cluster B** |
| 1 | 9 |
| 2 | 6 |
| 3 | 8 |
| 4 | 10 |
| 5 | 7 |

# Conclusion

On remarque donc que les clusters se remplissent quasiment pareil suivant si on n’utilise le classement par distance de Hamming minimale ou le classement par distance de Hamming maximale. Le premier classement consiste à regrouper les éléments qui ont une faible distance de Hamming entre eux. Le deuxième classement quant à lui consiste à séparer les éléments qui ont une forte distance de Hamming entre eux.

On remarque également que dans le cas présent, l’élément 7 peut aussi bien se trouver dans le cluster A que dans le cluster B, cela dépend de si on commence par trier les éléments en commençant par le cluster A ou par le cluster B. Donc l’élément 7 sera placé dans le premier cluster qui le récupère.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Cluster A** | **Cluster B** |  |  |  | **Cluster A** | **Cluster B** |
| 1 | 9 |  |  |  | 1 | 9 |
| 2 | 6 |  | OU |  | 2 | 6 |
| 3 | 8 |  |  |  | 3 | 8 |
| 4 | 10 |  |  |  | 4 | 10 |
| 5 |  |  |  |  | 5 | 7 |
| 7 |  |  |  |  |  |  |