|  |  |
| --- | --- |
| |  | | --- | |  | |

**Devoir #3 : Les réseaux SOM de Kohonen**

Session : Eté 2015

*Réseaux de neurones*

*Département d’informatique et de mathématiques*

*Présenté à : Professeur: H.Ezzaidi*

*Travail de : AMAMOU Houssem*

Code permanent : AMAH10029004

# Une carte linéaire

1. Les données ont été classifiées de la façon suivante.

|  |  |
| --- | --- |
| Donnée | Classe |
| 1 | 1 |
| 2 | 1 |
| 3 | 1 |
| 4 | 1 |
| 5 | 2 |
| 6 | 2 |
| 7 | 2 |
| 8 | 2 |
| 9 | 2 |
| 10 | 3 |
| 11 | 3 |
| 12 | 3 |
| 13 | 3 |
| 14 | 3 |
| 15 | 3 |
| 16 | 3 |

Ci-dessous le résultat trouvé dans la matrice p. La première ligne correspond aux coordonnées x et y du premier centre de gravité et ainsi de suite.

|  |  |
| --- | --- |
| X | Y |
| -2.4807 | 1.5090 |
| 4.0159 | 3.5047 |
| 3.0447 | -3.0846 |

Tableau représentant la matrice p

Le tableau ci-dessous représente les moyennes calculées des centres de gravité

|  |  |
| --- | --- |
| X | Y |
| -2.5 | 1.5 |
| 4 | 3.5 |
| 3 | -3.0714 |

On remarque que la différence entre les deux tableaux est très minime et que les résultats sont très proches donc le réseau a bien représenté les centres de gravité.

# La réduction des dimensions

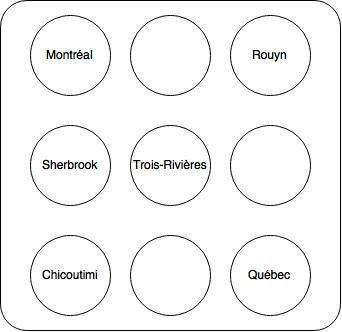
1. On remarque que les données ont été classées dans 5 différentes classes. On remarque que le réseau a projeté les données dans un espace à 1 dimension.

|  |  |
| --- | --- |
| Donnée | Classe |
| 1 | 1 |
| 2 | 2 |
| 3 | 3 |
| 4 | 4 |
| 5 | 5 |
| 6 | 5 |

Le tableau ci-dessous représente les coordonnées de chaque classe. La coordonnée est une abscisse sur l’axe X.

|  |  |
| --- | --- |
| Classe | Coordonnées |
| 1 | 6 |
| 2 | 4 |
| 3 | 3 |
| 4 | 2 |
| 5 | 1 |

# La préservation topologique



Résultat de la carte de Kohonen pour la représentation des villes

En observant la figure et la carte géographique, on remarque qu’il y a une certaine affinité dans la disposition des villes. Les villes les plus éloignées ont été mis de la même façon sur la carte comme exemple : Rouyn et Chicoutimi. On voit bien que ce n’est pas possible de représenter les distances avec une échelle réaliste mais il y a une certaine logique dans la disposition des villes par exemple mettre Trois-Rivières en Montréal et Québec.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Chicoutimi | Montréal | Québec | Sherbrooke | Trois-Rivières | Rouyn |
| Chicoutimi | 0 | 783.5222 | 478.7713 | 744.4555 | 686.8253 | 1454.311 |
| Montréal | 783.5222 | 0 | 523.9303 | 255.0588 | 281.7481 | 1448.262 |
| Québec | 478.7713 | 523.9303 | 0 | 445.2101 | 317.7184 | 1656.834 |
| Sherbrooke | 744.4555 | 255.0588 | 445.2101 | 0 | 263.4407 | 1538.672 |
| Trois-Rivières | 686.8253 | 281.7481 | 317.7184 | 263.4407 | 0 | 1589.876 |
| Rouyn | 1454.311 | 1448.262 | 1656.834 | 1538.672 | 1589.876 | 0 |

Tableau des distances euclidiennes entre les villes

On remarque que les distances sont bien respectées et reflète bien ces valeurs sur la carte de Kohonen. C’est-à-dire que les villes plus proches ont été mise à coté sur la carte de Kohonen et les villes les plus éloignées ont été éloignées de même sur la carte. La distance est représentée sous forme de voisinage de chaque ville.

# Le principe de voisinage

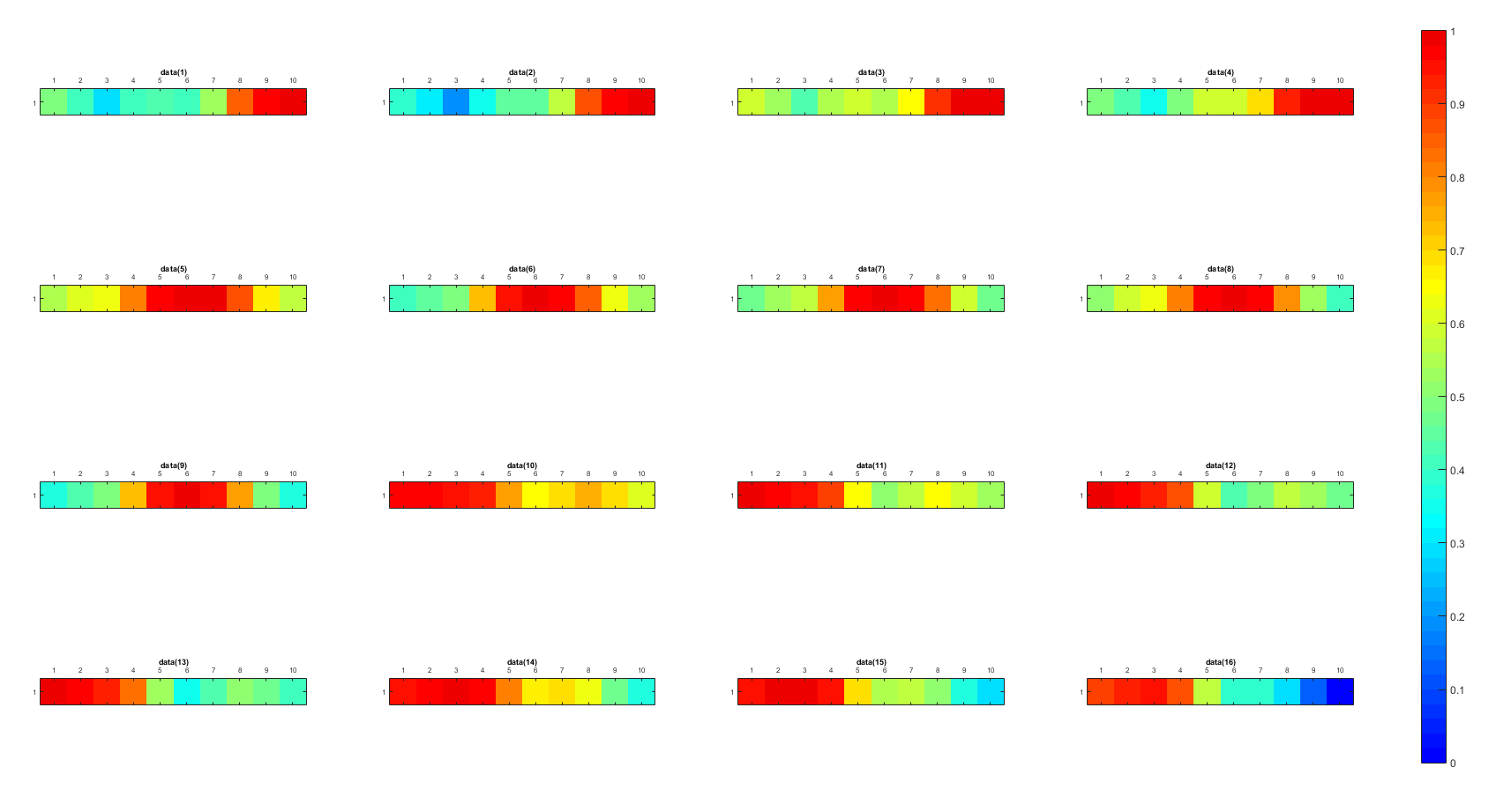
1. Les données ont été classées en 5 classes selon la variable c. En ce qui concerne le premier vecteur d’entrée la réponse du réseau est la suivante :

|  |  |
| --- | --- |
| C:\Users\Laimi\Desktop\rmap1.emf  Réponse du réseau pour le premier vecteur | C:\Users\Laimi\Desktop\rmap5.emf  Réponse du réseau pour le cinquième vecteur |
| C:\Users\Laimi\Desktop\rmap12.emf  Réponse du réseau pour le douzième vecteur | C:\Users\Laimi\Desktop\rmap4.emf  Réponse du réseau pour le quatorzième vecteur |

On remarque que le neurone gagnant est celui qui a la réponse la plus basse. En consultant la variable et c et la variable pMap qui indique l’endroit où est projeté cette classe sur la carte de Kohonen. Le neurone gagnant est le neurone numéro dix comme indiqué sur la figure.

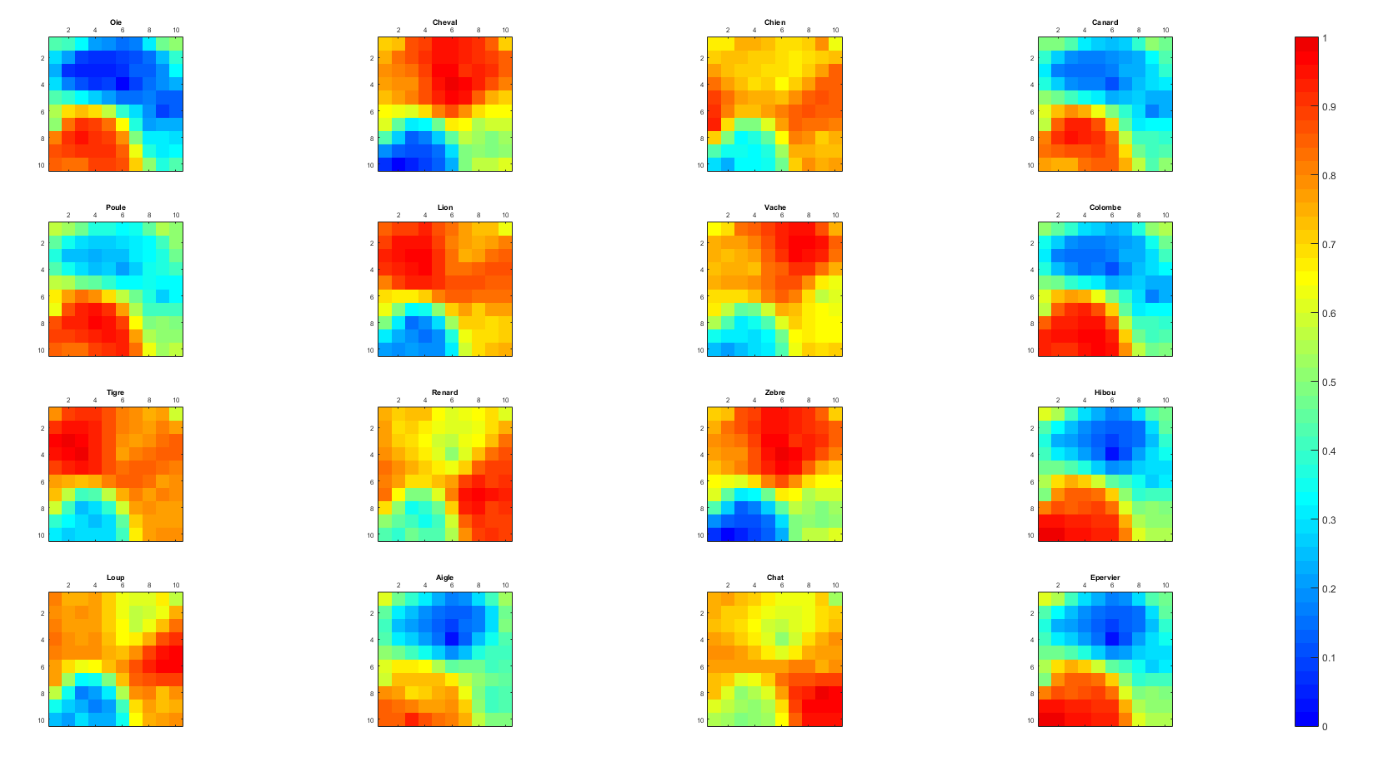
Le cinquième vecteur a été classé dans le neurone numéro sept comme le montre la figure et le confirme la matrice c et pMap. Le douzième vecteur a été projeté sur le deuxième neurone et le quatorzième vecteur sur le neurone quatre.

Le neurone gagnant obtient la réponse la plus basse car cet algorithme, en choisissant le neurone gagnant par compétition entre tous les neurones et en utilisant la fonction d’apprentissage sélectionnée, la réponse du réseau permet de distinguer le neurone gagnant et le voisinage de ce neurone qui va être projeté



Réponse du réseau pour tous les vecteurs d’entrée

# Le rayon d’apprentissage

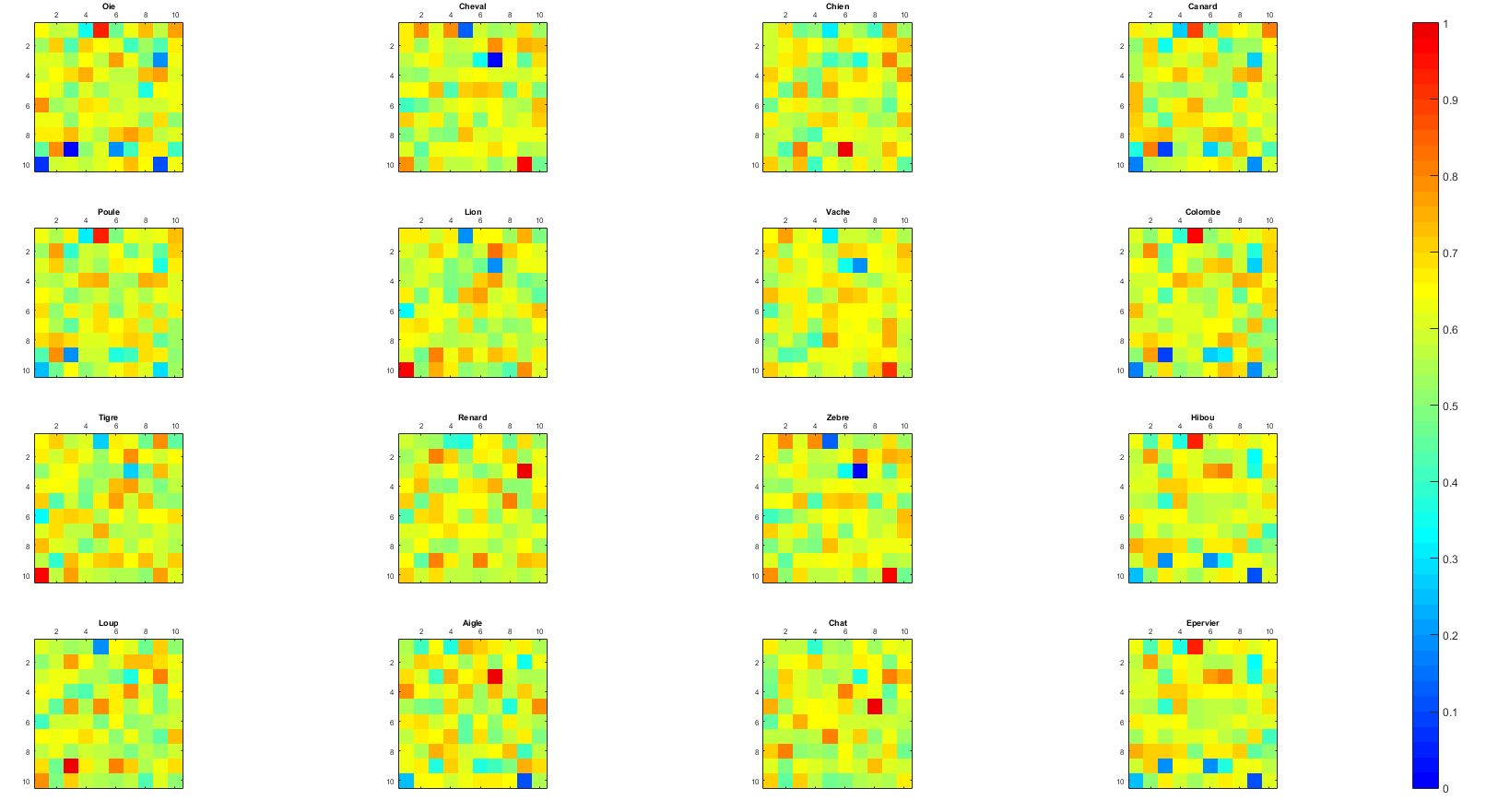


Résultat de la classification des données par les réseaux SOM de Kohonen

* En observant la carte de Kohonen, on remarque que les oiseaux ont été classés au coin gauche en bas de la figure. Autrement dit
* Non, dans cette région, il n’y a pas d’autres animaux que les oiseaux qui sont présents dans cette région.
* Oui, il y a des sous-groupes au sein des oiseaux. On retrouve les oiseaux de proie plus à gauche dans la carte par exemple l’aigle se trouve dans le coin gauche bas alors que le canard et l’épervier sont un peu plus à droite que l’aigle.
* La répartition des classes n’est pas aléatoire. Comme on peut le voir sur la figure le réseau a divisé les données en deux grandes classes : les herbivores et les carnivores. Les mammifères ont été divisé en deux classes les oiseaux et les herbivores tels que le zèbre ou la vache. De même qu’on retrouve un classement similaire chez les carnivores. D’un côté, on retrouve les canidés tels que le chien ou le renard et d’un autre coté les félins par exemple le lion et le tigre. Le réseau rassemble les animaux qui ont des similitudes au niveau physique et comportemental. C’est pour cette raison qu’on retrouve les animaux à peu près similaires appartenant à la même famille proche dans la carte.



Zones des différentes familles d’animaux classées

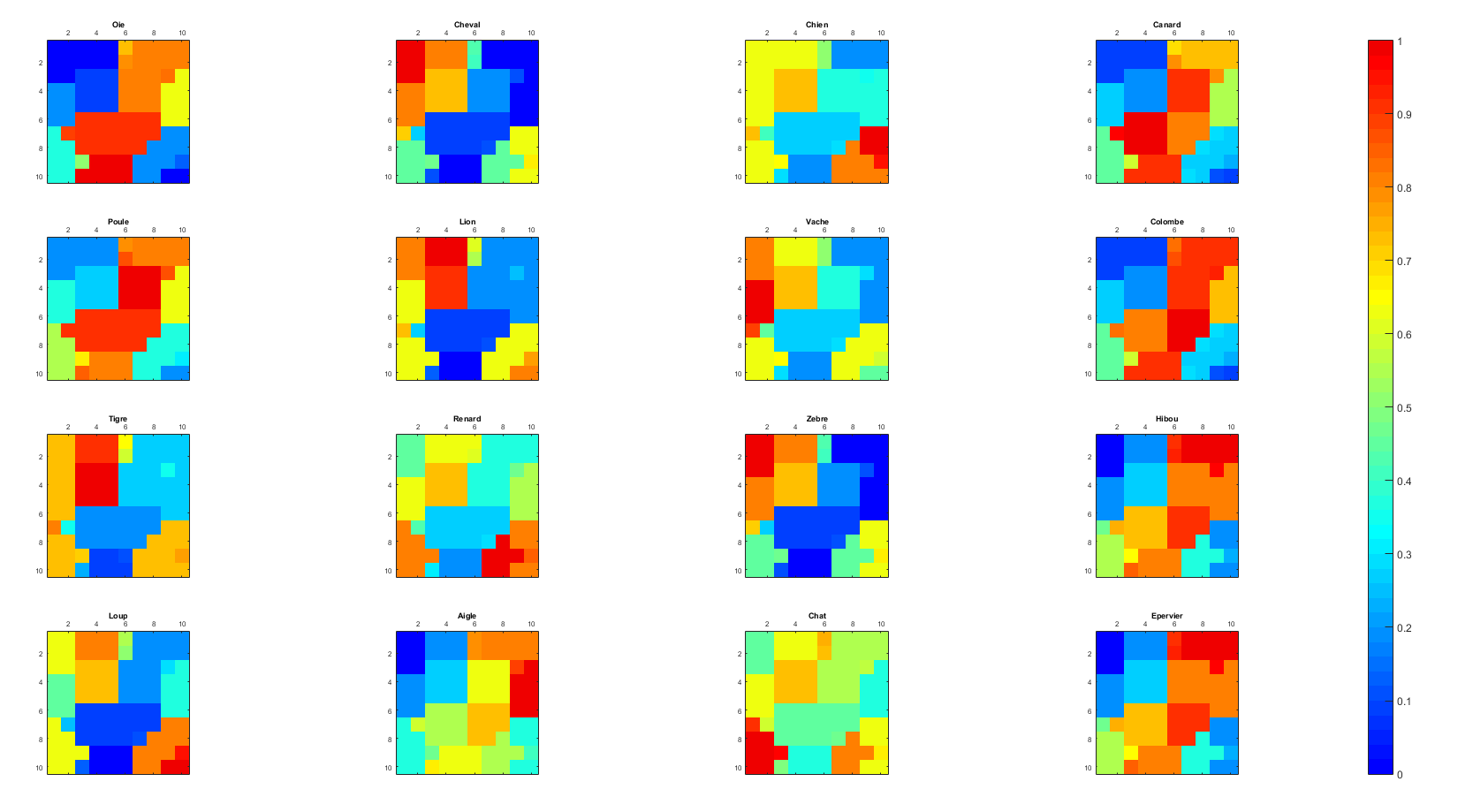


Carte de Kohonen avec un rayon d’apprentissage nul

Nous constatons que la carte de Kohonen avec un rayon d’apprentissage nul préserve la topologie du réseau mais d’une façon mois évidente. Alors que dans la première carte, om pouvait apercevoir des zones représentants chaque type d’animal, on se retrouve avec cette deuxième carte avec des zones plus réduites allant jusqu’à un seul carré dans la plupart des cas. On remarque que la carte n’est pas aussi éloquente que la première même si généralement les classes sont projetées au même endroit.

Remarque : Afin d’utiliser les fonctions de ce labo j’ai dû modifier le code Matlab de la fonction som pour l’affichage de la carte de Kohonen car ‘obtenais une erreur ne me permettant pas de voir les résultats. Pour cela, j’ai du vérifier le code de cette fonction et procéder à quelques modifications.

# La fonction d’apprentissage



Carte de Kohonen avec la fonction d’apprentissage implémentée par défaut

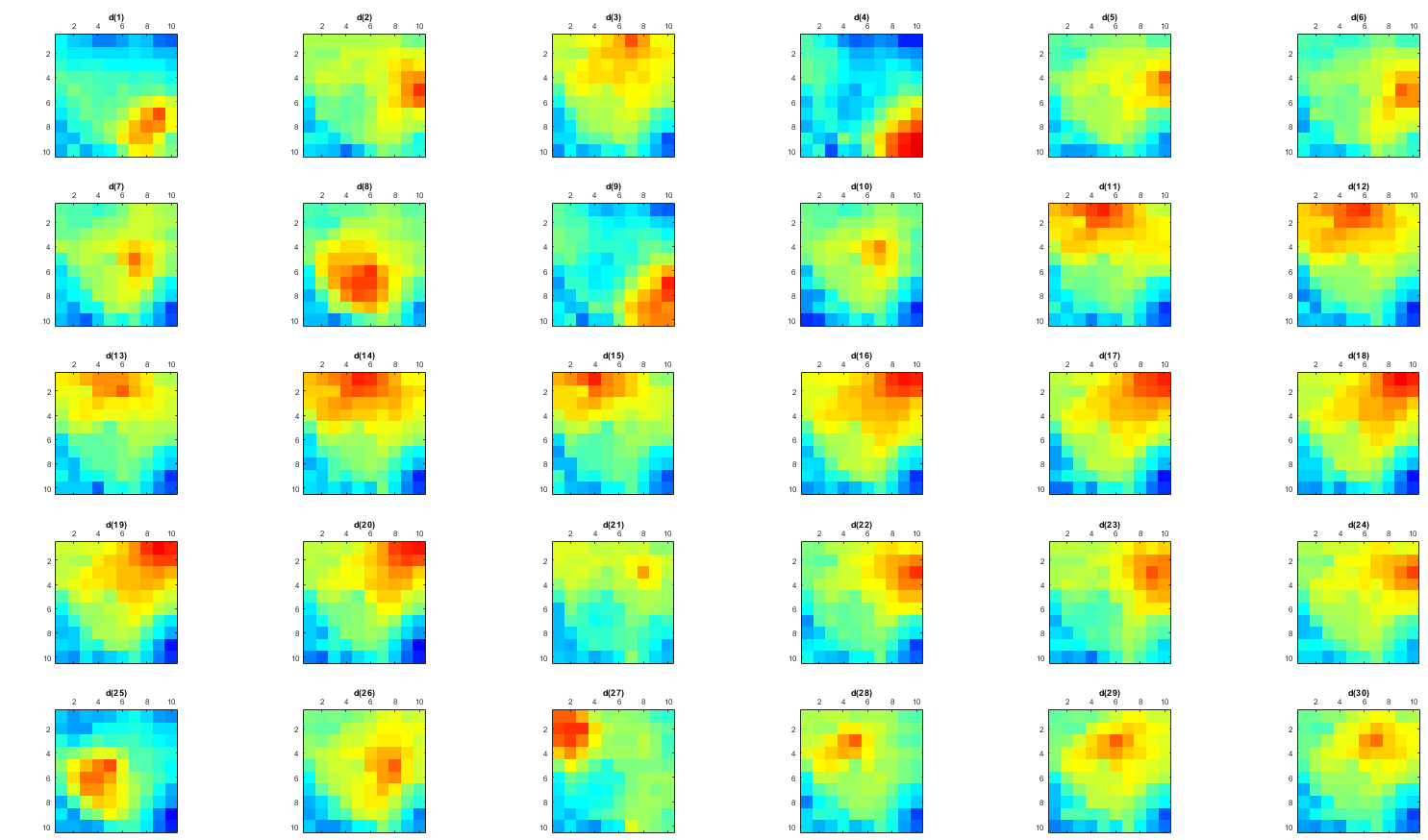
La topologie de la carte est préservée sauf que cette carte est moins granulée que la première carte. C’est-à-dire que cette carte n’est pas aussi détaillée que la première. Cette carte peut servir à obtenir des résultats moins flous et plus stricts. On pense que cette carte est capable de distinguer les frontières de chaque classe plus précisément.

1. Nous observons sur cette figure la variation de la fonction d’apprentissage au cours des epochs. Cette fonction décroît très rapidement. Elle atteint zéro à l’epoch 50.

C:\Users\Laimi\Desktop\alpha.emf

Variation de la fonction d’apprentissage alpha

# Un dernier essai



Carte de Kohonen pour les chiffres manuscrits

1. La disposition des données n’est pas très révélatrice de la classe d’appartenance des différents échantillons. Pour certaines classes les données sont très dispersées pour obtenir une frontière significative permettant de séparer les données. Comme il y a une certaine ressemblance entre certaine classes, par exemple zéro et huit, certains individus d’une classe peuvent converger vers le prototype d’une autre classe. Cette carte peut servir à distinguer les classes qui ont de fortes tendances à converger l’une vers l’autre. Si on arrive à distinguer les classes qui peuvent prêter à confusion et converger vers le prototype d’une autre classe, on peut se servir de cette information pour choisir de meilleurs prototypes permettant de limiter cet effet.