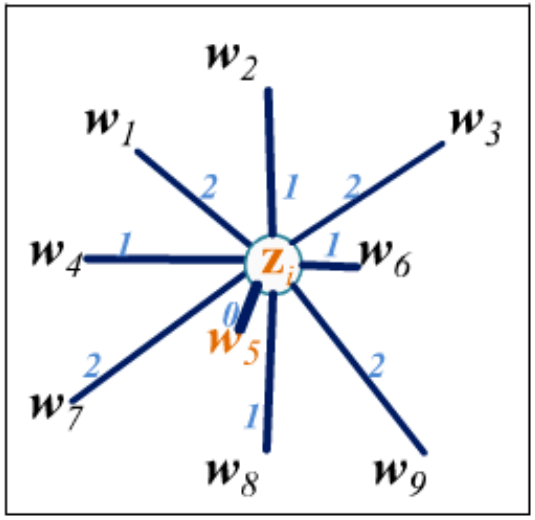
Illustration de la distance généralisée

Pour chaque observation $\mathbf{z}_i$, on calcule une « distance généralisée » à l'ensemble des neurones de la carte par l'intermédiaire de la fonction K^T:

$$d^T(\mathbf{z}_i, \boldsymbol{w}_r) = \sum_{c \in C} K^T(\delta(c, r)) \left\| \mathbf{z}_i - \boldsymbol{w}_c \right\|^2 \qquad (2)$$

Illustration avec $r=5$



Carte C

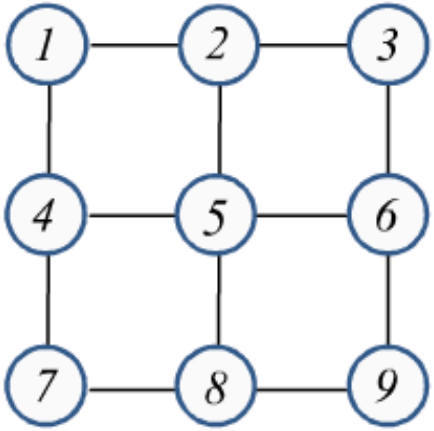
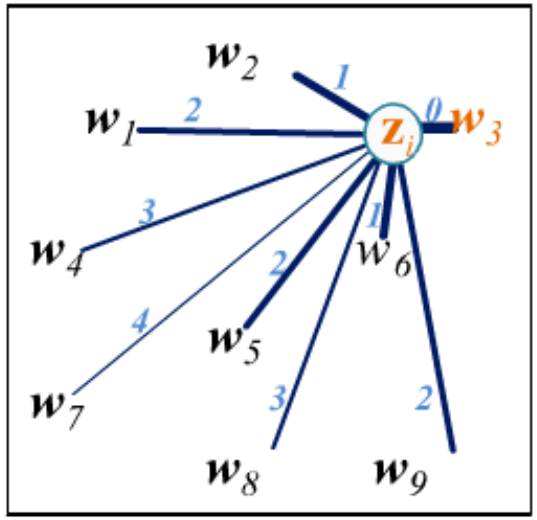
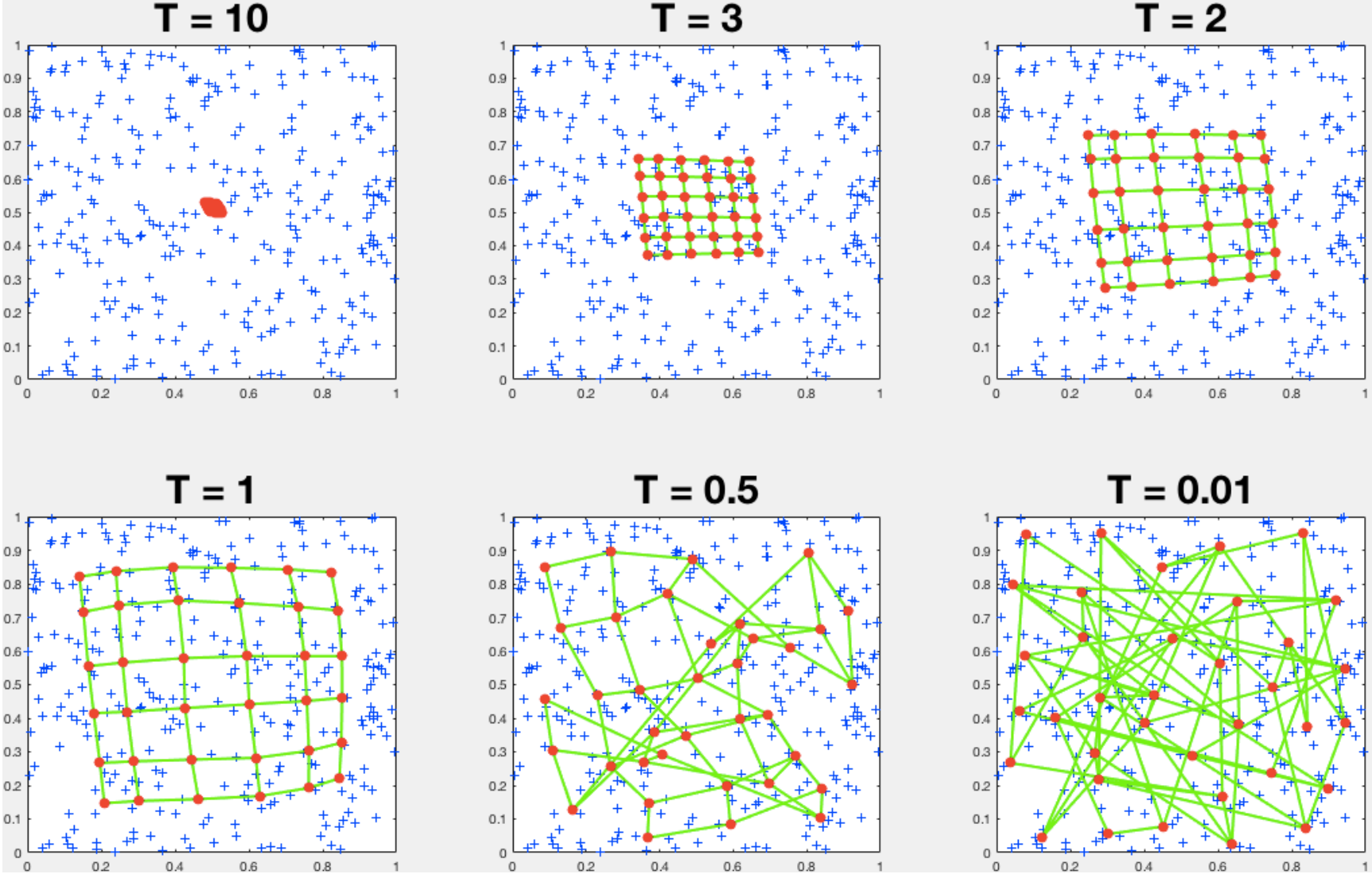


Illustration avec $r=3$



Optimisation non adaptative des cartes topologiques

Minimisation itérative de $J_{som}(\chi, W)$ pour un paramètre **T fixé**



Algorithmes de minimisation à valeur de T décroissante

$$J^T_{som}(\chi,\mathcal{W}) = \sum_{\mathbf{z}_i \in \mathcal{A}} \sum_{c \in C} K^T(\delta(c,\chi(\mathbf{z}_i)))\,\|\mathbf{z}_i - \mathbf{w}_c\|^2 = \sum_{\mathbf{z}_i \in \mathcal{A}} d^T(\mathbf{z}_i, \mathbf{w}_{\chi(\mathbf{z}_i)})$$

- Algorithme d'optimisation totale (version batch)
- Algorithme de Kohonen (version stochastique)

- **La minimisation est répétée un certain nombre de fois en faisant décroître la valeur de T.**
- **L'ordre apparaît pour les grandes valeurs de T, la carte peut alors se déployer pour recouvrir les données et faire diminuer la variance intra.**

Les paramètres déterminants de la minimisation sont :

- *L'intervalle de variation de T, la valeur initiale T_{max} et la valeur finale T_{min}*
- *Le nombre de fois où l'étape itérative est effectuée*
- *La fonction de décroissance du paramètre T dans l'intervalle $[T_{max}, T_{min}]$*

Algorithme d'optimisation totale (version batch)

Algorithme d'optimisation totale (version batch)

1. **Initialisation**

   t=0 : indice d'itération

   N_{iter} : nombre maximum d'itérations

   Choisir la structure et la taille p de la carte

   Choisir les p référents initiaux (en général d'une manière aléatoire) : $\mathcal{W}^{t=0}$

   Choisir les bornes de l'intervalle de température Tmin et Tmax.

2. **Etape de minimisation itérative t.** (à partir de t=1 ; $\mathcal{W}^{t-1}$ est donc connu)

   L'ensemble des référents $\mathcal{W}^{t-1}$ de l'étape précédente est connu

   - Calculer la valeur de T à utiliser par les phases d'affectation et de minimisation qui suivent, grâce à une fonction de décroissance : $T = f_{dec}(t, N_{iter}, T_{min}, T_{max})$
   - **Phase d'affectation** : mise à jour de la fonction d'affectation χ^t qui consiste à associer chaque observation $\mathbf{z}$ à un référent de $\mathcal{W}^{t-1}$ selon l'expression suivante :

   $$\chi^T(\mathbf{z}) = \arg\min_{r \in C} \sum_{c \in C} K^T(\delta(c,r)) \|\mathbf{z} - \mathbf{w}_c\|^2 = \arg\min_{r \in C} d^T(\mathbf{z}, \mathbf{w}_r)$$

   - **Phase de minimisation** : calcul des nouveaux référents $\mathcal{W}^t$, associés à χ^t, en appliquant la formule suivante :

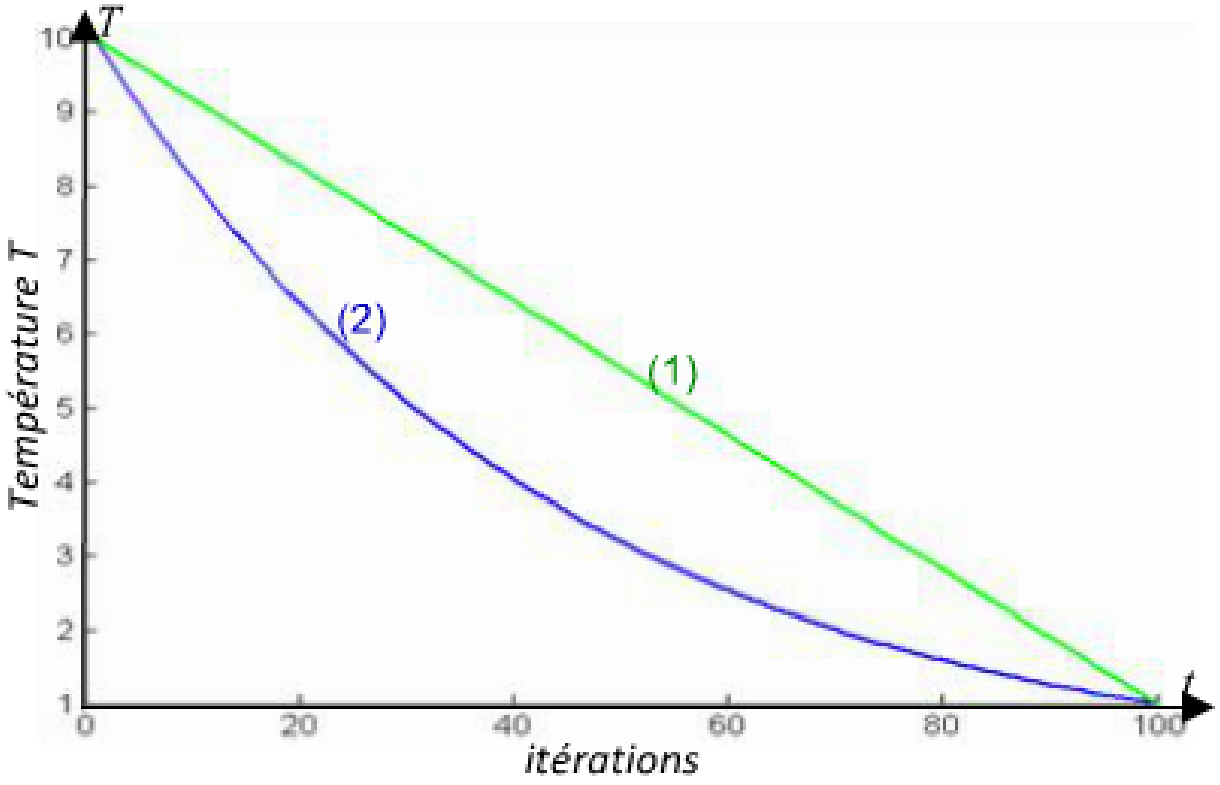
   $$\mathbf{w}_c^T = \frac{\sum_{r \in C} K^T(\delta(c,r)) \mathbf{Z}_r}{\sum_{r \in C} K^T(\delta(c,r)) n_r}$$

   n_r : le nombre d'observations de l'ensemble d'apprentissage affectées au neurone r

   Z_r : la somme de toutes les observations affectées au neurone r $\quad (= \sum_{\mathbf{z}_i \in \mathcal{A};\ \chi(\mathbf{z}_i)=r} \mathbf{z}_i)$

3. **Répéter** l'étape de minimisation itérative jusqu'à atteindre N_{iter} itérations ou une stabilisation de la fonction J_{som}^T.

Exemples de fonction de décroissante f_{dec}



(1) $$T = T_{\max} - (T_{\max} - T_{\min}) \frac{t-1}{Niter-1}$$

(2) $$T = T_{\max} \left(\frac{T_{\min}}{T_{\max}} \right)^{\frac{t-1}{N_{iter}-1}}$$

Algorithme de Kohonen : version stochastique

A χ constant, faire décroître la valeur de J^T_{som} par une méthode de gradient :

$$\mathbf{w}_c^t = \mathbf{w}_c^{t-1} - \mu^t \frac{\partial J^T_{som}}{\partial \mathbf{w}_c^{t-1}} \quad (5)$$

$$\frac{\partial J^T_{som}}{\partial \mathbf{w}_c} = 2\sum_{\mathbf{z}_i \in A} K^T\left(\delta(c, \chi(\mathbf{z}_i))\right)\left(\mathbf{z}_i - \mathbf{w}_c^{t-1}\right) \quad (6)$$

Pour une observation $\mathbf{z}_i$:

- détermination du neurone r gagnant par la fonction d'affectation :

$$\chi(\mathbf{z}_i) = arg\, min_c \left\| \mathbf{z}_i - \mathbf{w}_c \right\|^2 = r \quad (7)$$

- l'ensemble des référents $\mathbf{w}_c$ est recalculé, en fonction du neurone r gagnant, et en utilisant le gradient stochastique :

$$\mathbf{w}_c^{\,t} = \mathbf{w}_c^{\,t-1} - 2\mu^t K^T(\delta(c,r))(\mathbf{z}_i - \mathbf{w}_c^{\,t-1}) \quad (8)$$

Algorithme de Kohonen : version stochastique (suite)

1 Initialisation :

- t = 0 : indice d'itération
- N_{iter} = nombre maximum d'itérations
- définir p référents initiaux :⇒ $\mathcal{W}^{t=0}$
- choisir les bornes de l'intervalle de température $Tmin$ et $Tmax$

2 Étape de minimisation itérative t (à partir de t= 1; $\mathcal{W}^{t-1}$ est donc connu):

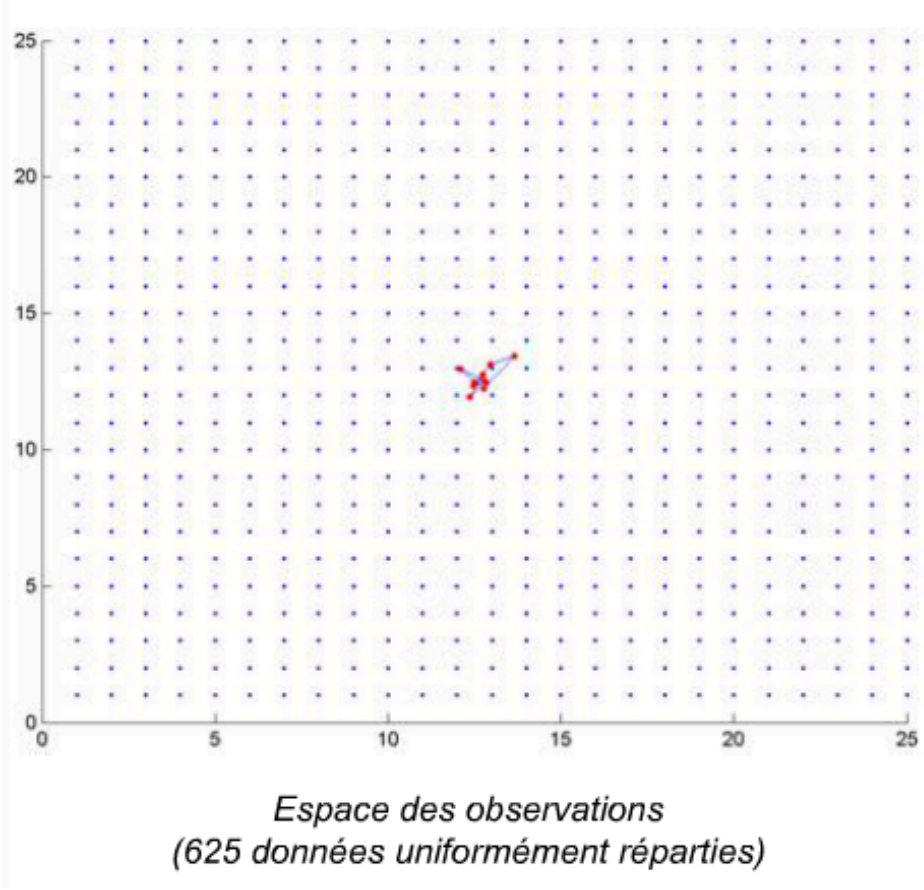
- choisir une observation $\mathbf{z}_i$
- calculer la nouvelle valeur de T à utiliser par les phases d'affectation et de minimisation qui suivent, grâce à une fonction de décroissance : $T=f_{dec}(t, Miter, Tmin, Tmax)$.
- *Phase d'affectation :*

  associer l'observation $\mathbf{z}_i$ au neurone $r=\chi^t(\mathbf{z}_i)$ définie par l'expression (7).
- *Phase de minimisation :*

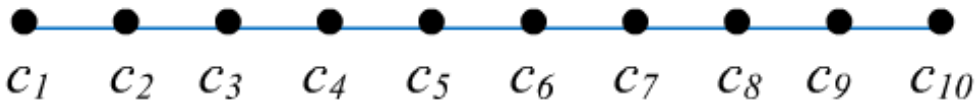
  calcul des nouveaux référents $\mathcal{W}^t$ en appliquant la formule (8) à chaque référent $\boldsymbol{w}_c$ qui utilise les distances sur la carte au neurone r sélectionné à l'étape d'affectation.

Répéter tant que $t < N_{iter}$ ou stabilisation de J^T_{som}

Un exemple simplifié : carte 1D

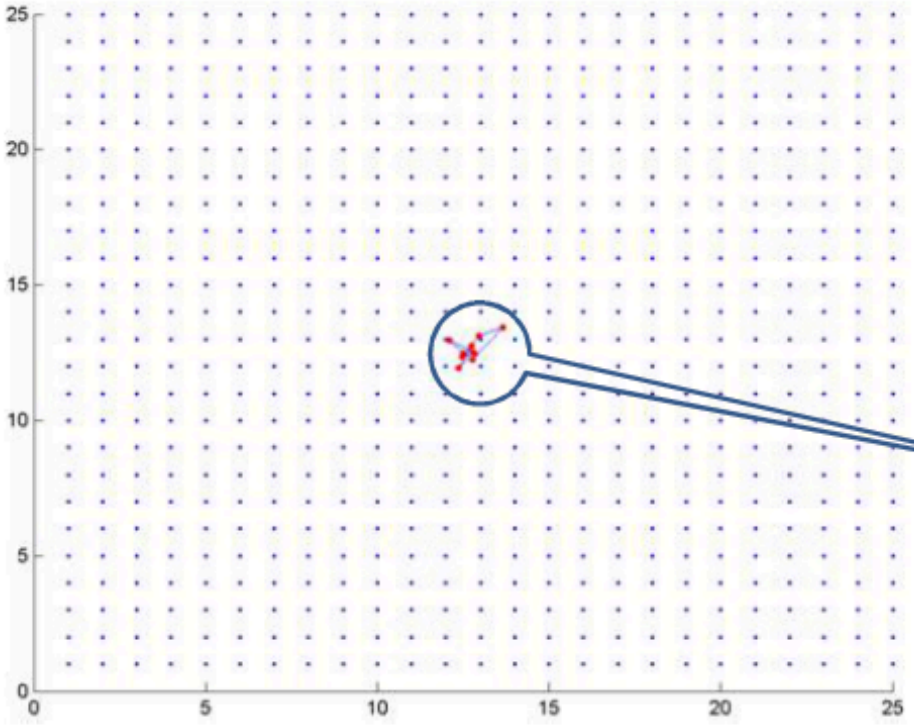


Espace des observations
(625 données uniformément réparties)

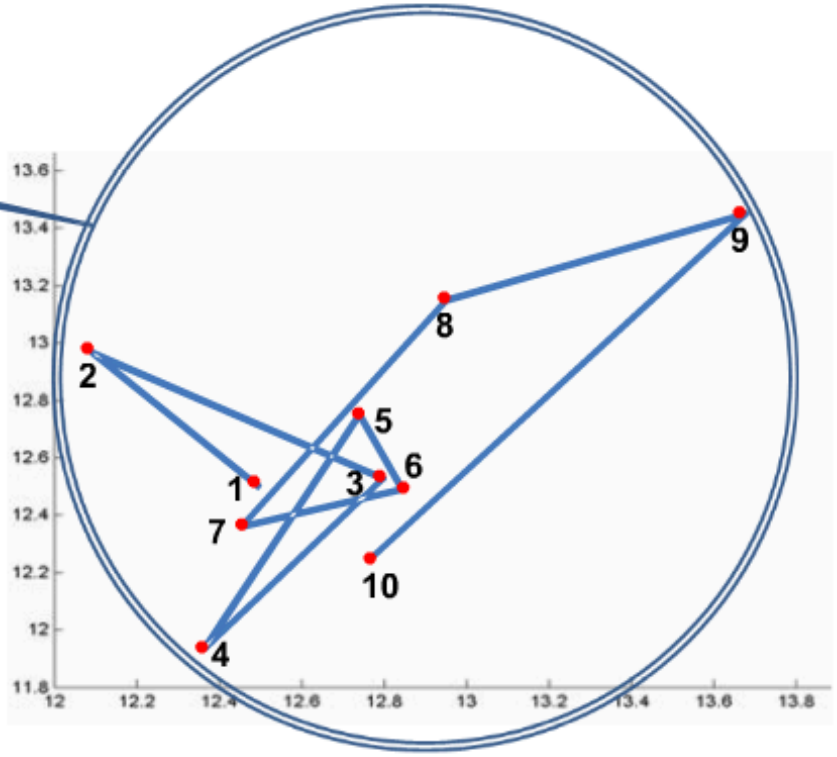


Architecture de la carte topologique unidimensionnelle

Un exemple simplifié : zoom sur les référents

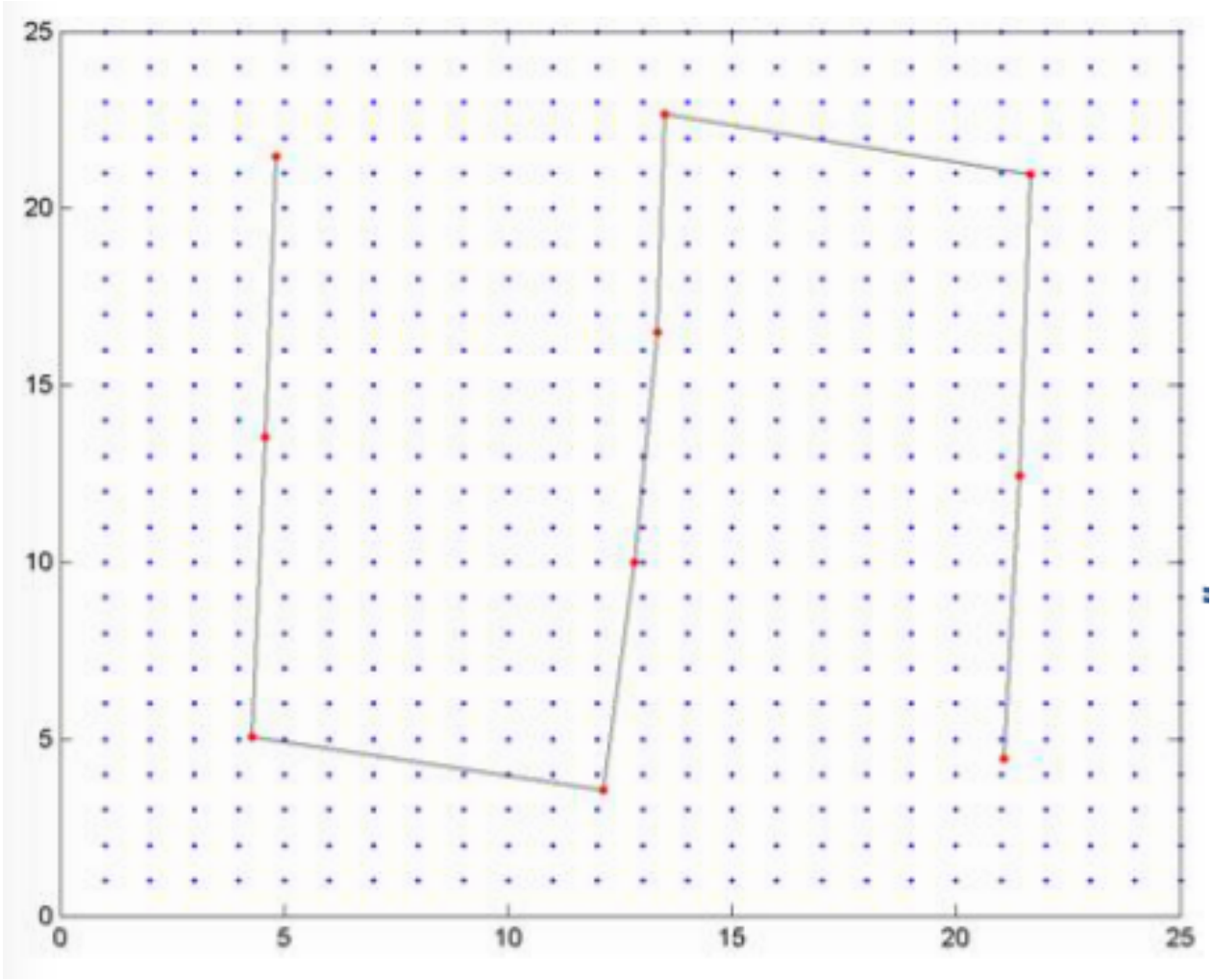


Espace des observations
(625 données uniformément réparties)



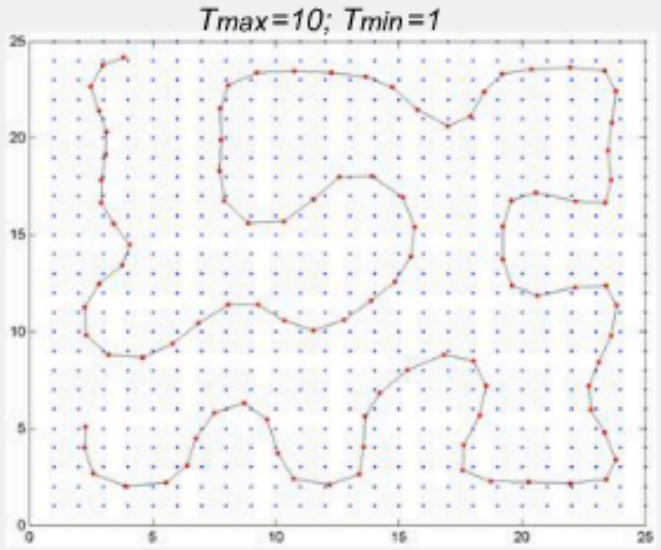
Initialisation aléatoire des référents ; ordre induit

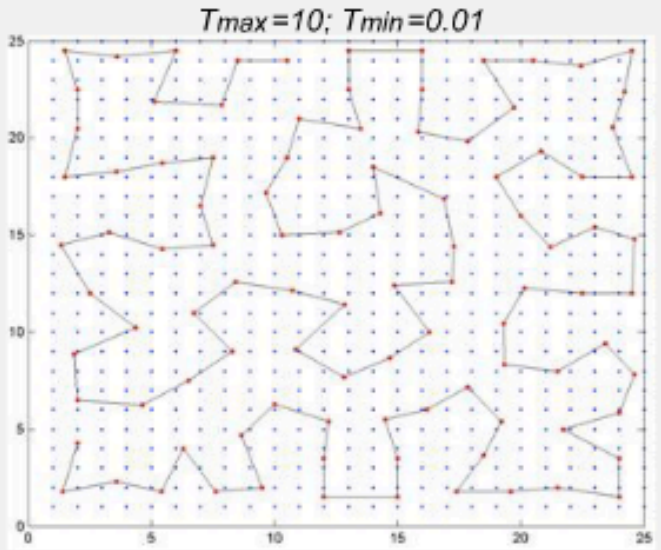
Un exemple simplifié : carte obtenue



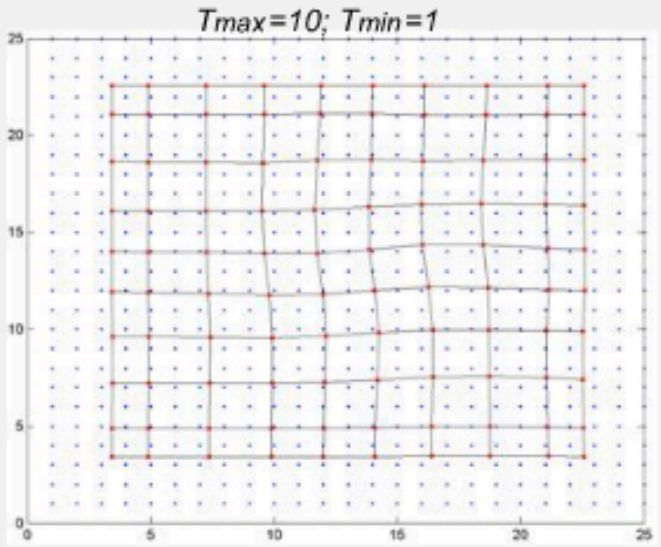
Un exemple simplifié : carte 1D et carte 2D

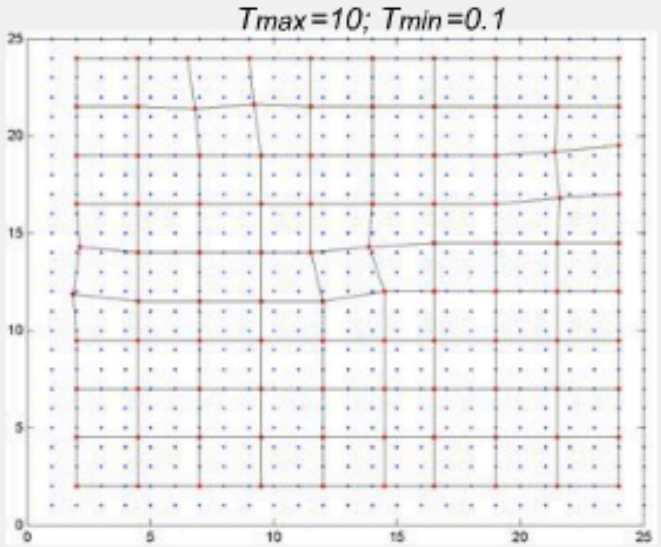
Carte topologique 1D



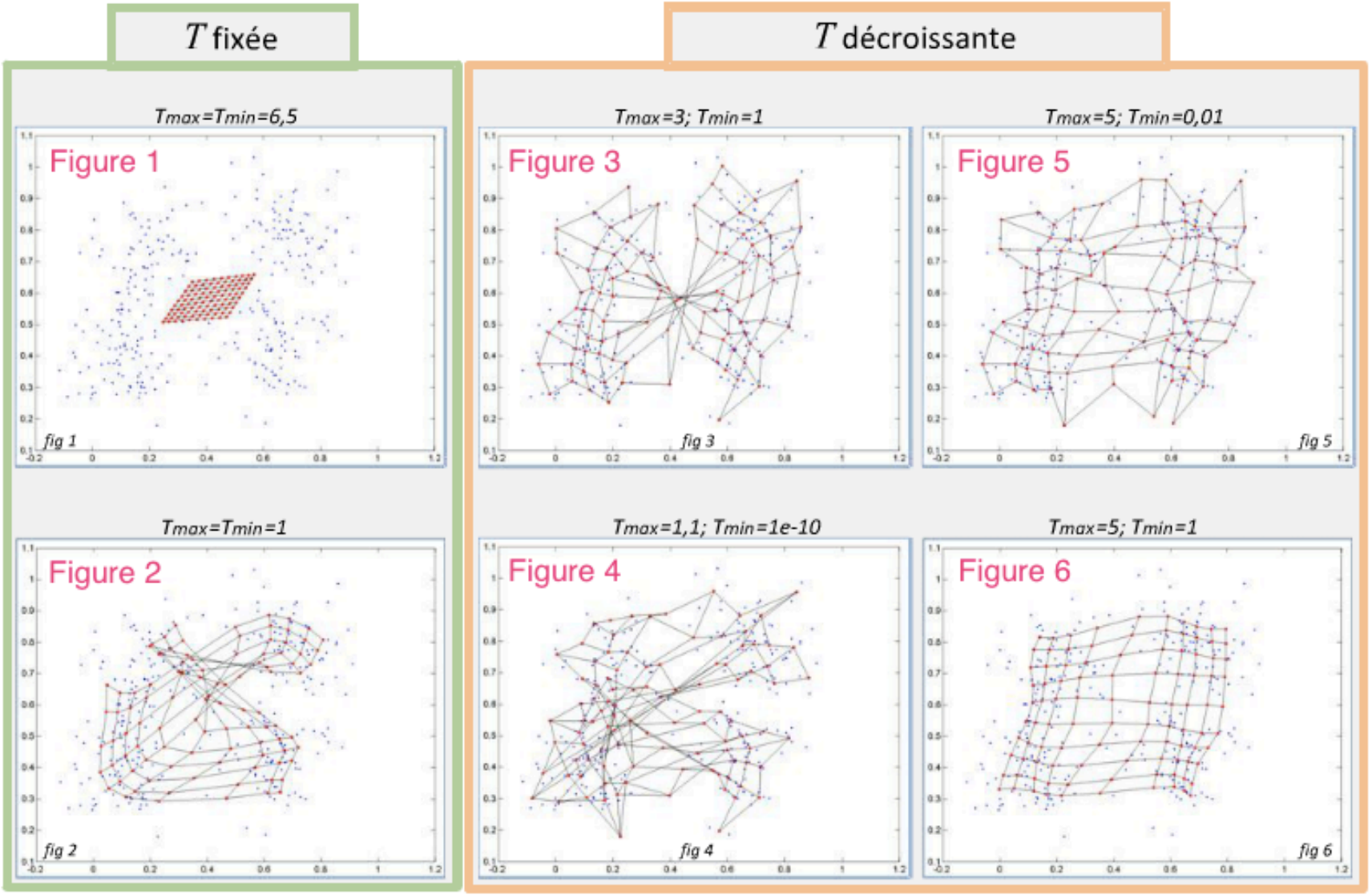


Carte topologique 2D

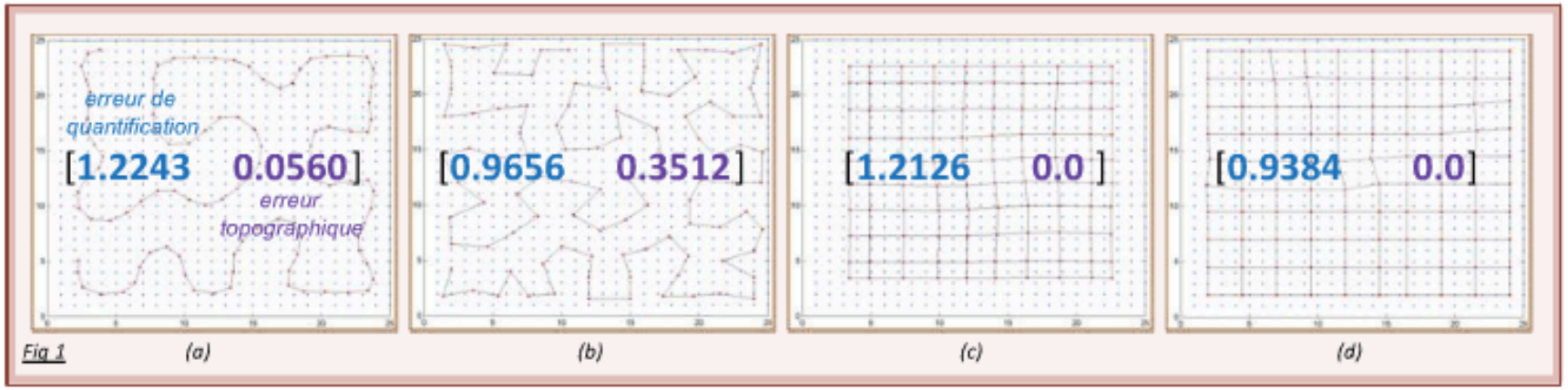


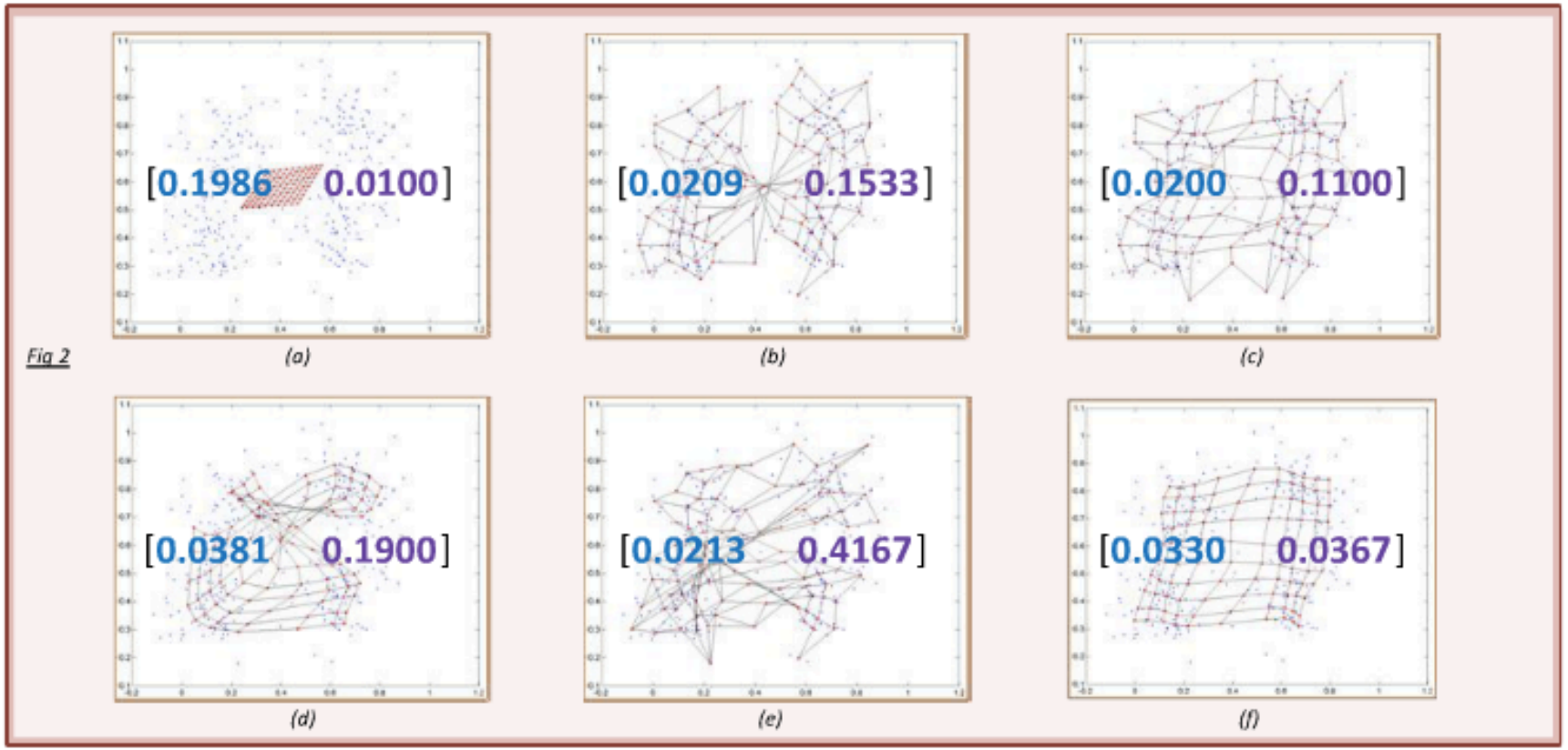


Exemples (suite) : T fixé et T décroissante sur différents intervalles, avec des données Gaussiennes



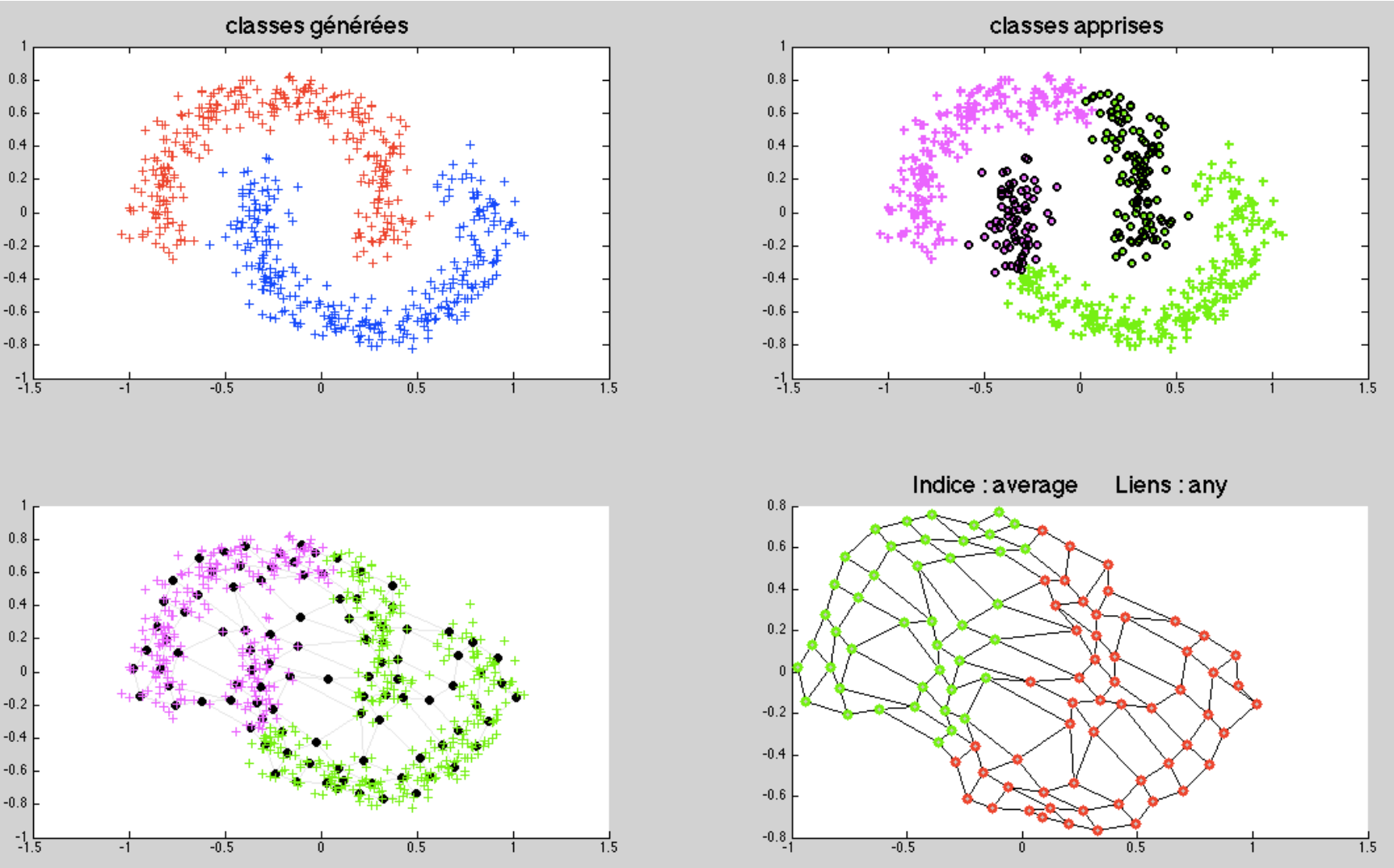
Qualité de la carte





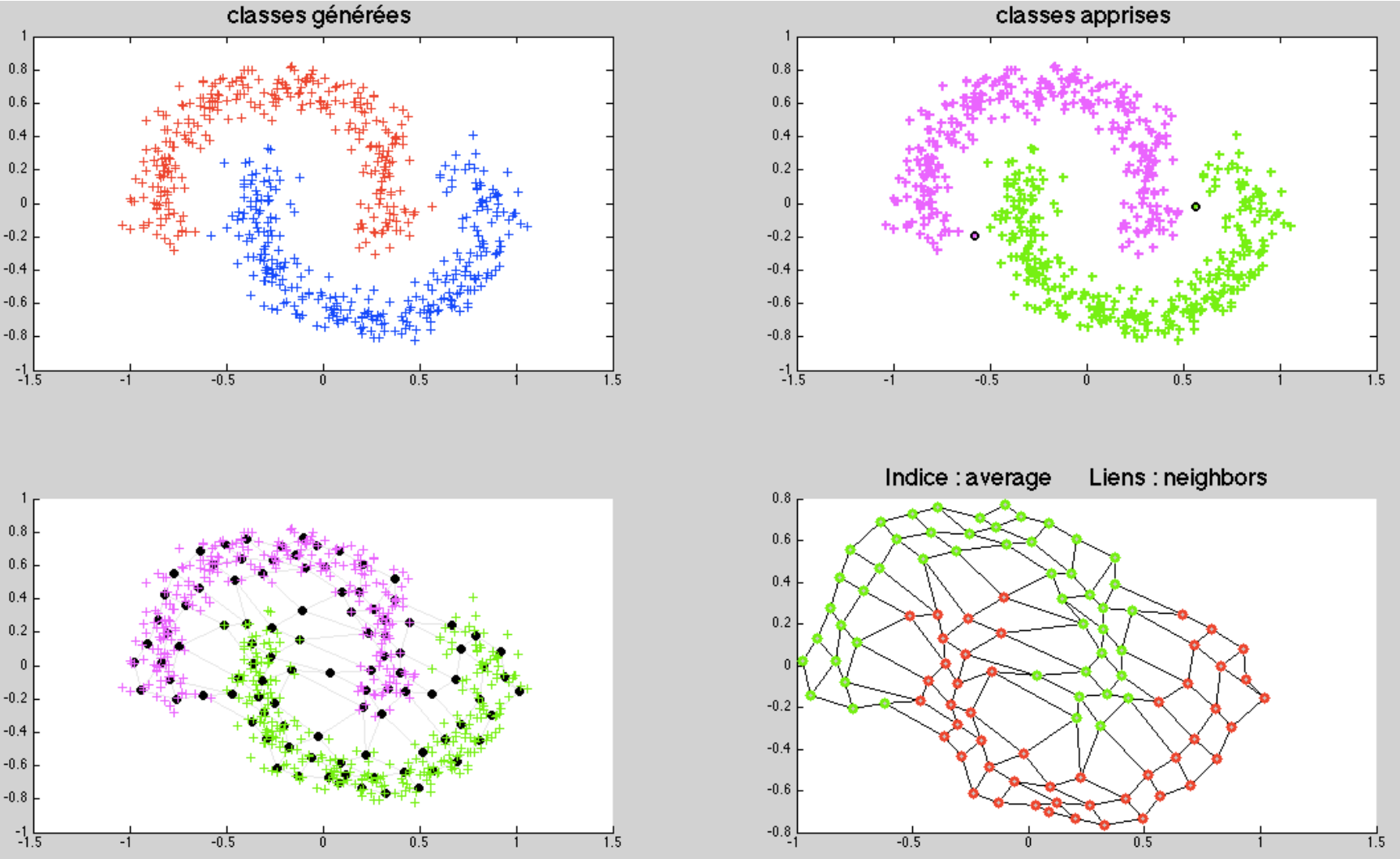
CAH sur les neurones d'une carte topologique (voir fiche de TP)

CAH classique



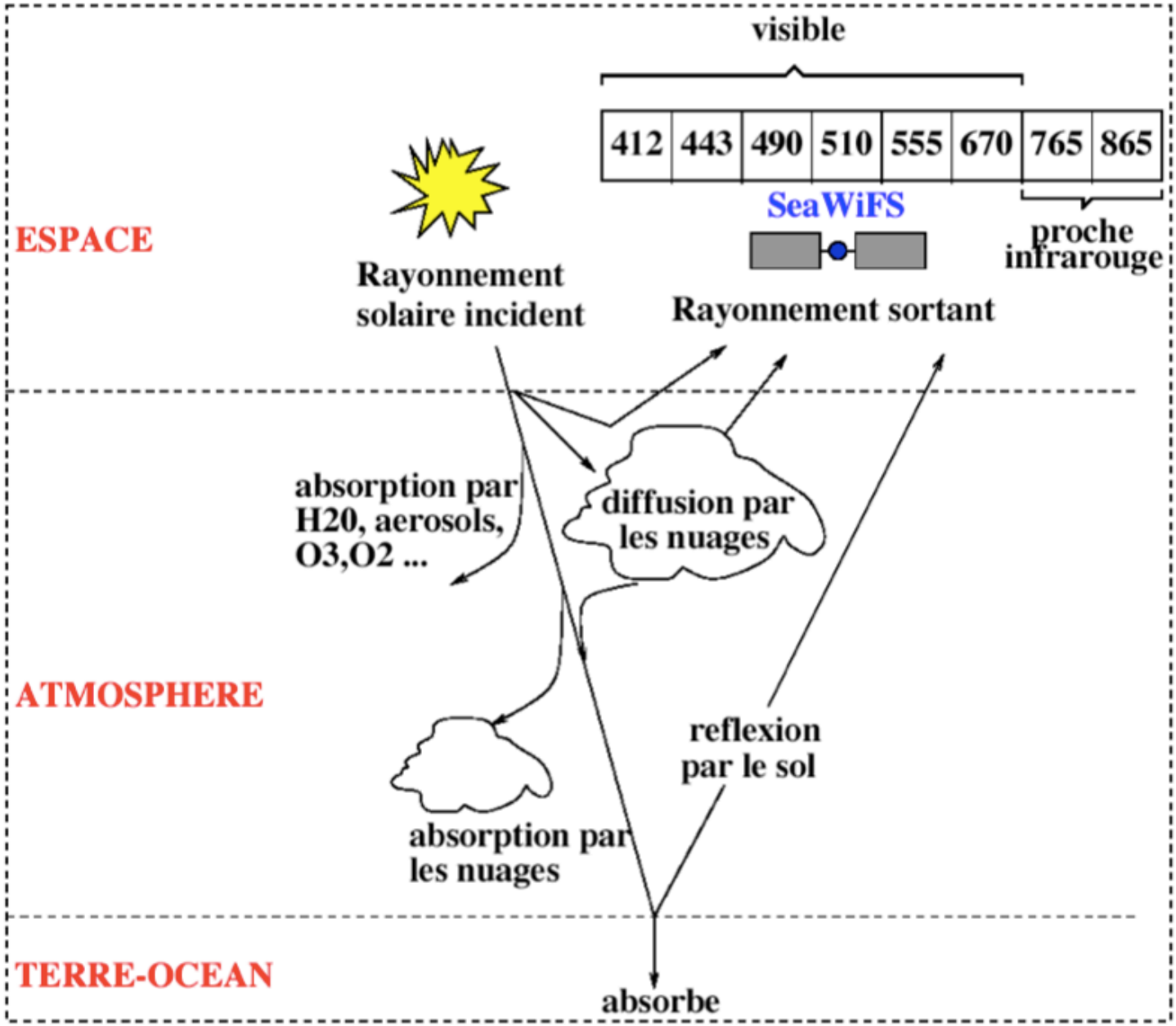
CAH sur les neurones d'une carte topologique (voir fiche de TP)

CAH en tenant compte de la carte



Une application en télédétection satellitaire
Etude de la couleur de l'ocan
(Voir fiche de TP)

Le problème physique

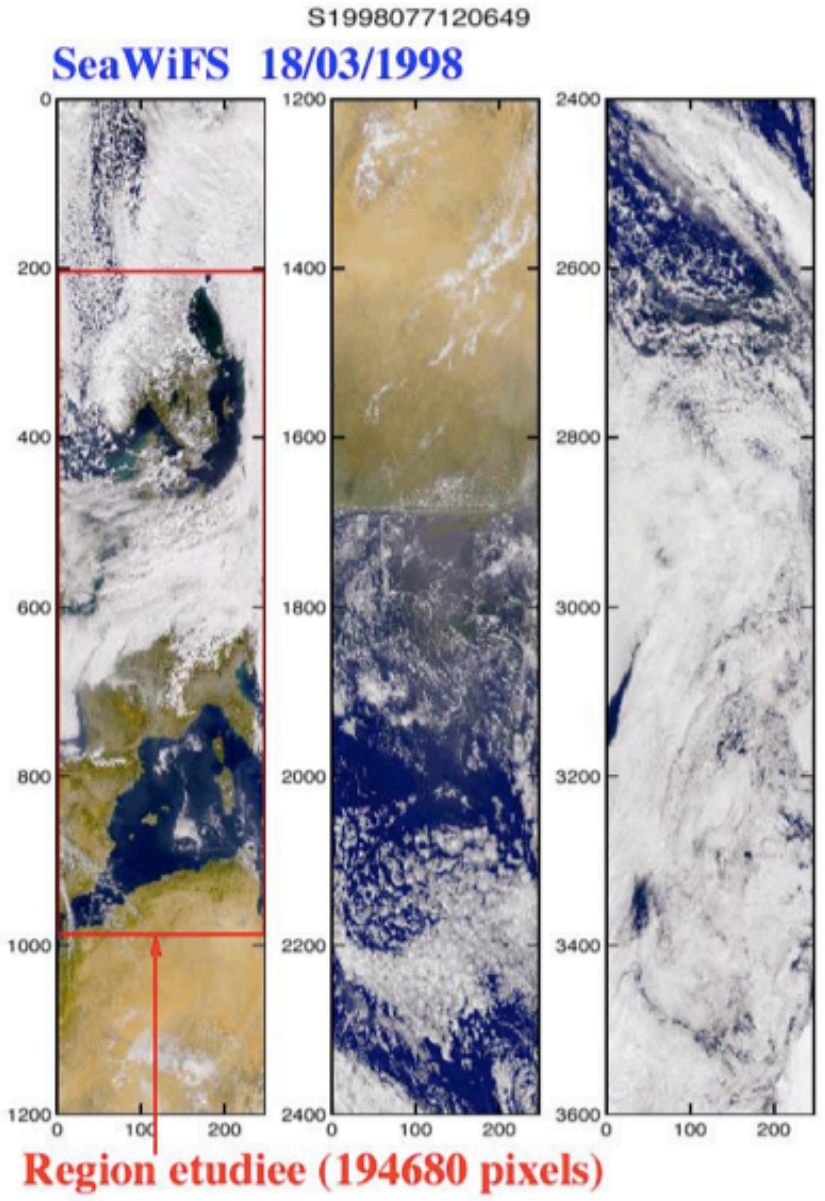


Les données

Image SeaWifs (Polder): analyse de la même scène selon 8 (5) longueurs d'onde

On cherche à déterminer les différents types d'aérosols

Bandes k	Longueurs d'ondes (nanomètres) λ_k
1	412
2	443
3	490
4	510
5	555
6	670
7	765
8	865



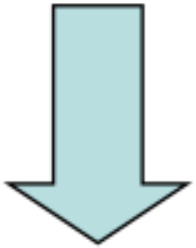
Les premières expériences : étude sur une image

Les données: image SeaWifs sur la Afrique de l 'ouest et les Iles Canaries 5 janvier 1999 échantillonnée d'une manière homogène (1 ligne sur 10). Chaque observation est constitué par les 8 longueurs d'onde.

Apprentissage: carte topologique 10*10 (algorithme PRSOM),

Classification hiérarchique : 2 classes (Séparation mer nuages)

4 classes (Recherche des types d 'aérosols)



TEST

Image entière

Image SeaWifs Afrique de l 'ouest et Iles Canaries

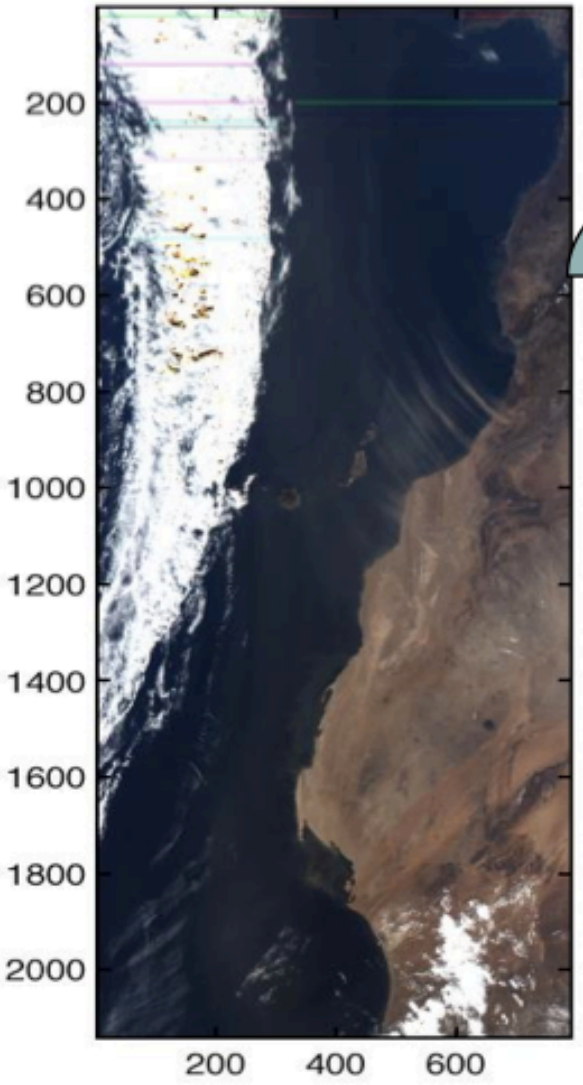


Image LAC
$(2141 \star 793)$

Image étudiée

Image expertisée par un autre algorithme (Merisse)

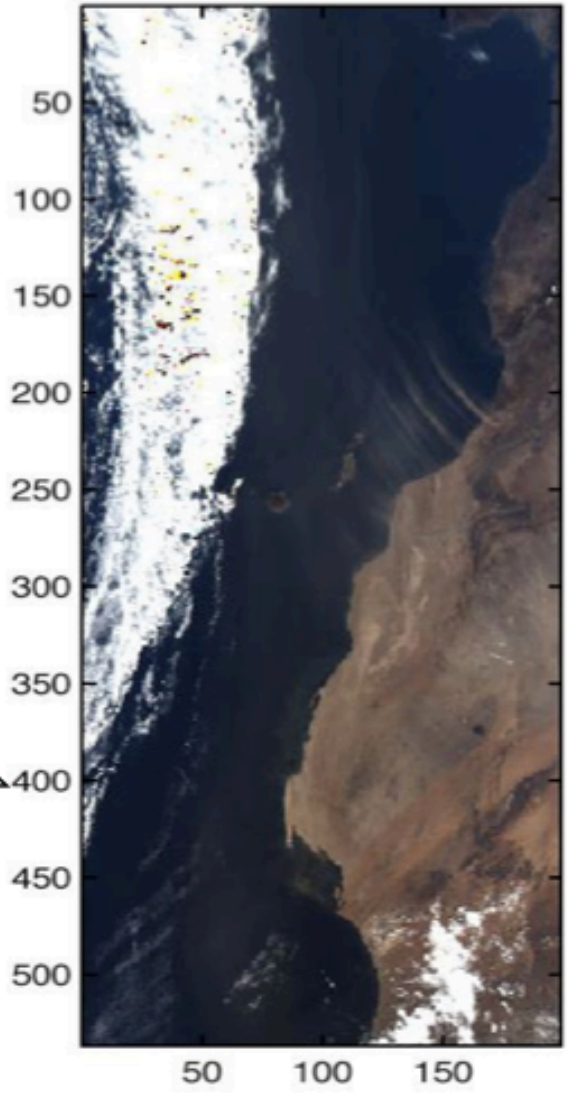
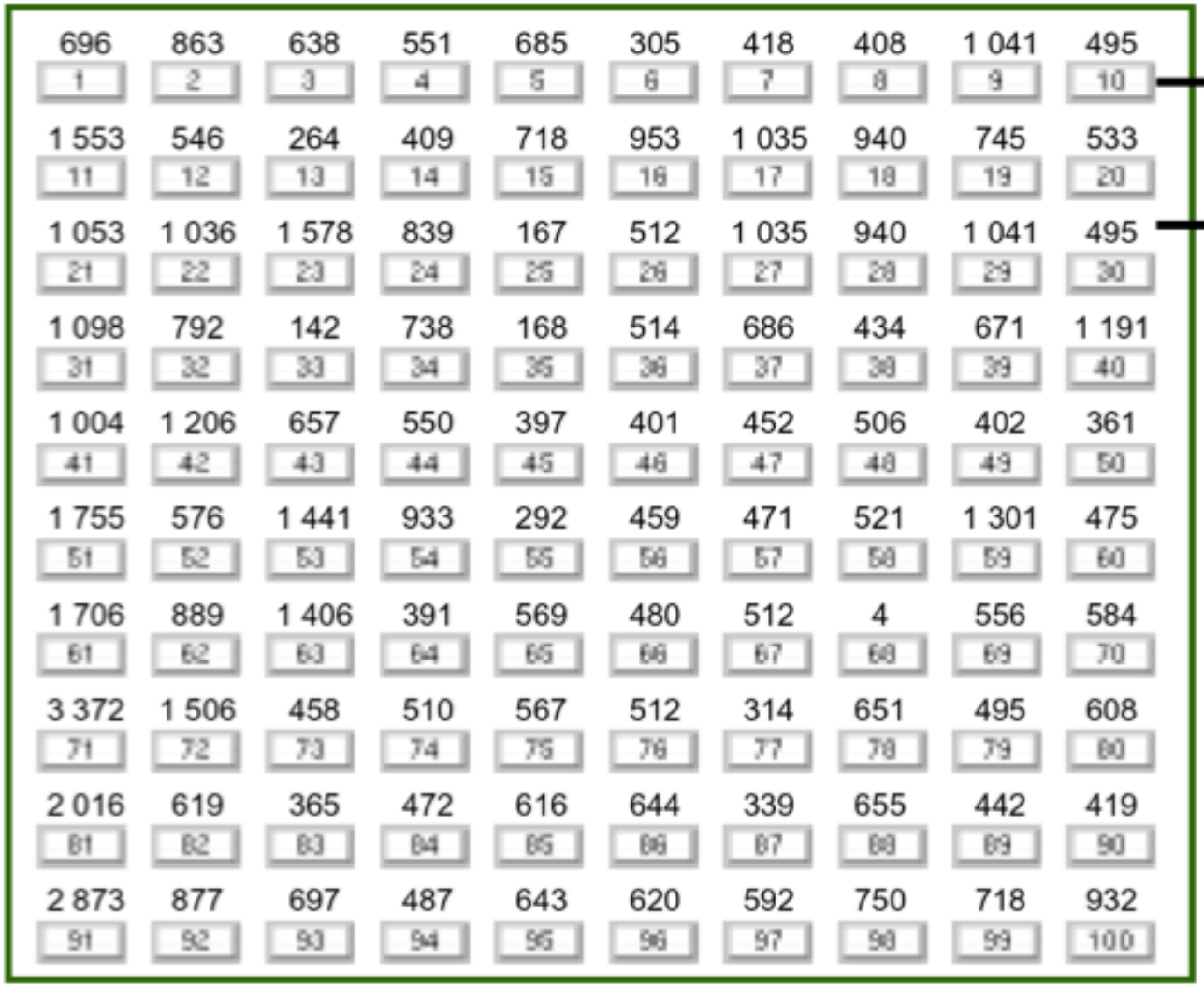


image GAC
$(536 \star 199)$

Visualisation de la carte

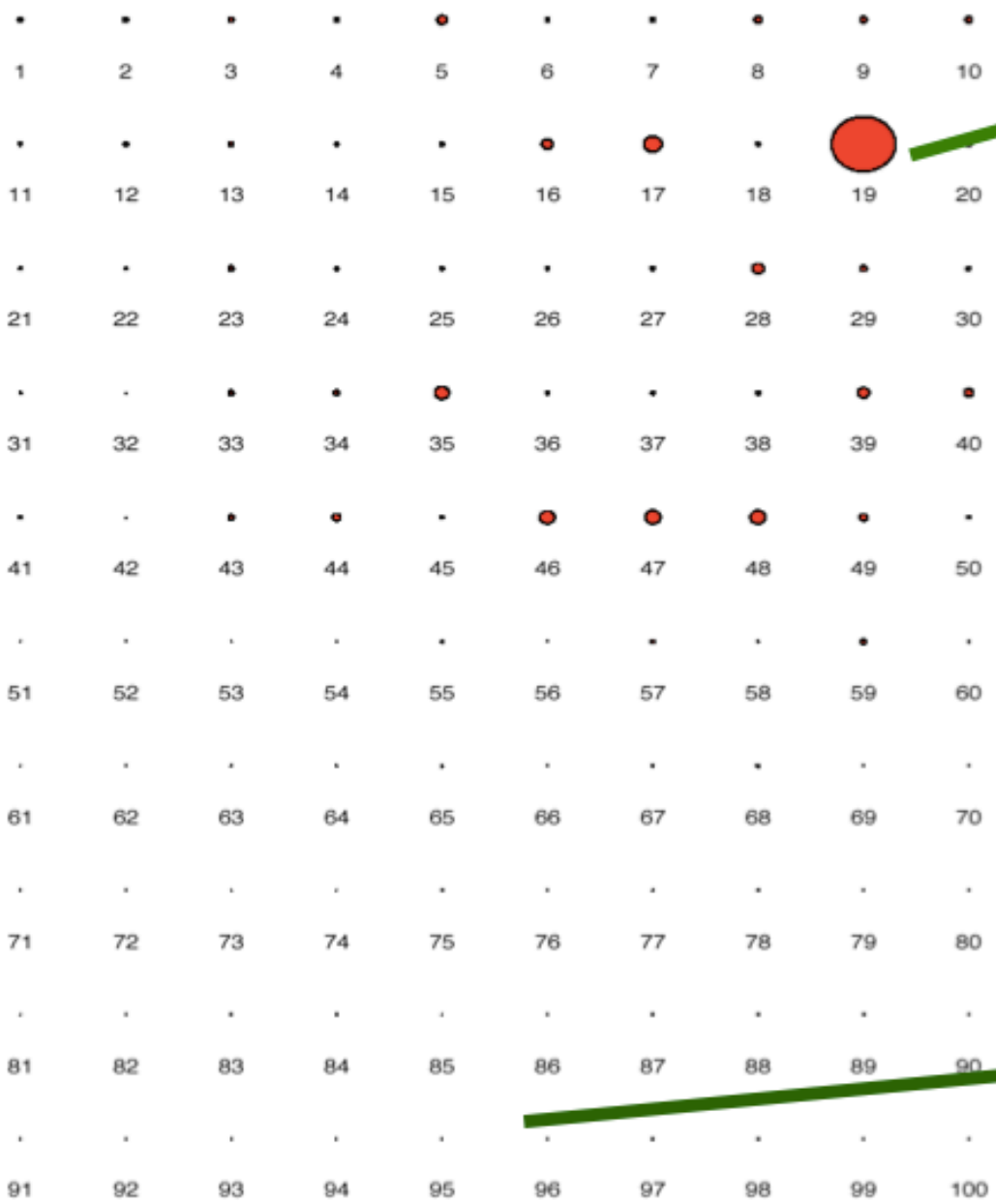


Numéro du neurone

Nombre de pixels captés

Le nombre de pixel capté par chaque neurone est une première indication sur la densité des données (spectre de R^8)

Visualisation des variances



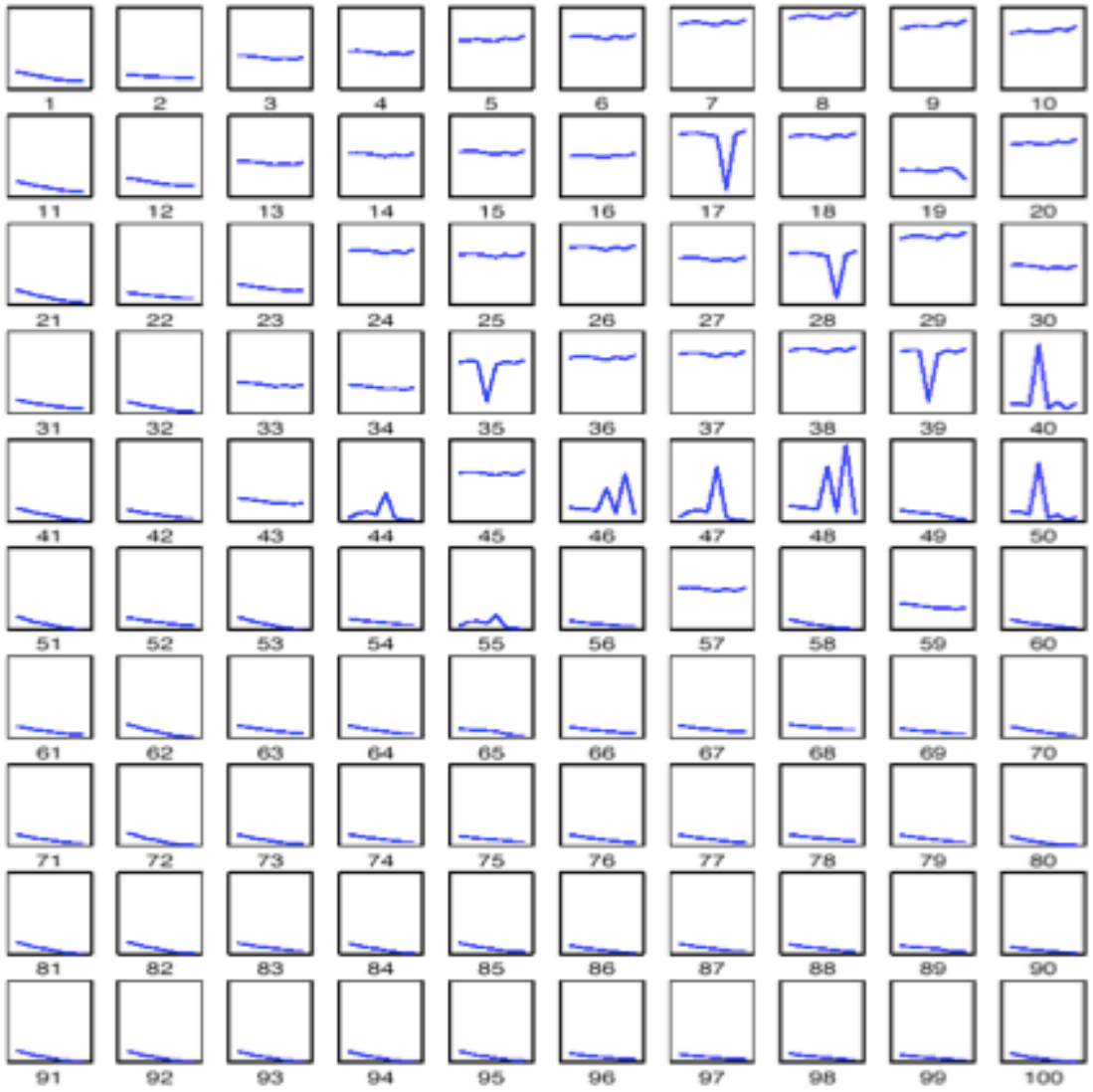
Zone de grande variabilité = **nuage**

Carte PRSOM (10*10)

La surface du disque est proportionnelle à la variance de la gaussienne attachée au neurone. Elle permet une première analyse de la carte

Zone très dense = **mer**

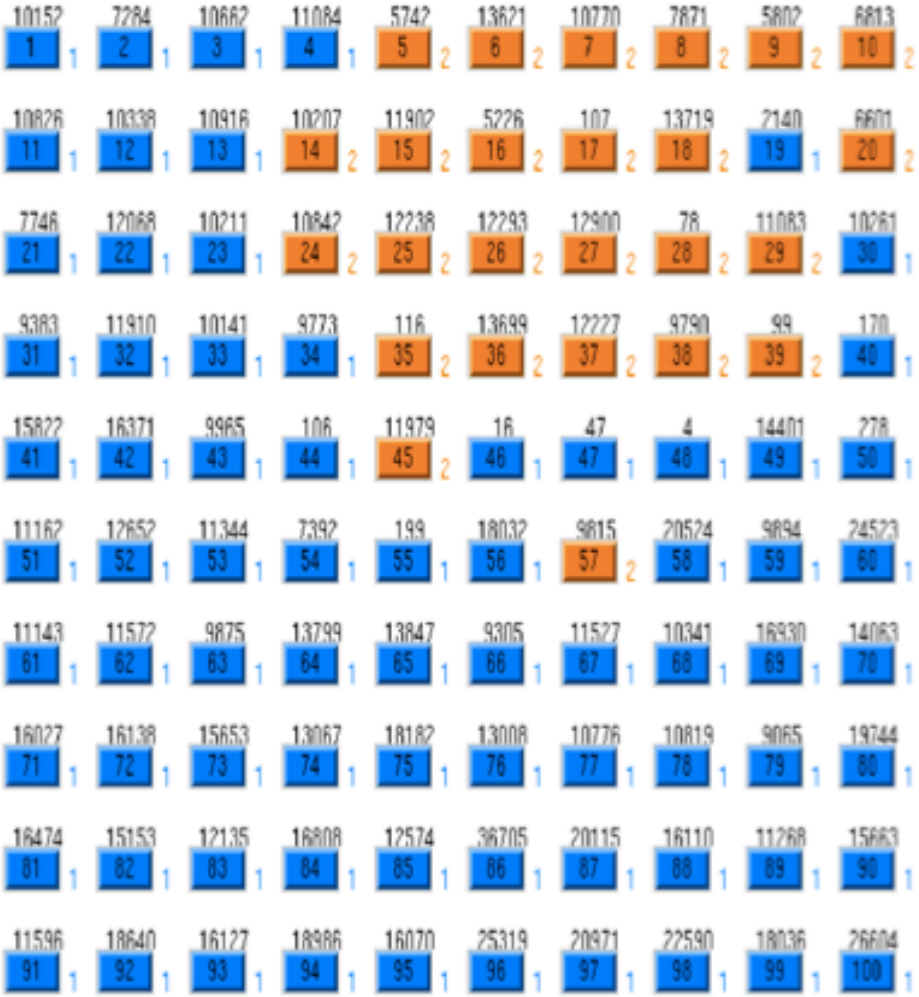
Visualisation simultanée des référents et de la carte



Permet d 'interpréter l 'ordre topologique obtenu en fonction des données. Chaque référent est un spectre de $\mathbf{R}^8$.

La forme des spectres des référents varie de manière ordonnée selon les deux directions.

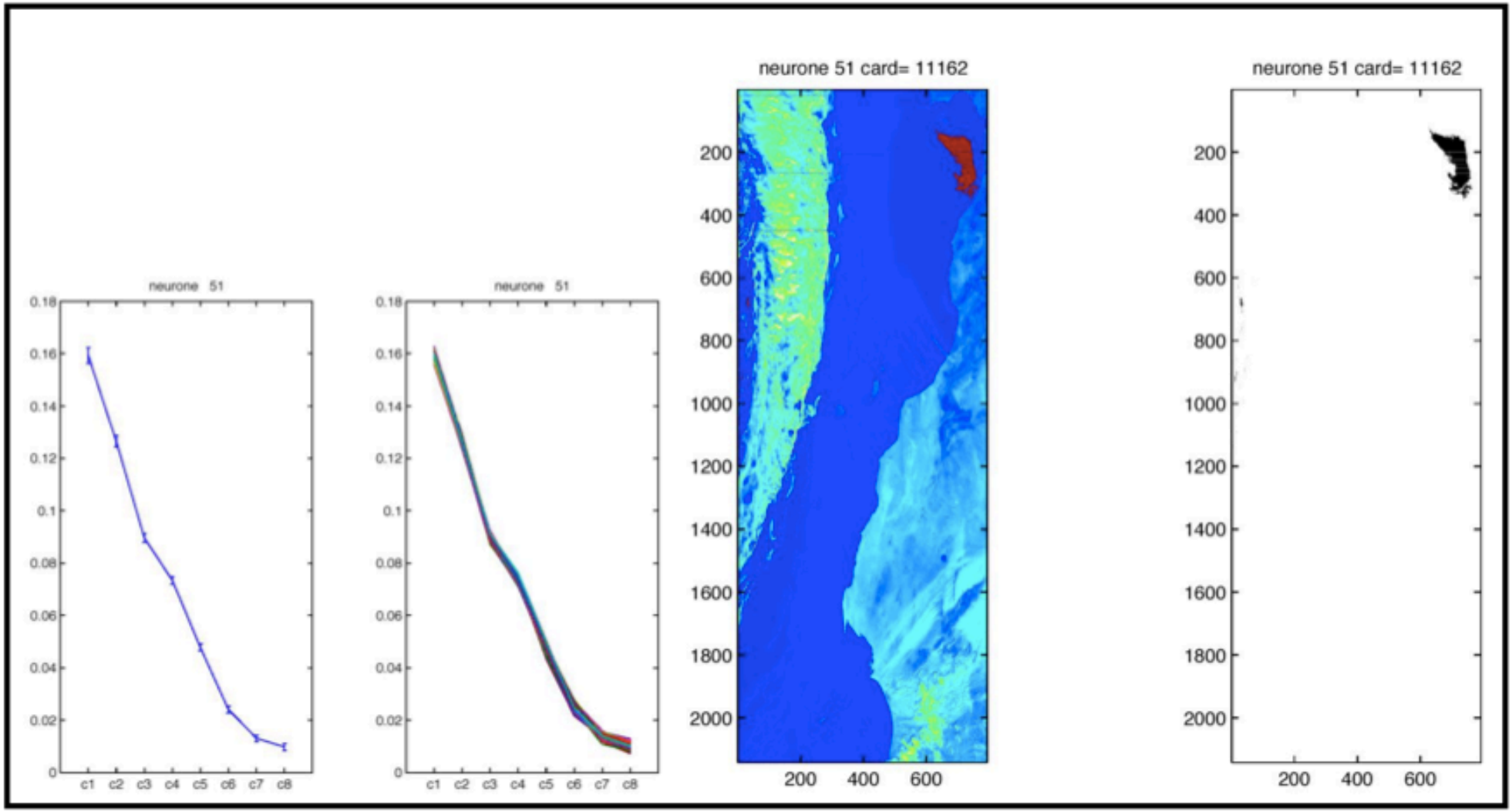
Comparaison de la classification SOM+CAH en deux classes avec le produit distribué par SeaWifs



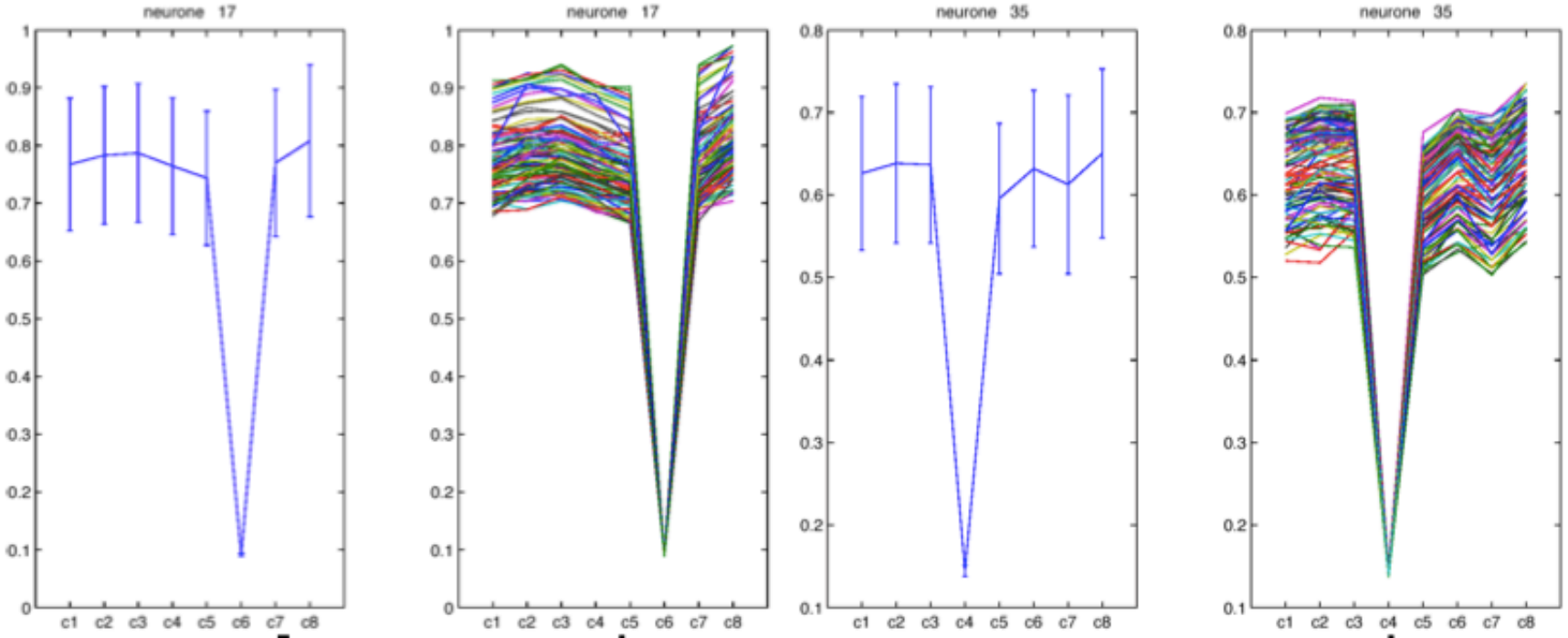
	PRSOM + CAH	
	Nuages	Mer visible
Nuages SeaWiFS	0.91	0.09

9% des pixels ne sont pas écartés par SOM et peuvent être analysés en terme d 'aérosols

Visualisation d 'un neurone, des spectres associés
Localisation du sous-ensemble de pixels sur l 'image



Visualisation de la partition : détection des données aberrantes

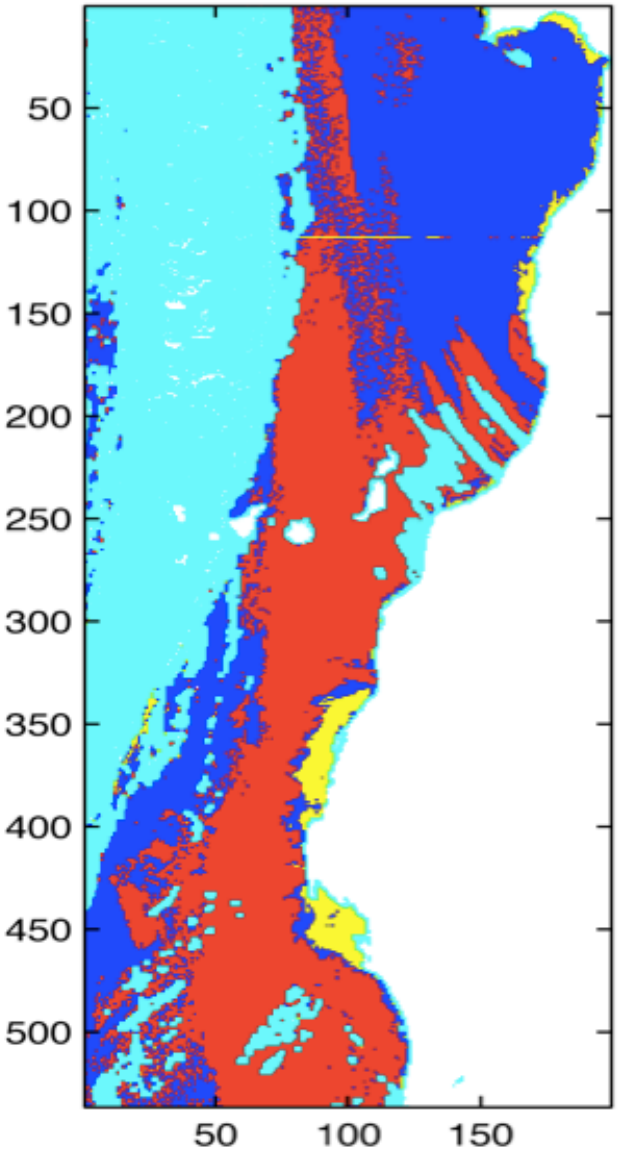


Spectre et variance des neurones 17 et 35

Spectres des pixels de l 'image affectés au neurone 17, 35

Tous les pixels pour lesquels un canal n 'a pas fonctionné sont affectés au même neurone

Image GAC étiquetée à l'aide d'un autre algorithme (Merisse)



Aérosol désertique: bleu foncé

Aérosol marin: rouge

Nuages: bleu clair

Eaux du cas2: jaune

La nature des aérosol ne pouvant pas être observée, la comparaison avec d 'autres méthodes permet de juger de la cohérence de la classification obtenue.

Les types proposés par l 'algorithme Merisse sont utilisés pour étiqueter les neurones

Matrice de confusion comparison entree la classification de la carte et celle de l'expert

		PRSOM + vote majoritaire			
		Classe1	**Classe2**	**Classe3**	**Classe4**
Expert	Aérosols marins	0.8	0.04	0	0.16
	nuages	0.03	0.91	0.01	0.05
	Cas 2	0.03	0.22	0.71	0.03
	Aérosols désertiques	0.1	0.04	0	0.86