

Redes Auto-Organizadas

- ✓ Redes neuronales que se organizan por sí mismas y están concebidas para clasificar conjuntos de datos para los que no se conoce a priori ningún tipo de organización.
- ✓ Objetivo: deducir automáticamente la clasificación más natural de los datos.
- ✓ La salida ante un dato concreto de entrada no se contrasta con ninguna referencia.
- ✓ El aprendizaje de una red auto-organizada es **no supervisado**.
 - ✗ Los datos deben tener un grado de redundancia elevado para realizar su clasificación.
 - ✗ La red divide el conjunto de datos en distintos subconjuntos (cluster), cada uno de los cuales agrupa a datos similares, con algún tipo de característica en común (clustering).
 - ✗ El desarrollo de un método de clustering requiere elaborar alguna medida de la semejanza entre los datos.
 - Distancia Euclídea.
 - Correlación, etc.
 - ✗ Cada cluster se representa por un prototipo: Elemento más representativo del cluster.
- ✓ La estructura de la red consiste de una sola capa de neuronas cada una de las cuales representa un cluster de datos.
- ✓ Entrenamiento de la red: Proceso de aprendizaje competitivo.

Clasificación Automática de Conjuntos de Puntos

- ✓ Datos: Vectores n-dimensionales formados con números reales.
 - ✗ X puntos x_i , con $i=1, \dots, M$ del espacio euclídeo n-dimensional.
 - ✗ Los puntos se reúnen en m conjuntos (clusters): X_k con $k=1, \dots, m$
 - ✗ Centroides de cada cluster c_k

$$c_k = \frac{1}{M_k} \sum_{x \in X_k} x$$

siendo M_k el número de puntos del conjunto X_k .

- ✓ La dispersión de cada cluster viene medida por:

$$\sum_{x \in X_k} \|x - c_k\|^2$$

- ✓ El agrupamiento óptimo de los puntos de X en m subconjuntos sería el que minimizase el índice:

$$I = \sum_{k=1}^m \sum_{x \in X_k} \|x - c_k\|^2$$

- ✓ Sin embargo el número de posibilidades de repartir los M elementos de X en m subconjuntos es muy elevado. Problema de minimización intratable.

- ✓ Simplificación que proporcione una solución subóptima.
- ✓ Cada clase (cluster) se representa por un prototipo.
 - ✗ Dados los m puntos prototipos y_1, \dots, y_m del espacio de \mathbb{R}^n , se asocia cada punto de X con alguno de estos prototipos.

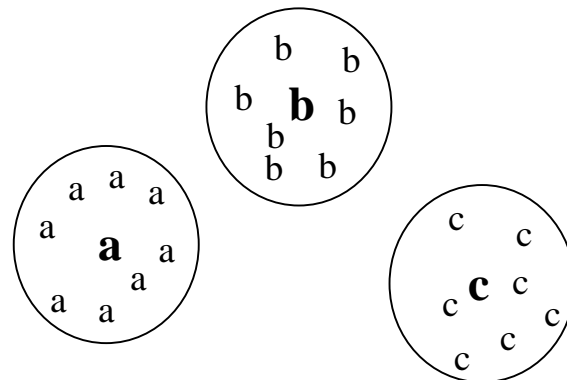
$$\mathbf{j} : X \rightarrow Y$$

- ✗ Fijados los prototipos la calidad de la clasificación viene medida por el índice:

$$I(\mathbf{j}) = \sum_{i=1}^M \|\mathbf{x}_i - \mathbf{j}(\mathbf{x}_i)\|^2$$

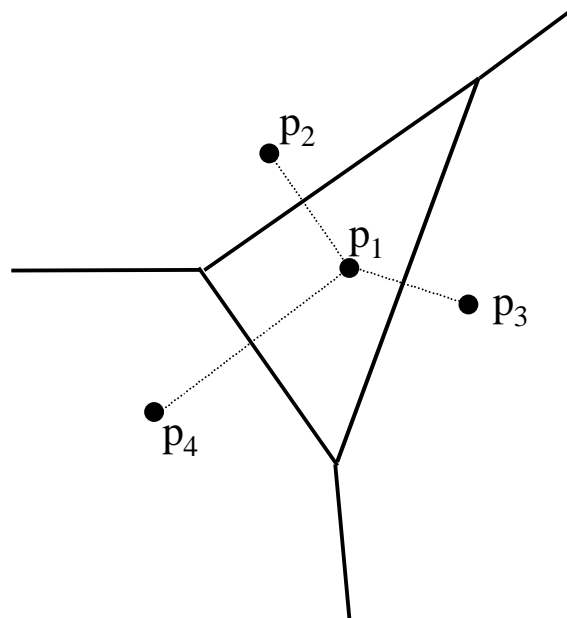
- ✗ La función φ que minimiza este índice es la restricción al conjunto X de la función que asocia a cada punto $\mathbf{x} \in \mathbb{R}^n$ el prototipo más cercano, es decir la que cumple:

$$\|\mathbf{x} - \mathbf{j}(\mathbf{x})\| = \min_{1 \leq k \leq m} \|\mathbf{x} - \mathbf{y}_k\| = d(\mathbf{x}, Y)$$



✓ Teselación de Voronoi

- ✘ El conjunto de puntos de \mathbb{R}^n que están más cerca de y_i , que de los restantes prototipos forma un poliedro (polígono en el plano) que se denomina poliedro de Voronoi. Este poliedro está limitado por los hiperplanos bisectores de los pares de puntos (y_i, y_j) para $i \neq j$.



- ✓ Con esta asociación se puede atribuir a cada conjunto de prototipos un índice de calidad:

$$I = \min_j \sum_{i=1}^M \|x_i - \mathbf{j}(x_i)\|^2$$

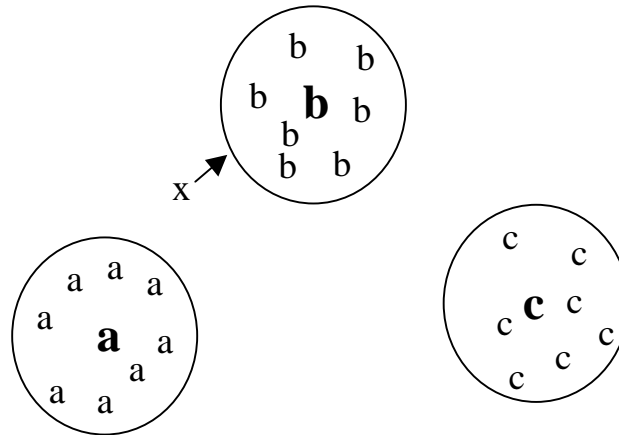
- ✓ Objetivo: elegir los prototipos de forma tal que se minimice el índice presentado anteriormente.
- ✓ Problema: la función a minimizar no es una función diferenciable.
- ✓ Se demuestra que cada uno de los prototipos y_i deben ser, en la situación de óptimo que minimice el anterior índice de calidad, el centroide de los puntos del cluster que representa.
- ✓ Algoritmo (*m-means algorithm*):
 - a) Decidido el número de grupos, se elige un conjunto de partida para los prototipos.
 - b) En cada etapa sucesiva, se procede a la clasificación de los datos de acuerdo con el criterio de asociar cada dato con su prototipo más cercano.
 - c) Efectuada la clasificación anterior, se calcula el centroide de cada cluster. Si el conjunto de centroides coincide con los prototipos de partida de la fase b, el algoritmo finaliza. En caso contrario se sustituyen los prototipos por los centroides y se regresa a la etapa previa.
- ✓ Se puede demostrar que el algoritmo converge hacia un punto en el que el gradiente del índice de calidad es un mínimo local pero no existe la garantía de que este mínimo sea el mínimo global del problema.

Algoritmos Básicos de Aprendizaje Competitivo

- ✓ El algoritmo de clustering previo necesita manejar, en cada etapa del cálculo, todos los datos para poder adaptar los prototipos. Con frecuencia los datos se presentan secuencialmente, y es conveniente ir estimando los prototipos a medida que se van obteniendo los datos.
- ✓ Admitiendo que los datos están ordenados en una sucesión $(x_k)_{k \in \mathbb{N}}$, y que ya se han determinado los cluster a que pertenecen los m primeros términos de la sucesión, para asignar el término $m+1$ se busca el término precedente más cercano:

$$|x_{m+1} - x_v| \leq \min_{1 \leq k \leq m} |x_{m+1} - x_k|$$

- ✓ De esta forma se asocia el nuevo dato al cluster al que pertenezca este vecino más próximo.



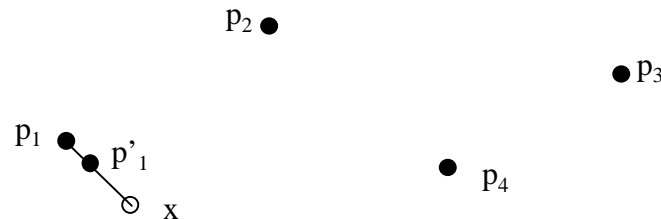
- ✓ Todo para el Ganador: Para cada nuevo dato x , se determina su prototipo más cercano.

$$|x - p_i| \leq \min_{1 \leq k \leq v} |x - p_k|$$

- ✗ Adaptando el prototipo ganador:

$$p'_i = p_i + \eta(x - p_i)$$

siendo η una tasa de aprendizaje predeterminada.



- ✗ Este algoritmo puede sesgar los datos hacia un determinado prototipo si este resulta continuamente dominante, de manera que alguno de los prototipos en situación inicial desventajosa puede no ganar nunca y no adaptarse a los datos. Se puede modular esta tendencia permitiendo que otros prototipos distintos del ganador también se adapten (leaky-learning).

$$p'_i = p_i + \mathbf{h}(k)(x - p_i)$$

- ✗ Existen diferentes opciones de modificar esta tasa a medida que avanza el algoritmo

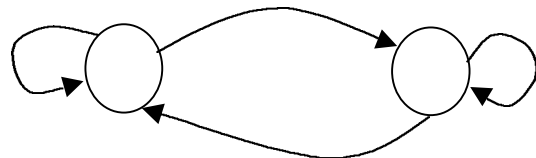
- ✓ En ocasiones es preciso ajustar también de forma adaptativa el número de clusters.
 - ✗ Es preciso introducir un criterio que permita decidir, antes de proceder a adaptar los prototipos existentes, si el nuevo dato pertenece razonablemente a uno de los grupos preexistentes o podría formar parte de otro no detectado hasta el momento.
 - ✗ Se fija un umbral de tolerancia y se exige que la distancia del nuevo dato al prototipo ganador sea menor que este valor umbral.
- ✓ Este aprendizaje competitivo se puede aplicar en situaciones de aprendizaje supervisado.
 - ✗ Si se dispone de una serie de datos de los que se sabe con seguridad el prototipo al que pertenecen, se determina para estos datos el prototipo al que deberían pertenecer p_i , según el algoritmo competitivo planteado. Sabiendo de antemano que estos prototipos son de la clase p_j , se premia el prototipo seleccionado si se acierta, y se penaliza si falla.

$$p'_i = \begin{cases} p_i + \mathbf{h}(x - p_i) & i = j \\ p_i - \mathbf{h}(x - p_i) & i \neq j \end{cases}$$

- ✗ Esta estrategia tiende a disminuir el número de clasificaciones incorrectas.

La Red MAXNET

- ✓ Red que permite buscar el máximo de varios números.

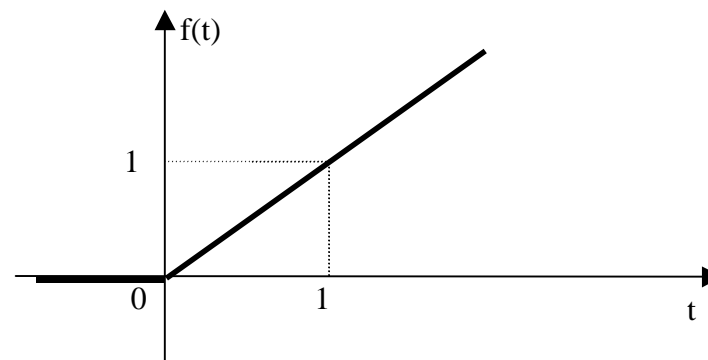


- ✓ La red está compuesta por una sola capa con tantas neuronas como números hayan de compararse. Además la conexión de cada neurona con las restantes es inhibitoria.

$$W = \begin{bmatrix} 1 & -e & -e & \dots & -e \\ -e & 1 & -e & \dots & -e \\ -e & -e & 1 & \dots & -e \\ \dots & \dots & \dots & \dots & \dots \\ -e & -e & -e & \dots & 1 \end{bmatrix}$$

- ✓ La función de transferencia es la función rampa:

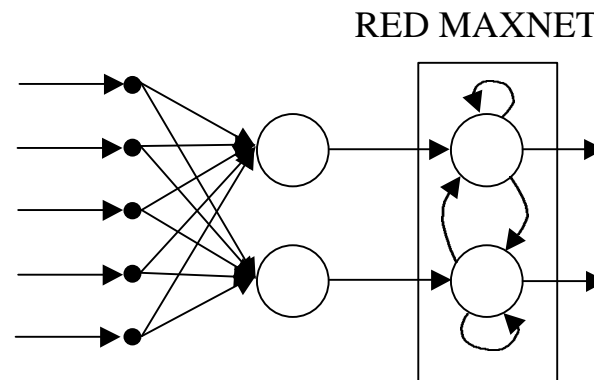
$$f(t) = \max(t, 0) = \begin{cases} t & t > 0 \\ 0 & t \leq 0 \end{cases}$$



- ✓ El vector de entrada se toma como estado inicial de la red. La red evoluciona conforme a las ecuaciones de adaptación:

$$x_i(k+1) = \max\left(0, x_i(k) - e \sum_{j \neq i} x_j(k)\right)$$

- ✓ El estado de la red converge hacia un punto en que la componente que inicialmente era máxima es la única no nula, situación en la que ya no se produce más alteración.
- ✓ La red MAXNET puede utilizarse para implementar una red competitiva basada en el criterio ‘todo para el ganador’:



La Red de Kohonen

- ✓ En una red neuronal con aprendizaje competitivo los prototipos pueden acabar siendo representados de cualquier forma por las neuronas.
- ✓ En la red de Kohonen se supone que las neuronas de la red están relacionadas topológicamente, de manera tal que puede hablarse de un grado de vecindad entre ellas, y se pretende que prototipos semejantes vengán representados por neuronas cercanas.
- ✓ La propiedad de que datos parecidos se representan en neuronas cercanas, robustece la representación de la información. Una perturbación influye mucho menos.
- ✓ La red de Kohonen está formada por una sola capa de neuronas dispuestas en los nudos de una reticulación del plano.
- ✓ Cada neurona de la red tiene asociada un conjunto de neuronas vecinas, que son las que están ubicadas alrededor de la neurona.



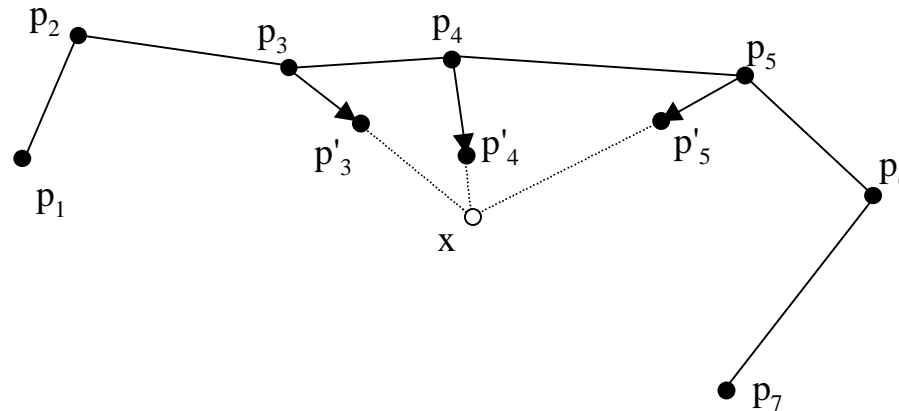
- ✓ En ocasiones las neuronas se distribuyen linealmente.
- ✓ El modo de aprendizaje de una red de Kohonen es prácticamente idéntico al de la red básica competitiva.
 - ✗ Es preciso tener en cuenta las relaciones de vecindad entre las neuronas de la red.
 - ✗ Formalización del proceso:
 - Red: retícula cuadrada de $n \times n$ puntos (conjunto A)
 - N_α : conjunto de índices que representan las neuronas que se consideran vecinas de la neurona α .
 - w_α : Prototipo almacenado en la neurona α .
 - Si en el instante k se presenta un nuevo dato 'x' a la red, se determina en primer lugar la neurona ganadora:

$$\|x - w_{a^*}(k)\| = \min_{a \in A} \|x - w_a(k)\|$$

- Seguidamente se adaptan los pesos de la neurona ganadora y la de sus vecinas.

$$w_b(k+1) = \begin{cases} w_b(k) + h(k)[x - w_b(k)] & \text{si } b \in N_{a^*} \\ w_b(k) & \text{si } b \notin N_{a^*} \end{cases}$$

- La tasa de aprendizaje η depende del instante k y debe decrecer asintóticamente para garantizar la estabilidad del algoritmo.



- ✘ Ponderar de diferente manera la tasa de aprendizaje que se aplica a cada neurona de la vecindad de la ganadora, ajustándola según sea mayor o menor el grado de vecindad.
- ✘ Posibilidad de que los entornos de cada neurona no sean estáticos sino que vayan cambiando a medida que avanza el proceso de aprendizaje.
- ✘ Se consiguen los mejores resultados si se toma al comienzo del proceso entornos muy amplios, que cubran aproximadamente la mitad de la red, y se van contrayendo con el paso del tiempo.

Redes de Resonancia Adaptativa

- ✓ Plasticidad: Capacidad del sistema para adaptarse a un entorno cambiante, en el que constantemente llegue a la red nueva información que debe ser analizada e integrada en la misma.
- ✓ Estabilidad: Necesidad de que la red mantenga la parte esencial de lo previamente asimilado.
- ✓ Las redes de resonancia adaptativa (ART), añaden un subsistema de orientación y vigilancia de la red (estado plástico y elástico).
- ✓ Arquitectura más complicada: Dos capas con realimentación interna.

