El Algoritmo SOM: un ejemplo de visualización informétrica.

Elio Atenogenes.
elio@www.dynamics.unam.edu

Resumen

El desarrollo de métodos de la informática y de la inteligencia computacional ha permitido el aprovechamiento de los ricos almacenes de información que están disponibles en formato digital. Metodologías como ViBlioSOM (Vizualización Bibliométrica con el algoritmo SOM) [22], usan la tecnología de redes neuronales y han probado ser de utilidad para asistir el descubrimiento de conocimiento y el análisis bibliométrico de los grandes volúmenes de información, contenido en las bases de textos de ciencia y tecnología que están disponibles en la actualidad. En este artículo presentamos el algoritmo SOM desde la perspectiva de las redes neuranales de aprendizaje no supervizado. Para ilustrar las capacidades de visualización de esta tecnología, presentamos también una muestra de mapas obtenidos automáticamente, a partir de datos de MedLine con la ayuda de la herramienta de software DataSOMining.

1. Introducción

El estudio de las neuronas y las redes neuronales se ha convertido en un tema ubicuo en el contexto de la Ciencia. Habiendo trascendido el ámbito de la Biología, ha atraído la atención de diversos especialistas: informáticos, físicos, matemáticos, sicólogos, ingenieros de varias especialidades e incluso economistas. Hoy, tanto los Neurocientíficos, como los investigadores del campo de la Ciencia Cognitiva, de la Inteligencia Artificial, de la Robótica, de la Cibernética y particularmente de la Ciencia y la Ingeniería de la Computación, observan atentamente y aprovechan el conocimiento obtenido a partir de la modelación matemática de las redes neuronales. En la perspectiva moderna, el cerebro constituye un *sistema complejo* cuya fisiología puede explicarse en términos de sus propiedades dinámicas.

Las neuronas y las redes neuronales son *sistemas dinámicos* no lineales que pueden ser modelados por ecuaciones diferenciales o mediante la iteración de mapeos (ecuaciones en diferencias). El análisis de la dinámica de los modelos matemáticos está aportando conocimiento útil para entender el funcionamiento del sistema nervioso y el cerebro, así como también para inventar y construir

dispositivos cibernéticos basados en un emergente paradigma computacional, el llamado *conexionismo*, inspirado en la anatomía y fisiología natural.

La modelación de neuronas se basa en sistemas de ecuaciones diferenciales parciales y no lineales. Por ejemplo, la modelación de la fenomenología básica según el modelo Hodgkin y Huxley (ganadores del premio Nobel 1963 de Medicina y Fisiología) requiere de un sistema de cuatro ecuaciones diferenciales parciales. Si se toma en cuenta solamente el fenómeno de excitabilidad y no se considera la propagación espacial del impulso nervioso se logra un modelo simplificado en términos de dos ecuaciones diferenciales ordinarias, conocido como el modelo de FitzHugh-Nagumo [2]. Sin embargo, está demostrado que otros importantes fenómenos, como las ráfagas periódicas de impulsos nerviosos ("bursting phenomenum"), requiere mínimamente tres ecuaciones diferenciales [1]. Para estudiar otros fenómenos se pueden utilizar modelos más simplificados, por ejemplo, el análisis de la respuesta (sincronizada o caótica) de las células nerviosas a una señal periódica, se puede investigar utilizando modelos con una sola ecuación diferencial [6].

Es un hecho sorprendente que con algunos modelos de neurona altamente simplificados se pueden construir redes cuya dinámica guarda un interesante paralelo con algunos comportamientos que se observan también en el escenario biológico [17]. Tal es el caso de las redes de neuronas binarias estudiadas por MacCullock y Pitts [20] a mediados del siglo pasado. Estas células binarias tienen solamente dos posibles estados, uno de exitación y otro de reposo, que pueden ser caracterizados simbólicamente por los números 1 y 0 respectivamente. Usando lógica simbólica, estos autores investigaron las propiedades computacionales que tienen las redes de este tipo de *neuronas formales* y pudieron probar que si estas redes se implementan utilizando almacenes de memoria ilimitada, resultan equivalentes a la clase de máquinas que Alan M. Turing demostró, en 1937, que son computacionalmente universales [24].

Los modelos de neuronas binarias se siguen utilizando exitosamente en la actualidad y una amplia variedad de redes de neuronas artificiales de importancia tecnológica, constituyen sistemas dinámicos discretos determinados por las iteraciones de un mapeo que actua sobre el espacio de estados de la red neuronal.

2. Redes de Procesadores Neuronales

Las *redes neuronales artificiales (RNA),* son artefactos (modelos) matemáticos que pueden simularse computacionalmente via software e implementados en hardware como procesadores neuronales computacionales. La implementación en hardware proporciona una gran ventaja desde el punto de vista del desempeño de la red y permite el procesamiento de grandes volúmenes de datos a gran velocidad.

El modelo matemático de la red puede considerarse constituído por dos elementos básicos:

1. Un modelo de procesador neuronal individual, que gobierna la dinámica

de cada uno de los elementos de la red y

2. Una arquitectura que determina el mapa de interconexiones del conjunto de procesadores, la cuál puede ser representarse como una gráfica dirigida.

El funcionamiento de la red neuronal está gobernado por un conjunto de vectores de pesos, cada elemento de este conjunto corresponde a una neurona. Las componentes de estos vectores juegan el papel de las sinapsis que se establecen entre las neuronas biológicas. El *proceso de entrenamiento* consiste en establecer los valores de los vectores de pesos de manera que el funcionamiento de la red sea óptimo.

El diseño de este objeto cibernético (inspirado en la arquitectura biológica), trasciende el ámbito del procesamiento serial, típico de la computadora digital tradicional y proporciona la capacidad que se le atribuye al *procesamiento paralelo y distribuído.* La literatura se refiere a este nuevo paradigma para la solución de problemas como la teoria conexionista. Estas nuevas *computadoras*, en lugar de programarse de acuerdo al paradigma clásico de Von Neumann, se entrenan por medio de un procedimiento de reforzamiento, como lo hacen las neuronas del sistema nervioso.

2.1. Aprendizaje no Supervisado

Sorprendentemente, el entrenamiento de una red, puede llevarse a cabo de una manera no supervisada. Este esquema de aprendizaje es muy útil para la solución de problemas en los cuales lo que se pretende es, fundamentalmente, la búsqueda e identificación de estructuras, algún tipo de organización o jerarquías de los datos, sin contar con alguna forma de conocimiento a priori útil para ejercer un entrenamiento**.**

En este caso las redes parten de una forma de *medir la similitud* entre los datos de entrada (para ver distintas medidas de similitud consultar [10]) y partiendo de dicha medida son capaces de encontrar de manera automática, patrones de similitud dentro del conjunto de datos de entrenamiento [4], agrupando a los elementos de este conjunto en conglomerados *(clusters),* de manera que datos similares se agrupen dentro del mismo conglomerado y datos disímiles se ubiquen en conglomerados ajenos. Estos descubrimientos pueden realizarse sin ningún tipo de retroalimentación con el medio externo y sin la utilización de información *a priori,* como se ha dicho anteriormente.

Dentro del contexto de los procesos cognitivos del cerebro, la forma de situar al *aprendizaje no supervisado* es considerándolo semejante a los procesos inconscientes, en los cuales ciertas neuronas del cerebro aprenden a responder a un conjunto específico y recurrente de estímulos provenientes del medio externo, es de esta manera como se construyen los llamados "mapas sensoriales" en el cerebro.

Varias áreas del cerebro, especialmente la corteza cerebral, están organizadas de acuerdo a distintas modalidades sensitivas [17]: hay áreas que se especializan en algunas tareas específicas (ver Figura 1), ejemplos de estas tareas son: el control del habla y análisis de señales sensoriales (visual, auditivo, somatosensorial, etc.) . Las distintas regiones del mapa sensorial aprenden a reconocer

estímulos específicos del medio ambiente. Como consecuencia, la información que cada cúmulo de neuronas reconoce, se ubica dentro de cierta categoría entre los estímulos que se reciben del exterior.

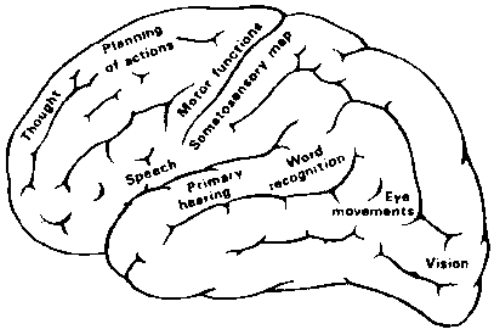

Figura 1: Áreas Cerebrales.

La manifestación más clara del aprendizaje no supervisado de las *redes neuronales* en el sentido fisiológico es que "el aprendizaje puede suceder únicamente cuando hay redundancia en la presentación de los datos" [3]. "Sin una retroalimentación con el exterior sólo la redundancia puede proveer de información útil acerca de las propiedades del *espacio de entrada*" [21]. En la práctica esta redundancia se obtiene mediante la utilización iterada (reciclaje) de un conjunto de datos dentro del conjunto de entrenamiento [17].

El *proceso de aprendizaje* en una red neuronal se entiende como la aplicación iterada de un algoritmo de *actualización* de los *vectores de pesos* W, con la finalidad de que la red neuronal mejore las respuestas que emite al procesar el conjunto de datos de entrada. Dicho conjunto es de la forma $X = \{x(t) \mid t \in T\}$ donde T es un conjunto linealmente ordenado que puede ser continuo (intervalo de números reales) o discreto (subconjunto de números enteros consecutivos). Al conjunto X se le denomina *conjunto de entrenamiento* y al espacio $\mathbb{X}$ de todos los posibles valores de los datos de entrada se le denomina *espacio de entrada.* Normalmene en el espacio de entrada se consideran datos multidimensionales dentro de un espacio métrico $(\mathbb{X}, \|\|)$.

El proceso de aprendizaje de una red neuronal produce una dinámica en los vectores de referencia, ya que los vectores de pesos cambian en cada tiempo de iteración t, en función del dato de entrada $x(t)$. En términos de sistemas dinámicos, lo que se busca es precisamente alcanzar un equilibrio en el sistema o encontrar una solución estable. A la manera en la cual se actualizan los vectores de pesos, se le conoce como *regla de aprendizaje.*

$$W(t+1) = F(W(t), x(t))$$

Una de las formas de aprendizaje no supervisado, es el de las redes de *entrenamiento competitivo.* En estas redes, las neuronas reciben de manera idéntica la

información de entrada sobre la cual compiten (ver Figura 2). Dicha competencia consiste en determinar cual de las neuronas es la que mejor representa a un estímulo de entrada dado. Como resultado de esta competencia solo una neurona es activada en cada momento [17]. Estas competencias determinan un sistema

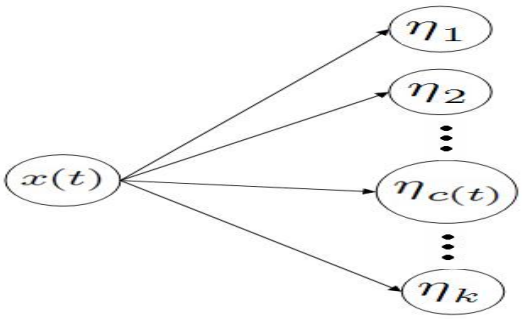

Figura 2: La neurona ganadora de un dato en x es η_c y la neurona η_j es una de las unidades vecinas.

dinámico discreto, donde en cada tiempo de iteración t se determina la neurona ganadora $c(t)$. Una forma bastante común de establecer esta competencia es elegir como neurona ganadora aquella cuyo vector de pesos $w_{c(t)}(t) \in W(t)$, en este caso vector de referencia, es más parecido al dato de entrada $x(t)$, es decir, $c(t)$ queda determinado de manera que:

$$\left\|x(t) - w_{c(t)}(t)\right\| = \min_{i=1}^{k} \{\|x(t) - w_i(t)\|\}. \tag{1}$$

"El proceso de entrenamiento competitivo de una *red neuronal* es estable, si después de un número finito de iteraciones, ningún patrón en el conjunto de aprendizaje cambia de representante"[21]. De manera que en el caso del aprendizaje competitivo, la regla de aprendizaje también dependerá de la forma de determinar la neurona gandora, es decir:

$$W(t+1) = F(W(t), x(t), c(t)).$$

Una forma de lograr la estabilidad es forzando a un parámetro α denominado *factor de aprendizaje* a decrecer y eventualmente converger a cero. De esta manera la red dejará de aprender y por lo tanto se mantendrá estable. Sin embargo, este *congelamiento artificial* del aprendizaje ocasiona que se pierda la plasticidad de la red, es decir la habilidad de adaptarse a nuevos datos. El dilema entre forzar la estabilidad y mantener la plasticidad durante el proceso de entrenamiento en una red neuronal es conocido como: "dilema de estabilidad-plasticidad"de Groosberg. En términos de la modelación matemática, este congelamiento implica la incorporación de componentes autónomas dentro de la regla de aprendizaje, de manera que:

$$W(t+1) = F(W(t), x(t), c(t), t).$$

Una de las ventajas más significativas en las aplicaciones, es que generalmente los modelos de redes neuronales basados en aprendizaje competitivo tienen arquitecturas muy simples y cuentan con algoritmos de entrenamiento más rápidos que las demás redes neuronales.

La más utilizada de las *redes neuronales* no supervisadas competitivas, es la propuesta por Kohonen: Mapas Auto-organizantes (SOM por sus siglas en inglés). El resultado del *aprendizaje competitivo* en el caso del SOM es una partición del conjunto de datos de entrada inducida por la forma en la que se reparten los datos en sus respectivas neuronas ganadoras (de acuerdo a 1). Ésta partición se realiza de manera que datos similares son agrupados por la red y representados por una sola neurona. Dicha neurona es la unidad ganadora para cada uno de los datos asociados, durante la última iteración en el proceso de entrenamiento. Por lo tanto, la agrupación de los datos se realiza de manera automática, basándose en la similitud entre los datos y en la distribución de las respectivas neuronas ganadoras, localizadas a lo largo y ancho de una retícula bidimensional.

3. El SOM

El SOM (Self-Organizing Map) es un algoritmo útil para llevar a cabo la autoorganización y visualización de grandes conjuntos de datos multidimensionales de manera eficiente. Este algoritmo tiene como resultado una función

$$\varphi : \mathbb{X} \rightarrow \mathcal{N}$$

del espacio de entrada a una *red plana de neuronas*. De esta manera se define una proyección del conjunto de datos multidimensionales $\mathbb{X}$ a un espacio perceptible definido por un arreglo bidimencional del conjunto de neuronas $\mathcal{N} = \{\eta_1, ..., \eta_k\}$.

La principal característica de esta función es la denominada *preservación de la topología*, la cual establece que si $x, y \in \mathbb{X}$ son cercanos entonces $\varphi(x)$ y $\varphi(y)$ serán neuronas cercanas.

La visualización del conjunto de datos permite que las relaciones de similitud que se presentan entre los datos en el espacio multidimensional puedan ser observadas en un despliegue bidimensional denominado "mapa".

El SOM fue presentado en 1982 por T. Kohonen [16], desde entonces se han producido miles de artículos de investigación (una gran lista de artículos se puede cosultar en [18]) y ha sido aplicado en una amplia variedad de campos de investigación [17].

La principal razón de la aceptación que ha tenido el SOM es su capacidad de presentar, de manera automática, un mapa en el cual se puede observar una descripción intuitiva de la similitud entre los datos; el despliegue bidimensional tiene la propiedad de exhibir la información contenida en los datos de manera ordenada y resaltando las relaciones de similitud. A continuación se exponen algunos conceptos generales relativos a la naturaleza y utilidad del algoritmo SOM.

3.1. Arquitectura

Las neuronas $\mathcal{N}$ interactuan entre ellas por medio de relaciones laterales que se activan durante la actualización de los pesos. Esta relación es más fuerte cuando la distancia física entre dos neuronas es pequeña (ver figura 3). Es decir, entre más cerca están dos neuronas, mayor es la interacción (intercambio de información) entre ellas.

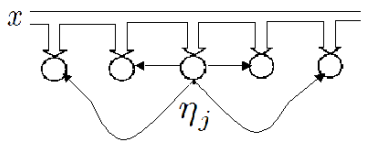

Figura 3: Representación de una neurona η_j y sus conexiones con la entrada x y las neuronas vecinas.

La arquitectura de la red es una retícula con una configuración rectangular o hexagonal. La localización de cada neurona sobre la retícula está representada por su *vector de localización* $r_i = (p_i, q_i)$. En la figura 4 se muestran las configuraciones o tipo de retícula más usados con los correspondientes $r_i = (p_i, q_i)$ en cada nodo. Estas configuraciones tienen la ventaja de que la métrica determinada por su adyacencia (como gráficas) corresponde a su métrica dada en términos de sus distancias en el plano. Cabe señalar que la configuración hexagonal es más conveniente para efectos de visualización.

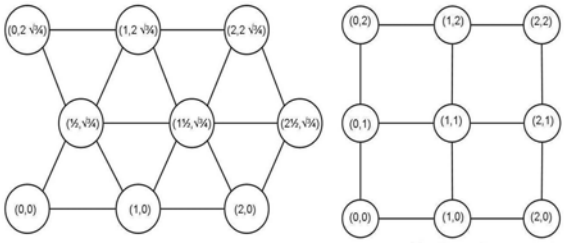

Figura 4: Configuraciones hexagonal y rectangular en la retícula del SOM.

En el algoritmo SOM básico, la configuración de los nodos (hexagonal o rectangular) y el número k de neuronas se fijan desde el principo. Normalmente se definen las distancias entre las unidades del mapa de acuerdo a la distancia *Euclidiana* entre los vectores de localización, sin embargo, en ocasiones es más

práctico usar otras funciones de distancia, como la métrica Manhattan o la Mahalanobits. Esto dependerá del tipo de vectores en el conjunto de entrenamiento.

3.2. Entrenamiento

Durante el proceso de entrenamiento del SOM , se utiliza un conjunto finito de datos $X = \{x_0, ..., x_{m-1}\} \subset \mathbb{R}^n$. Para cada tiempo t, se puede determinar un dato de entrada $x(t)$ para la red neuronal, reciclando el conujunto X. De manera que

$$x(t) = x_{t \bmod m}. \tag{2}$$

Es decir, Para valores de t superiores a m, se reciclan los elementos del conjunto X manteniendo el orden de la primera presentación.

El algoritmo de entrenamiento es el siguiente:

1. Se difine la condición inicial de los vectores de referencia $W(0)$ de manera alealoria y se presenta el dato $x(0)$.

2. Para la presentación del dato $x(t)$ se determina la neurona ganadora $\eta_{c(t)}$ de acuerdo a (1).

3. Para toda $j \in \{1, ..., k\}$ se actualiza al vector de referencia $w_j(t)$ de acuerdo a la siguiente regla de aprendizaje:

$$w_i(t+1) = w_i(t) + \alpha(t) h_{(c,i)}(t) \left[x(t) - w_i(t)\right] \tag{3}$$

4. Se presenta el dato $x(t+1)$ y se repíte el cíclo desde el paso 2.

Este proceso se repíte hasta un número determinado de iteraciones. En caso de que el índice $c(t)$ no esté bien definido; es decir, cuando para un dato $x(t)$ existan dos $\eta_e, \eta_d \in \mathcal{N}$ tal que

$$\|x(t) - w_e\| = \min_{i=1}^{k} \{\|x - w_i\|\} = \|x(t) - w_d\|,$$

la selección de un único $c(t)$ debe hacerse de manera aleatoria.

Cada vez que se determina una neurona ganadora $\eta_{c(t)}$, la idea clave en el algoritmo de aprendizaje es que aquellas neuronas que se encuentran dentro de una vecindad de $\eta_{c(t)}$ en el arreglo bidimensional también aprenderán de la entrada $x(t)$. Para determinar la magnitud del aprendizaje de

una neurona η_i en términos de la distancia con la neurona ganadora $\eta_{c(t)}$ se define la denominada *función vecindad* que es de la forma

$$h_{(c,i)}(t) = h(\left\|r_{c(t)} - r_i\right\|, t) \in [0, 1]. \tag{4}$$

Independientemente de cual sea la forma explícita de la función (4), debe ser tal que $h_{(c,c)}(t) = 1$ para todo t; además para cada t fijo, $h_{(c,i)}(t)$ debe ser decreciente en función de $\left\|r_{c(t)} - r_i\right\|$ y cumplir con $h_{(c,i)}(t) \to 0$ cuando $\left\|r_{c(t)} - r_i\right\|$ se incrementa.

Una de las definiciones más simples que se encuentran de la función vecindad es la siguiente:

$$\begin{aligned} h_{(c,i)}(t) &= 1 \text{ si } i \in N_c(t) \\ h_{(c,i)}(t) &= 0 \text{ si } i \notin N_c(t) \end{aligned} \tag{5}$$

en este caso $N_{c(t)}$ es una vecindad de $\eta_{c(t)}$ sobre la retícula que se define de la siguiente manera:

$$N_c(t) = \left\{ i \in \mathbb{N} \mid \left\| r_{c(t)} - r_i \right\| \leq \rho(t) \right\} \tag{6}$$

donde $\rho(t)$ es el radio de la vecindad en el tiempo t.

Otra forma común de la función vecindad está dada en términos de la función Gausiana:

$$h_{(c,i)}(t) = \exp(\frac{\|r_c - r_i\|^2}{2\rho^2(t)}), \tag{7}$$

en este caso $\rho(t)$ corresponde al ancho promedio de $N_c(t)$.

Para efectos de la convergencia del algoritmo, la variación del radio a través del tiempo debe cumplir que $t_i \leq t_j \Longrightarrow \rho(t_i) \geq \rho(t_j)$ y además $\rho(t) \to 0$ cuando $t \to \infty$. Se recomienda que $\rho(1)$ sea más grande que la mitad del diámetro de la red [17].

La función $\alpha(t)$ es el *factor de aprendizaje*, y en este caso cumple con la condición $0 < \alpha(t) < 1$ y es no creciente de manera que $\alpha(t) \longrightarrow 0$ cuando $t \longrightarrow \infty$.

Tanto $\rho(t)$ como $\alpha(t)$ son componentes autónomas de la regla de aprendizaje y su principal objetivo es garanizar la convergencia del algoritmo apartir de producir cambios cada vez más locales (centrados en la neurona ganadora) y de menor magnitud.

Durante la evolución del proceso de entrenamiento, los vectores de pesos son modificados de manera que cada uno de estos se vuelva representante de una porción en el espacio de entrada. Durante esta evolución es común observar dos etapas: *ordenamiento global y refinamiento.*

3.3. Etapas del entrenamiento

- **Ordenamiento Global:** Esta etapa consiste en establecer los pesos de cada una de las neuronas, para que éstas sean capaces de identificar cierto subconjunto característico dentro del conjunto de datos X y para que las relaciones de cercanía entre las distintas neuronas del mapa, reflejen la similitud entre los datos. Es importante señalar que la selección óptima de estas funciones y sus parámetros, solo pueden ser determinadas experimentalmente; ya que no existe algún resultado analítico que garantice dicha selección óptima.

- **Refinamiento:** Después de la fase de ordenamiento, los valores de $\alpha(t)$ deben ser pequeños y decrecer lineal o exponencialmente. La precisión final del mapa dependerá del número de pasos en esta etapa final de la convergencia, la cual debe ser razonablemente larga. Nótese que el algoritmo es computacionalmente ligero y que el conjunto X puede ser reciclado para lograr tantos pasos como sea necesario [17].

Una vez concluido el proceso de entrenamiento, el SOM define una *regresión no-lineal* que proyecta un conjunto de datos de dimensión alta, en un conjunto de vectores de referencia de la misma dimensión, pero que corresponden a neuronas sobre una malla bidimensional, en donde se pueden observar las relaciones de similitud y la distribución de los datos. De esta manera es posible construir una *representación bidimensional de un conjunto de datos multidimensional.*

En la figura 5 se pueden observar los vectores de pesos de un SOM bidimencional de 100 neuronas con una condición inicial de los vectores de referencia aleatoria y entrenado a partir de un conjunto de 2000 puntos distribuidos uniformemente en la superficie de dos toros encadenados, el cual es reciclado en 10 ocasiones. El algoritmo implementado utiliza como función vecindad a la dada en 7, estas imagenes fueron realizadas por el sistema de software LabSOM que se está desarrollando en el *Laboratorio de Dinámica no-Lineal*, de la *Facultad de Ciencias* de la *U.N.A.M.*

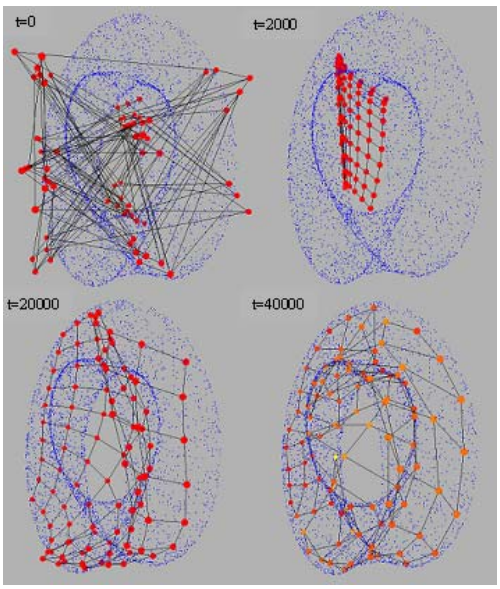

Figura 5: Visualización de los vectores de referencia durante distintas etapas del entrenamiento.

4. Aplicación del SOM en el Análisis Cienciométrico

Una de las aplicaciones en las que el SOM ha tenido mayor impacto, es en el *Análisis Inteligente de Datos*. En este campo emergente, se combinan los métodos de *aprendizaje de máquina* –redes neuronales, algoritmos genéticos, autómatas, sistemas expertos, etc.– con las técnicas tradicionales de análisis de datos –estadística multivariada, análisis exploratorio, etc.–para producir poderosos métodos que nos auxilian en la tarea de analizar efectiva y eficientemente grandes cantidades de datos multidimensionales. Estas herramientas son incorporadas en el desarrollo de sistemas de software para el *Descubrimiento de Conocimiento en Bases de Datos* (KDD por sus siglas en ingles [11]). Muchos de estos sistemas implementan al SOM en tareas como la exploración de datos [15], aglomeración (clustering) de los datos [8], *minería de datos y texto* [19] y *visualización de información* [12]. Algunos de estos sistemas de software están especialmente diseñados para aplicaciones tales como la visualización de información en *finanzas* [13] o el análisis del comportamiento del consumidor [9].

Un ejemplo de desarrollo de estos sistemas es el software DataSOMining desarrollado en el Laboratorio de Dinámica no Lineal de la Facultad de Ciencias de la U.N.A.M.. Este sistema ejecuta una metodología que hemos denominado ViBlioSOM [22], con la finalidad de descubrir y representar conocimiento a partir del procesamiento de grandes bases de datos bibliográficos y de patentes.

La metodología ViBlioSOM (Visualización Bibliométrica con la red neuronal SOM), está diseñada para aplicar el proceso KDD al análisis bibliométrico, mediante la aplicación secuencial de varios sistemas de software propietarios. El sistema DataSOMining integra las funciones de estos sistemas de software propietario, en un solo sistema modular.

En esta trabajo se presenta un ejemplo de aplicación del SOM en KDD, dentro del campo emergente denominado *cienciometría* (análisis bibliométrico en textos científicos). El objetivo, en este caso, es el descubrimiento de información útil para la investigación científica y el desarrollo tecnológico [23]. La aplicación del SOM a la cienciometría, en particular cuando se realiza sobre bases de datos de información biomédica, hace que esta herramienta se considere como parte de las técnicas de la *bioinformática* [7], en esta misma linea se encuentran las aplicaciones para la representación visual de expresiones genómicas [14].

4.1. Las matemáticas como herramienta de investigación en la biología

Hasta hace unas décadas, la investigación biológica había sido fundamentalmente experimental y, aparte de la estadística, pocas técnicas matemáticas habían encontrado un nicho de aplicación. Desde el año de 1957 hasta el primer semestre del año 2007, la Biblioteca Nacional de Medicina (NLM) de Estados Unidos ha indizado alrededor de 16 millones de documentos relacionados con

temas de Biomedicina (medicina, enfermería, odontología, oncología, medicina veterinaria, salud pública, ciencias preclínicas y otras áreas de las ciencias de la vida).

La NLM utiliza la ontología MeSH (*Medical Subjet Heading*) para indizar la literatura biomédica. Esta ontología organiza los términos descriptores con los que se indizan las publicaciones, en 15 categorías principales y cada categoría se ramifica en series de subcategorías cada vez más concretas o específicas. Esta ontología es revisada anualmente por un equipo de profesionales, con la finalidad de incorporar temas emergentes.

La Categoría de Ciencias Físicas (*Physical Sciences Category*) es una de las 15 categorías que integran el MeSH. En esta categoría se encuentran subcategorías relacionadas con temas físicos, químicos, matemáticos, biológicos, etc. En la figura 6 se observa la ramificación de la subcategoría *Mathematics*.

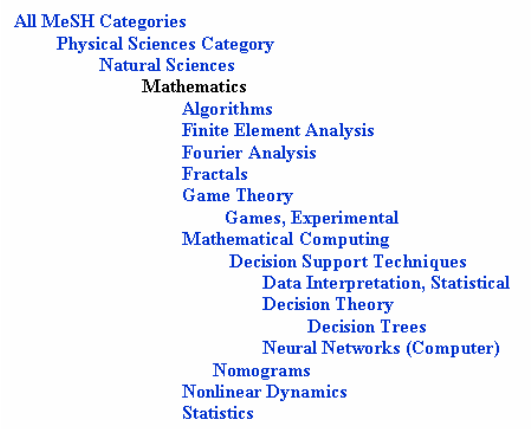

Figura 6: Ramificación de la subcategoría Mathematics.

Actualmente existen 1,210,583 documentos registrados en MedLine que tienen como uno de sus descriptores algún término perteneciente a la categoría Mathematics. Lo primero que se observa es que la gran mayoría (1,101,897) de estos documentos están indizados con algún tema dentro de la subcategoría Statistics.

Sin embargo, también se observa que en las últimas décadas se ha incrementado el uso de otras ramas de las matemáticas.

A partir de la década de los 70 se han incorporado a la investigación biomédica términos como *Mathematical Computing*, *Algorithms*, *Finite Element Analisys*, *Fractal*, *Nonlinear Dynamics* y *Neural Networks*, revelando la importancia que han ido adquiriendo las matemáticas en general, así como la tendencia a la utilización de herramienta en temas como por ejemplo Algorithms y Nonlinear Dynamics. En este estudio nos interesa observar como se han ido incorporando estos temas matemáticos, en la investigación biológica. En cosecuencia consideraremos aquellos temas matemáticos no estadísticos.

En este escenario se plantean de manera natural preguntas como ¿En qué áreas de la biología estos métodos han ido encontrando un espacio de aplicación?, ¿Qué tienen en común y qué relación guardan entre sí los temas biológicos que son susceptibles de un tratamiento matemático similar?.

Nuestra intención en la aplicación que aquí exponemos, es presentar a la red neuronal SOM como una herramienta útil para dar respuesta a este tipo de preguntas. Una de las principales características de los mapas construidos es, que apartir de un análisis cualitativo de los mismos, se puede descubrir información útil que no es evidente *a priori*.

Para poder utilizar el modelo SOM expuesto anteriormente, es necesario transformar los datos en vectores. Ahora bien, en el ejemplo que nos ocupa, los datos son ficheros bibliográficos de PubMed. Estos ficheros son archivos de texto que contienen información capturada en campos como: autor, título, resumen, etc.. En particular, está el campo de los términos MeSH que los indizadores asignaron al fichero en cuestión.

Consideremos al conjunto $M = \{mat_1, ..., mat_{13}\}$ de aquellos términos matemáticos (no estadísticso) presentes en el MeSH, $B = \{b_1, ..., b_{1668}\}$ al conjunto de los términos biológicos y A al conjunto de los ficheros con algún término en M.

Partiendo del hecho de que MedLine es una base de datos especializada en temas biomédicos, podemos suponer que si un tema biológico b aparece como descriptor en algún fichero de A entonces el o los temas matemáticos que aparecen en ese fichero son herramientas de investigación en el tema biológico b. Dado $a \in A$ y kw un término MeSH diremos que $kw \in a$ si kw aparece como descriptor en el fichero a.

En esta aplicación del SOM, cada elemento del conjunto de entrenamiento corresponde a la vectorización de un tema biológico. Cada entreda de un vector respresenta la relación del tema biológico con cada tema matemático. Esta relación se mide a partir de la probabilidad de que aparezca el tema matemático cuando aparece el tema biológico y se considera como espacio de busqueda el conjunto A, i.e. si se considera $b \in B$, entonces su representación vectorial $\bar{b}$ está dada por:

$$\bar{b}_j = \Pr(m_j \in a \mid a \in A, b \in a).$$

En los mapas generados el sistema DataSOMining produjo la auto-organización de los temas biológicos desde la perspectiva de los temas matemáticos. Cada tema biológico quedó ubicado en algún lugar del mapa de acuerdo a una distribución, según la cual dos temas biológicos están más cercanos, cuanto más similar es la forma en que los distintos temas matemáticos son utilizados en ellos.

En este ejemplo para la visualización se utilizaron mapas de componentes y mapas de conglomerados obtenidos mediante el algoritmo de clustering SOM-Ward. Cada tema biológico se representó como un punto en un espacio ($13 -$ *dimensional*), en el que cada dimensión representa un diferente tema matemático. La distribución de puntos se replica en el mapa bidimensional mediante un proceso de proyección (no lineal) que mapea temas biológicos relacionados en puntos cercanos en el mapa.

El mapa de conglomerados es la proyección en la malla de los cúmulos de datos (figura 7). Esta proyección segmenta la malla en distintas regiones que tienen la característica de albergar en su interior temas de biología en los cuales las matemáticas intervienen de una manera semejante. En la cartografía que muestra la figura 1 quedan determinados distintos territorios que representan las distintas clases de similaridad que resultan al hacer este tipo de clasificación.

Figura 7:

Para cada componente se puede producir también un mapa (mapa de componente) que exhibe la distribución de los valores de esa componente en las distintas regiones del mapa (figura 8). Entre más oscuro sea el tono de gris de una región de un mapa de componente, el tema matemático que corresponde a esa componente tiene un valor mayor. Esto significa que la herramienta matemática en cuestión, coocurre con mayor frecuencia con los temas biológicos que están en las regiones más oscuras. El análisis de estos mapas permite también observar, por ejemplo, la relación que existe entre los distintos temas biológicos de acuerdo al uso que ellos hacen de los temas de matemáticas. Por ejemplo, comparando la figura 7 con la figura 8, observamos que las técnicas matemáticas etiquetadas como Dinámica no lineal, se utilizan notablemente en los siguientes temas: Bioquímica, Sistemas Biológicos y Electrofisiología. Por otro lado, analizando el mapa de la componente correspondiente a Redes Neuronales Artificiales, apreciamos su preeminencia en los campos de Neurofisiología, Biotecnología, Microbiología, Biología Molecular y Neurología.

Por otra parte, el análisis comparativo de dos o más mapas de componentes

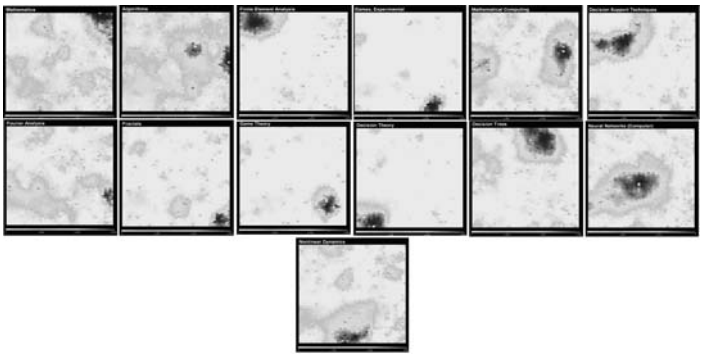

Figura 8: Mapas de Componentes

permite identificar regiones (temas biológicos) en cuyas investigaciones están involucrados dos o más temas matemáticos (identificando la intersección, en los distintos mapas de componentes, de las regiones que tienen un mismo tono de gris). Por ejemplo en los mapas de componentes de Redes Neuronales y Dinámica no Lineal existen regiones comunes en las cuales los dos temas muestran una presencia importante (figura 9).

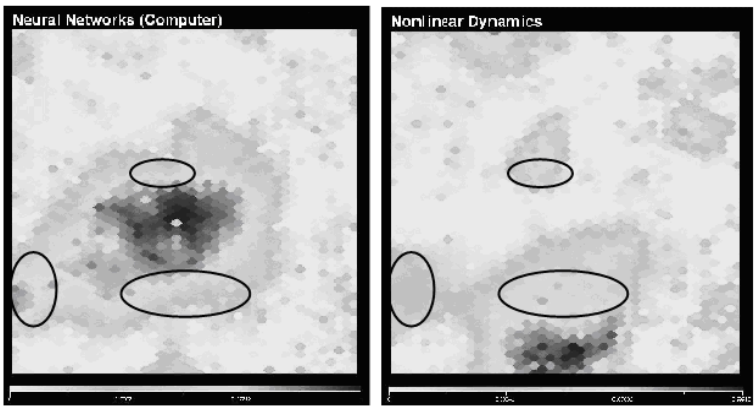

Figura 9: Regiones en donde las componentes Redes Neuronales y Dinámica no Lineal tiene una presencia importante.

Referencias

[1] M. Baer, J. Rinzel, H. Carrillo: *A Three Variable Autonomous Phase Model For Neuronal Parabolic Bursting*, Differential Equations With Aplications

to Biology and Industry, World Scientific, Singapore, 1996. pp. 1-11.

[2] C. Barriga, H. Carrillo, F. Ongay, *El Modelo de FitzHugh-Nagumo para el Potencial Eléctrico de una Neurona*, Aportaciones Matemàticas, Serie de Comunicaciones 32 (2003) pp. 31-49.

[3] Barlow H. B., *Unsupervised learning*, Neural Computation, 1:151-160, 1989.

[4] Bigus J., *Data Mining with neural networks*, Mc GrawHill, USA, 1996.

[5] Boden M.A., "The Philosophy of Artificial Intelligence", Oxford University Press, 1990.

[6] H. Carrillo, M. Mendoza, F. Ongay, *Integrate-and-Fire Neurons and Circle Maps*, WSEAS TRANSACTIONS ON BIOLOGY AND BIOMEDICINE Issue 2, Vol. 1, April 2004, pp. 287-293.

[7] H. Carrillo, M.V. Guzmán, E. Villaseñor, E. Valencia, R. Calero, L. E. Morán y A. Acosta, *Minería de Datos con Redes Neuronales Artificiales: Aplicación en Vacunas Tuberculosis*, Congreso Internacional de la Información. INFO'2004.LA HABANA. Abril del 2004. ISBN 95923404044.

[8] Endo M.,Uendo M.,Tanabe T., *A Clustering Meted Using Hierarchical Self-Organizing Maps*, Journal of VLSI Signal Processing 32, 105-118, 2002.

[9] visitar la pagina www.eudaptics.com para conocer las capacidades de uno de estos sistemas.

[10] Everitt B., *Cluster Analysis*, London : Heinemann educational books, 1974.

[11] Fayyad U., Piatetsky-Shapiro G., Smyth P., *Knowledge Discovery and Data Mining: Towards a Unifying Framework*, Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, Oregon, USA, 1996.

[12] Flexter A., *On the use of Self-organizing Maps for Clustering and Visualization*, The Austrian Institute of Artificial Intelligence, 1999.

[13] Guido J. Deboeck, T. Kohonen, *Visual Explorations in Finance: With Self-Organizing Maps*, Springer Finance, 1998.

[14] Herrero J., Valencia A., Dopazo J., *A hierarchical unsupervised growing neural network for clustering gene espression patterns*, Bioinformatics Vol. 17, 2001.

[15] Kaski S., *Data Exploration Using Self-Organizing Maps*, Ph. D. Thesis, Helsinki University of Technology, Finland, 1997.

[16] Kohonen T. , *Self-organized formation of topologically correct feature maps*, Biological Cybernetics, 43, 1982.

[17] Kohonen T., *Self-Organizing Maps*, 3ra Edición, Springer-Verlag, 2001.

[18] Un listado de algunas de las publicaciones puede ser consultado en: http://www.cis.hut.fi/nnrc/refs/

[19] Lagus, K. *Text Mining with the WEBSOM*, Dissertation for degree of Doctor of Science in Technology, Helsinki University of Technology, 2000.

[20] McCulloch, Warren S., Pitts, *A Logical Calculus of the Ideas Immanent in Nervous Activity*, Bulletin of Mathematical Biophysics 5, pp. 115-133, 1943 (publicado en [5] pp. 22-39).

[21] R. Rojas, *Neural Networks: A Systematic Introduction*, Springer, 1996.

[22] Sotolongo G., Guzmán M.V., Carrillo H, *ViBlioSOM: Visualización de Información Bibliométrica mediante el Mapeo Auto-Organizado*, Revista Española de Documentación Científica, 2002, pp. 477 - 484.

[23] Sotolongo G., Guzmán M. V., Saavedra O., Carrillo H., *Mining Informetrics Data with Self-organizing Maps*, in: M. Davis, C.S. Wilson, (Eds.), "Proceedings of the 8 th International Society for Scientometrics and Informetrics", ISBN:0-7334-18201. Sydney, Australia July 16-20. Sydney: BIRG; 2001: 665-673.

[24] Turing A. M., *On Computable Numbers, with an Application to the Entcheidungsproblem*, Proc. London Mathematical Society 43, 1937.