Facultat de Ciències Econòmiques i Empresarials
Departament de Matemàtica Econòmica, Financera i Actuarial

LA MEMÒRIA ALS MERCATS FINANCERS: UNA ANÀLISI MITJANÇANT XARXES NEURALS ARTIFICIALS

Mª Teresa Sorrosal i Forradellas

2005

Cap. 2. L'APROXIMACIÓ CONNEXIONISTA: UN ENTORN PER A L'ANÀLISI DE LA MEMÒRIA ALS MERCATS FINANCERS

"How is it possible, using a collection of relatively simple elements connected to one another, to implement the basic functions of associative memory?"

Teuvo Kohonen (1989:4)

2.1. Introducció.

En els estudis científics relatius a les capacitats cognitives humanes s'han contemplat dues aproximacions, la simbòlica i la connexionista. En la primera d'elles es tractava d'analitzar les teories relatives a la ment mitjançant el model funcional de l'ordinador. Un computador rep una informació i després de processar-la, ofereix com a resposta unes dades de sortida. L'adjectiu de simbòlica és degut a que en la seqüència *input*-processament-*output*, les dades són representades mitjançant símbols.

D'altra banda, models i teories cognitives també tenen el seu origen en el funcionament del cervell. En aquest cas, són les neurones i les connexions entre elles les que serveixen com a base de coneixement. Els sistemes constituïts per un nombre d'aquestes unitats, connectades entre sí i processant la informació en paral·lel s'ha simulat en programes informàtics als que s'anomenen Xarxes Neurals Artificials (XNA). Nosaltres les proposem per a modelar la capacitat mnemònica. Qualsevol tipologia existent de XNA reté la informació que es subministra al sistema en les unitats que el formen (memòria a curt termini), de la que s'extreu coneixement com a resultat de la generalització d'un conjunt d'exemples (memòria a llarg termini) que es va modificant per tal d'adaptar-se a futures presentacions (aprenentatge).

Aquest capítol té com a finalitat presentar els aspectes conceptuals, metodològics i matemàtics de l'instrumental que ens permetrà analitzar quantitativament i qualitativa els diferents efectes de la memòria en un mercat financer. Serà en els capítols corresponents a cada accepció on detallarem la manera en que les XNA poden ser emprades en la seva anàlisi i on oferirem també els resultats de l'aplicació a un exemple concret. Una de les avantatges del paradigma connexionista és que permet homogeneïtzar la metodologia amb la que tractem els efectes del passat amb influències sobre l'evolució futura d'un mercat.

Tot seguit exposarem breument el concepte i els elements que formen una XNA, intentant donar una visió general del seu funcionament, però posant especial èmfasi en aquelles característiques que les fan apropiades per a la nostra recerca. De la tipologia de XNA únicament detallarem les que han estat utilitzades per a dur a terme les aplicacions empíriques posteriors.

2.2 Nocions bàsiques.

2.2.1. Definicions de XNA.

En el capítol 1 ens hem referit a l'element fonamental per a la memòria de l'ésser humà: la neurona, i en concret, la unió entre dues neurones, la sinapsi. El nostre cervell és una xarxa de connexions entre les 100.000 milions de neurones que conté. Connexions que canvien constantment com a resultat de l'aprenentatge de l'individu.

Les XNA són programes informàtics que reprodueixen aquest esquema. Estan formades per elements simples anomenats *unitats* o *neurones artificials*[1], unides per connexions en les que hi ha associat un pes que recull el grau d'intensitat de la unió i el signe d'aquesta. Mitjançant l'*algorisme d'aprenentatge* es modificarà el valor dels pesos fins a arribar a un resultat òptim per al problema que aborda el sistema.

Podem trobar moltes definicions de XNA, entre les que destaquem les següents:

> *"Una xarxa neural és un processador distribuït constituït per unitats de processament simples i amb estructura paral·lela que té una tendència natural a emmagatzemar*

[1] Per simplificació, ens hi referirem únicament com a neurones.

> *coneixement experimental, fent-lo apte per al seu ús. Se sembla al cervell en dues coses:*
>
> 1. *El coneixement és adquirit per la xarxa del seu entorn mitjançant un procés d'aprenentatge.*
> 2. *Aquest coneixement s'emmagatzema en els pesos sinàptics o connexions entre neurones."* Haykin, S. (1999:2) [Traduït al català]

i

> *"Les xarxes neurals artificials són un sistema adaptatiu que resol problemes mitjançant la imitació de l'arquitectura dels sistemes neurals biològics. Les capes de neurones artificials estan interconnectades mitjançant pesos que representen la força de les connexions sinàptiques entre les neurones."* Cox, E. (1993:27) [Traduït al català]

En ambdues es posen de relleu les característiques principals d'aquest instrument: una estructura formada per elements simples altament interconnectats i la seva inspiració en el sistema neural biològic. La definició que dóna T. Kohonen (1988:251) resumeix de forma concisa ambdues idees. Segons l'autor,

> *"Les xarxes neurals artificials són xarxes d'elements (normalment adaptatius) interconnectats massivament en paral·lel i les seves estructures jeràrquiques que pretenen interactuar amb els objectes del món real de la mateixa manera que ho fa el sistema nerviós biològic".* [Traduït al català]

Malgrat la seva inspiració biològica, no hem d'oblidar que una XNA és un programa informàtic, encara que no tot programa informàtic sigui una XNA. Així, la seva implementació requereix d'una formalització algorísmica susceptible de tractament matemàtic. A la complexitat d'aquest procediment cal afegir-li la pròpia del context al que volem aplicar-les. La incertesa de l'entorn financer ha conduït a les disciplines científiques implicades a cercar instruments més adequats per aproximar-se a aquesta realitat, com ara la teoria de subconjunts borrosos, els algorismes genètics o la teoria de grafos. Des d'aquesta perspectiva, és important destacar la interpretació d'una XNA com un grafo (Gil Aluja, 1995) on els vèrtexs realitzen la funció de les neurones i els estats permanents, mitjançant els corresponents arcs, emmagatzemen la informació que es troba a la memòria.

2.2.2. Elements bàsics d'una XNA.

Alan Turing, al 1936, fou el primer investigador en considerar l'estudi del cervell com un nou camí en el món de la computació. Però foren un neurofisiòleg, Warren McCulloch, i un matemàtic, Walter Pitts, els que desenvoluparen al 1943 una teoria sobre el funcionament de les neurones, elaborant les bases teòriques de la computació neural[2].

La unitat bàsica per a les XNA ja hem esmentat que és la neurona. Aquestes unitats s'agrupen en diferents nivells, anomenats *capes*, amb la característica de que les senyals d'entrada a una mateixa capa provenen del mateix origen i les senyals que emeten es dirigeixen a un destí comú. En funció del nivell al que pertanyen es distingeixen tres tipus de neurones: les d'*entrada*, que reben la informació directament de l'exterior del sistema, les de *sortida*, que envien la senyal de resposta obtinguda per la xarxa, i les *ocultes* que són les que es troben entre les dues anteriors, podent o no existir i sent variable el nombre de capes ocultes en cada xarxa.

Tota neurona, en qualsevol instant, està caracteritzada per un valor numèric denominat *valor o estat d'activació* que coincideix amb els estats de *repòs* i d'*excitació* (de forma absoluta o en algun grau). La pertinença a un o altre depèn del propi estat en el que es troba la neurona i dels mecanismes d'interacció que actuen entre ella i les seves veïnes. La *funció d'activació* és la que determina l'estat d'activació en el període següent.

Existeix una altra funció, la *funció de transferència,* que converteix l'estat actual d'activació en una senyal de sortida. Depenent de la xarxa que es desitgi utilitzar i de l'objectiu per al que es construeixi, s'utilitza un tipus o altre de funció de transferència, sent de les més comuns la funció esglaó, la lineal i la sigmoidal, així com variacions d'aquestes, com ara la tangent sigmoidal[3].

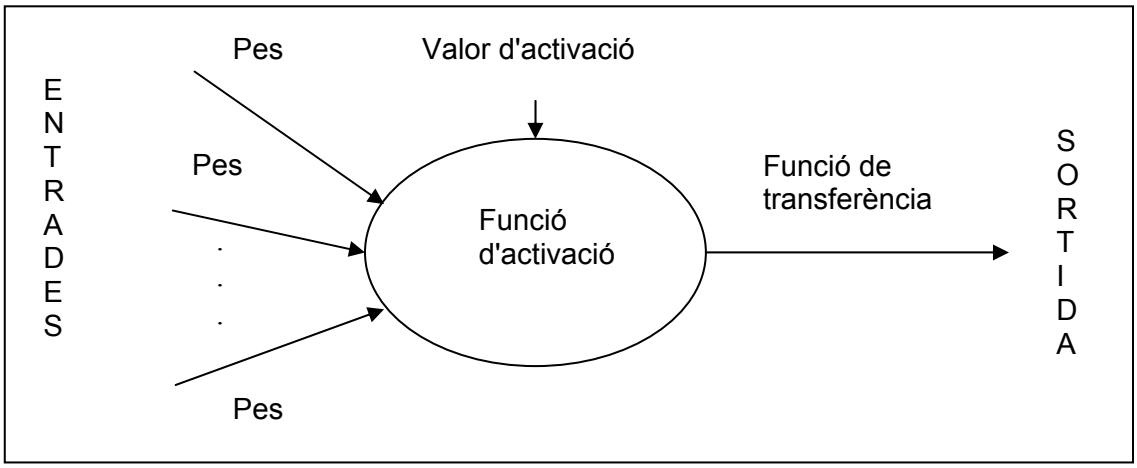
[2] McCulloch, W.S. i W.H. Pitts (1943).

[3] A Torra (2004:60) trobem una taula detallada de funcions de transferència. També se'n proposen a Haykin (1999), Hilera (1995), Refenes (1995) o Freeman (1993). És de destacar el caràcter no lineal de les XNA, degut a la possibilitat d'utilitzar funcions que no tenen aquesta propietat. En ocasions és la naturalesa de la pròpia xarxa la que limita la possibilitat d'escollir la funció de transferència, com en les xarxes amb algorisme d'aprenentatge per propagació de l'error, on és necessari que siguin contínues. En altres ocasions és l'usuari o dissenyador de la xarxa qui opta per una o altra.

La senyal de sortida s'envia a altres unitats de la xarxa, modificant-se segons vagi passant per les sinapsis, mitjançant els *pesos* que es troben associats a la connexió entre dues neurones. L' adquisició de coneixement (aprenentatge) s'assoleix gràcies a la modificació del valor d'aquests pesos, on s'emmagatzema la informació que forma part de la memòria.

L'*entrada neta* que finalment rebrà cada neurona després de tot el procés de transmissió d'informació serà la suma del producte de cada senyal individual que envia una unitat de la XNA pel valor de la sinapsi que connecta ambdues neurones (coneguda com a *regla de propagació*) més un valor anomenat *llindar d'activació*.

Es pot observar la similitud entre el funcionament d'una neurona biològica i una d'artificial comparant el següent gràfic (2.1.) amb el presentat en el capítol anterior (pàgina 23):



Gràfic 2.1. Esquema d'una neurona artificial.

2.2.3. Estructura d'una XNA. Classificacions.

En el disseny d'una xarxa neural artificial, el primer que s'ha de decidir és quina serà l'arquitectura adequada per a la funció que desitgem que desenvolupi. Decidir sobre l'arquitectura, també anomenada *topologia*, és decidir el nombre de capes, el nombre de neurones que formarà cada capa, el grau de connexió entre elles i la naturalesa de les connexions, així com l'algorisme d'aprenentatge que s'utilitzarà en la modificació dels pesos.

En funció del nombre de capes que formen la xarxa es distingeix entre:

- XNA *monocapa*. El disseny d'una xarxa amb una sola capa facilita l'operativa dins del sistema i és capaç de desenvolupar de forma adequada tasques relacionades amb l'autoassociació, reconèixer una informació d'entrada quan aquesta està incompleta o presenta soroll. Les connexions entre les neurones de l'única capa són *laterals*, és a dir, entre neurones d'una mateixa capa, ja siguin excitadores (pesos positius) o inhibidores (pesos negatius) i també poden haver-hi d'*autorrecurrents*, la connexió d'una neurona amb ella mateixa.

- XNA *multicapa*. En cas de que existeixi més d'una capa de neurones, indistintament de que es trobin capes ocultes o no, han de distingir-se a la seva vegada dos nous casos:

    - XNA amb connexions cap endavant (*feedforward*). El flux d'informació es propaga sempre en ordre ascendent des de la capa d'entrada fins a la de sortida. Són xarxes especialment útils en el reconeixement i classificació de patrons.

    - XNA amb connexions cap endavant i cap enrera (*feedforward/feedback*). La informació es propaga cap a capes posteriors però pot també retrocedir a capes més properes a la d'entrada, amb valors modificats per les connexions que ha anat trobant en el seu recorregut. Es produeix així una certa retroalimentació del sistema amb informació interna. Tot i que també poden ser usades en la classificació de patrons, són més adequades per a l'heteroassociació, és a dir, associació d'una informació o patró d'entrada amb una diferent informació de sortida.

Segons la relació que mantinguin les dades d'entrada amb les de sortida, es distingeix entre:

- XNA *heteroassociatives*. La informació d'entrada s'associa a una sortida que no guarda cap semblança amb la primera. La finalitat de la xarxa és aprendre parelles de dades del tipus (X, Y), de manera que davant l'entrada X, es reconegui la seva informació associada Y i es presenti com a resposta. En el nostre cervell també es dóna aquesta associació entre records de diferent naturalesa, i en el camp de les

finances s'utilitzen XNA heteroassociatives tant per a la predicció com per a la classificació.

- XNA *autoassociatives*. A la informació d'entrada se li associa una informació que es fa coincidir amb ella mateixa, convertint-se el parell de dades a aprendre en un combinat de la forma (X, X). El seu objectiu és reconstruir dades d'entrada que són presentades de forma incompleta o que degut al soroll que han suportat es troben distorsionades. Es pretén que davant una entrada X', el sistema el reconegui i ofereixi com a sortida el seu valor net de distorsions X. Dins la memòria biològica, la ressonància seria el procés que hom utilitza com a mecanisme d'autoassociació i en el camp financer, aquestes XNA s'empren en el reconeixement de patrons.

Al capítol anterior (apartat 1.4) hem posat de relleu la importància de la noció d'aprenentatge. Entès com la modificació del comportament d'un sistema induït per la interacció amb l'entorn i com a resultat de l'experiència, que resulta en l'establiment de nous models de resposta a estímuls externs, el concepte tant és aplicable als éssers humans com a les XNA. En el cervell, l'aprenentatge és degut a l'alteració dels valors sinàptics de les connexions entre neurones. En les XNA, és el procés mitjançant el qual es modifiquen els pesos associats a una connexió, com a resposta a la presència de nova informació d'entrada. Els canvis que es realitzen durant aquest procés es redueixen a l'eliminació, modificació o creació de connexions entre neurones a través d'un algorisme matemàtic, l'*algorisme d'aprenentatge*, que canvia els valors dels pesos d'un període al següent. El procés conclourà quan s'assoleixi l'estabilitat en els pesos.

Podem establir una doble classificació segons el tipus d'aprenentatge[4] emprat:

1.a. XNA amb *aprenentatge supervisat*. En el procés d'aprenentatge intervé un agent extern que comprova la sortida de la xarxa i en cas de que no coincideixi amb la sortida desitjada, es modifiquen els pesos per tal d'apropar-se iterativament al valor buscat.

En els mercats financers, s'apliquen XNA supervisades amb finalitats predictives, utilitzant dades històriques, i per tant conegudes, per a que la xarxa detecti la relació entre els valors passats d'una variable i n'obtingui els paràmetres adequats per a representar-la. Els mateixos paràmetres són emprats per a projectar i

[4] La forma concreta que adopta l'algorisme d'aprenentatge es detallarà únicament en les XNA que s'apliquen en la tesi.

estimar futurs valors de la variable. Nosaltres les apliquem a l'anàlisi de la memòria-dependència.

1.b. XNA amb *aprenentatge no supervisat*. Són xarxes on no és necessària l'ajuda de cap agent extern en el procés d'aprenentatge, motiu pel qual es diu que són capaces d'autoorganitzar-se. La pròpia XNA, sense influències alienes, genera una resposta a una determinada entrada basant-se generalment en funcions de distància que mesuren la semblança entre les dades introduïdes en el sistema.

En els mercats financers s'apliquen a tasques de reconeixement de patrons o agrupació, tant d'actius com d'inversors, en funció de les seves característiques. Nosaltres les fem servir en l'estudi de la memòria-patró, però també en el cas més general de la memòria col·lectiva.

2.a. XNA amb *aprenentatge on line*. A l'igual que l'ésser humà està immers en un continu aprenentatge, també en aquestes XNA els pesos es modifiquen sempre que es presenta nova informació. Degut a aquest caràcter dinàmic són susceptibles de presentar inestabilitats.

L'adaptació constant dels pesos fa adequat el seu ús en entorns oberts, on l'univers del problema no està restringit i és possible l'aparició de noves situacions a analitzar per la xarxa. Malgrat el context financer s'ajusti a un entorn com el descrit, que una XNA tingui aprenentatge *on line* o *off line* depèn del model concret que s'utilitzi. Les xarxes ART per a la memòria-patró són un exemple del primer tipus i les *back-propagation* i els mapes de Kohonen són xarxes *off line*.

2.b. XNA amb *aprenentatge off line*. En elles està clarament diferenciada una etapa d'entrenament en la que es duu a terme la modificació dels pesos, i una fase de funcionament o operació en la que s'entén que la xarxa ja ha après, i fixats els pesos, únicament s'introdueix nova informació d'entrada per a obtenir una resposta. Caldrà tornar a entrenar la xarxa sempre que es vulgui adquirir nou coneixement.

Únicament com a detall operatiu, mencionarem una darrera classificació de XNA basada en la naturalesa de les dades que es presenten com a valors d'entrada i dels que s'obtenen com a sortida.

- Les XNA *analògiques* accepten valors reals, continus, com a dades del sistema. Acostumen, però, a normalitzar-se per a treballar amb valors compresos dins els intervals tancats [0, 1] o bé [-1, 1]. Són adequades per a tractar amb rendibilitats, volums de negociació, tipus d'interès, increments percentuals, etc.

- Si la XNA és *discreta*, també anomenada *binària,* les seves entrades i sortides únicament podran prendre els valors 0 (alternativament, -1) ó 1. En aquest cas no es poden incloure dades quantitatives de rendibilitat, sinó qualitatives, com ara augments (1) i disminucions (0) de valors, aplicats, per exemple, a la detecció de tendències.

- El cas *híbrid* és aquell en el que la informació d'entrada a la XNA pren valors de la recta dels reals però la sortida continua pertanyent al conjunt {0,1}. Són aplicables quan es pretén discriminar entre situacions de solvència o fallida, entre clients morosos i no morosos, etcètera.

De les diferents tipologies de XNA descrites i de les seves combinacions, es dedueix que hi ha un ampli ventall de models possibles per al funcionament d'una xarxa, donant lloc a tantes altres famílies d'arquitectures. A més, la varietat augmenta amb l'aparició de nous models que milloren els ja existents o que incorporen nous elements per a ajustar-se als requeriments dels problemes que van sorgint. Això fa molt difícil poder ser exhaustius en un llistat que reculli totes les diferents classes de xarxes. D'entre les més estudiades i utilitzades destaquem: *Adaline/Madaline, Bidirectional Associative Memory, Adaptive Resonance Theory, Associative Reward Penalty, Back-propagation, Boltzmann Network, Cauchy Machine, Cognitron/Neocognitron, Counter-propagation, Fuzzy Associative Memory, Hopfield, Learning Vector Quantizer, Perceptró, Topology Preserving Map*, etc.

2.3. XNA PER AL TRACTAMENT DE LA MEMÒRIA ALS MERCATS FINANCERS.

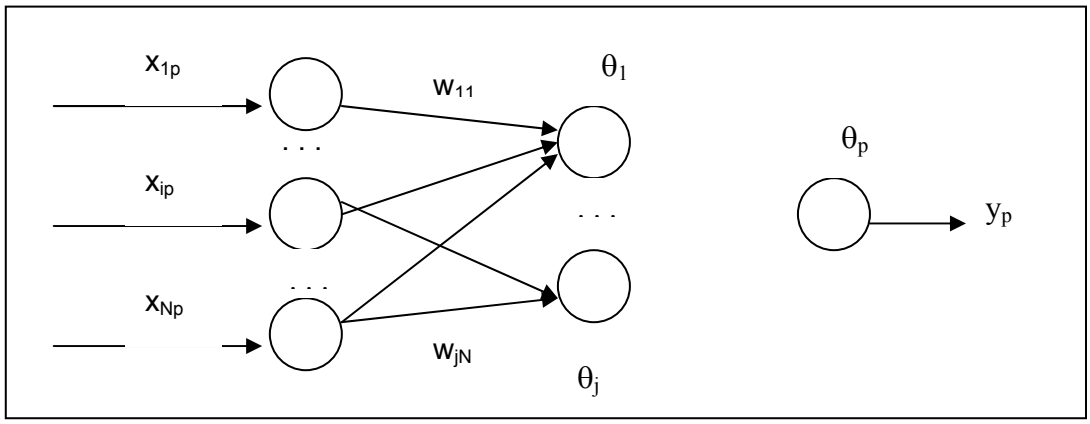
Les tres XNA que emprarem per a l'anàlisi de les diferents accepcions de memòria que hem distingit en els mercats financers són: les *back-propagation*, els mapes de Kohonen i les xarxes ART (*Adaptive Ressonance Theory*). A continuació en detallem el funcionament. En cadascuna d'elles s'estudiarà el seu origen, l'estructura, l'algorisme d'aprenentatge i les seves aplicacions més comunes.

2.3.1. Xarxes *Back-Propagation.*

Són el resultat de les investigacions de D. Rumelhart, G. Hinton i R. Williams (1986). Correctament, ens hi hauríem de referir com a *xarxes perceptró multicapa amb algorisme d'aprenentatge per retropropagació de l'error*, ja que provenen del primer model de XNA que es coneix, el perceptró, i el tret més rellevant del seu funcionament està en la manera en que es modifiquen els pesos per tal de disminuir l'error comès pel sistema, tal com veurem posteriorment. D'ara en endavant, però, ens hi referirem únicament com a xarxes *back-propagation*.

Són les XNA més utilitzades. Entre les seves aplicacions destaquen l'ajust de sèries temporals amb la finalitat de predicció, reconeixement de patrons coneguts, codificació d'informació, compressió i descompressió de dades, etc. També l'àmbit econòmic i financer ha incorporat[5] aquesta metodologia tant en el context investigador com en l'operativa més propera al funcionament quotidià dels mercats financers. Malgrat tot el ventall d'aplicacions existent, l'aspecte més rellevant per a nosaltres i el que volem destacar aquí, està en relació al seu ús, en el següent capítol, com a instrument per a l'anàlisi de la memòria-dependència en una sèrie temporal financera.

Una xarxa *back-propagation* 3-2-1, amb tres neurones a la capa d'entrada, una capa oculta amb dues neurones i sortida unitària, té l'estructura representada en el següent gràfic:



Gràfic 2.2. Estructura d'una *back-propagation* 3-2-1.

[5] Entre els treballs on s'han emprat xarxes *back-propagation* es troben: Pérez-Rodríguez, Jorge V. i Salvador Torra (2001) : " Diversas Formas de Dependencia No Lineal y Contrastes de Selección de Modelos en la Predicción de los Rendimientos del Ibex-35" en *FEDEA*; Skabar, Andrew i Ian Cloete (2001): "Neural Networks, Financial Trading ant the Efficient Market Hipótesis" en *Twenty-Fifth Australian*

Cal matisar els següents aspectes:

- És una xarxa heteroassociativa. Un vector d'entrada, anomenat *patró*, és associat amb un vector de sortida diferent. Al vector d'entrada el denotarem com $X_p = (x_{1p}, x_{2p}, ..., x_{Np})$ on l'índex p fa referència al p-éssim patró amb el que s'entrena la xarxa dins del conjunt de patrons $\{X_1, X_2, ..., X_p, ..., X_P\}$ i P és el nombre de patrons del que es disposa.
- La capa d'entrada té N neurones, tantes com components defineixen el patró.
- La capa de sortida té M neurones, una per a cada element que composa la sortida de la xarxa, $Y_p=(y_{1p}, y_{2p}, ..., y_{Mp})$, nombre que estarà en funció de les components de la sortida realment desitjada $D_p= (d_{1p}, d_{2p},..., d_{Mp})$.
- El nombre de capes ocultes i de neurones en elles és variable. Tot i que no hi ha cap mètode per a determinar el nombre adequat, hi ha algunes propostes com les que figuren a T. Masters (1994), en la que $L= (N \cdot M)^{1/2}$ sent L el nombre de neurones ocultes, o la que apareix a Neural Work (1994), on L=P/[5(N+M)].
- Són XNA amb aprenentatge supervisat i *off line*. Degut a aquesta darrera característica, hi haurà una fase d'entrenament en la que els pesos de la xarxa es modifiquen mentre que en la fase de funcionament, romanen constants. Per ser supervisada, en la fase d'entrenament se li facilitarà a la xarxa un conjunt de P parells de la forma (X_p, D_p), on D_p és el vector de sortida desitjat per a la xarxa quan se li presenti el vector X_p. L'única manera per a arribar al valor desitjat és mitjançant la informació que rep d'entrada, que està donada, i els pesos w_{ji} de les connexions entre les neurones *i* i *j*, que haurà d'adaptar per a que el resultat sigui l'esperat.

L'algorisme d'aprenentatge que s'utilitza és el conegut com la regla delta generalitzada o regla de Widrow-Hoff. Un cop definit l'error que s'ha produït per comparació entre el resultat $Y_p=(y_{1p}, y_{2p},..., y_{Mp})$ i el vector desitjat, es procedeix a la minimització a través de canvis successius en els pesos, proporcionals al gradient decreixent de la funció error. És per aquesta raó que el procés d'optimització s'anomena de *descens pel gradient*.

*

Computer Science Conference, Melbourne. Vol. 4.; Rahimian, Eric; et al. (1996): " Bankruptcy Prediction by Neural Network", en *Neural Networks in Finance and Investing* [Trippi i Turban (1996:243-259)].

Els passos que cal seguir per tal d'implementar una *xarxa back-propagation*, resumits de forma esquemàtica, són:

1º. Es presenten els patrons d'entrada $X_p = (x_{1p}, x_{2p}, ..., x_{Np})$ al sistema, associant cada component a una neurona de la capa d'entrada.

2º. La informació es propaga a les neurones de les capes ocultes, rebent cadascuna d'elles la següent entrada: $Net_{jp} = \sum_{i=1}^{N} w_{ji} \cdot x_{ip} + \theta_j$. Els pesos w_{ji} inicialment són aleatoris i el paràmetre θ_j, que és opcional, rep el nom de *llindar* de la neurona j.

3º. El valor de sortida de cada neurona oculta és $Y_{jp} = f_j (net_{jp})$, on f_j és la funció de transferència de les unitats que pertanyen a aquesta capa. Degut a l'algorisme d'aprenentatge que utilitzen les xarxes *back-propagation*, és una condició necessària que la funció de transferència sigui contínua i derivable. La funció lineal f(x) = x, la sigmoidal $f(x) = 1/(1+e^{-\alpha x})$ i la tangent sigmoidal $f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$ compleixen els requisits.

4º. Es repeteix el mateix procediment fins a arribar a la capa de sortida, on $Net_{kp} = \sum_{j=1}^{L} w_{kj} \cdot y_{jp} + \theta_k$, sent w_{kj} el pes de la connexió entre la neurona oculta *j* i la neurona *k* de la capa de sortida i θ_k novament el llindar, però d'una neurona de la capa de sortida. El valor que dóna com a resposta el sistema s'obté de $Y_{kp} = f_k (net_{kp})$ on f_k és la funció de transferència per a la neurona k, amb les mateixes característiques de continuïtat i derivabilitat anteriorment esmentades.

5º. L'error comès per la xarxa s'obté comparant el valor de sortida amb el desitjat. Per a un patró p, $E_p = \frac{1}{2} \sum_{k=1}^{M} \delta_{kp}^2$ on $\delta_{kp} = (d_{kp}-y_{kp}).f'_k (net_{kp})$ i f'_k és la derivada de la funció de transferència associada a la neurona k. Si s'utilitza una funció lineal, $\delta_{kp} = (d_{kp} - y_{kp})$, mentre que si s'utilitza la sigmoidal, $\delta_{kp} = (d_{kp} - y_{kp})\, y_{kp}\, (1 - y_{kp})$ i amb la tangent sigmoidal $\delta_{kp} = (d_{kp} - y_{kp})\, (y_{kp} +1)\, (1 - y_{kp})$.

6º. L'error és ara el que es propaga en sentit invers al flux d'informació dins de la xarxa. Per a cada neurona oculta caldrà calcular $\delta_{jp} = f'(net_{jp}) \cdot \sum_k w_{jk} \cdot \delta_{kp}$, convertint-se en l'expressió $\delta_{jp} = \sum_k w_{jk} \cdot \delta_{kp}$ si la funció de transferència és lineal, $\delta_{jp} = x_{ip} (1 - x_{ip}) \cdot \sum_k w_{jk} \cdot \delta_{kp}$ si és sigmoidal i $\delta_{jp} = (x_{ip} + 1)(1 - x_{ip}) \cdot \sum_k w_{jk} \cdot \delta_{kp}$ en el cas que treballem amb funcions tangent sigmoidals.

7º. L'aprenentatge es concreta en la modificació dels pesos de la xarxa. Els nous pesos són, per a les neurones de la capa de sortida $w_{kj}(t+1) = w_{kj}(t) + \Delta w_{kj}(t+1)$ on $\Delta w_{kj}(t+1) = \alpha \, \delta_{kp} \, y_{jp}$, i per a les neurones ocultes $w_{ji}(t+1) = w_{ji}(t) + \Delta w_{ji}(t+1)$, amb el valor $\Delta w_{ji}(t+1) = \alpha \, \delta_{jp} \, x_{ip}$. En ambdós casos, α és un paràmetre que representa la *taxa d'aprenentatge*.

8º. El procés es repeteix introduint novament tots els patrons fins que l'error comès per la xarxa es considera acceptable.

*

Per a concloure, comentarem dos problemes que poden sorgir en relació a la bondat dels resultats obtinguts: la convergència de la funció error cap a un mínim i la mesura de l'error total comès pel sistema. En relació al primer aspecte, la velocitat de convergència de la funció error cap a un mínim dependrà tant de la superfície concreta que dibuixi la funció error com del valor escollit per a la taxa d'aprenentatge. No és segur però, que el mínim assolit sigui el global. Depenent dels pesos aleatoris inicials i de la seva modificació, la xarxa pot establir-se en equilibri en un mínim local. Per tal d'evitar aquest fet caldria reinicialitzar novament els pesos amb uns valors diferents i modificar la taxa d'aprenentatge. De totes maneres, si l'error total és assumible, el mínim local pot ser un bon resultat. Es fa necessari, en conseqüència, disposar d'una mesura per a l'error total comès per la XNA. Una de les més utilitzades és l'error quadrat mig normalitzat o NMSE (*Normalized Mean Squared Error*)[6].

[6] Donat un conjunt P de parells de valors d_p (el valor desitjat) i els valors resultants de la xarxa, y_p, el valor del NMSE, resultat de dividir l'error quadrat mig entre la variància de la sèrie, s'obté mitjançant l'expressió:

2.3.2. Els mapes de Kohonen.

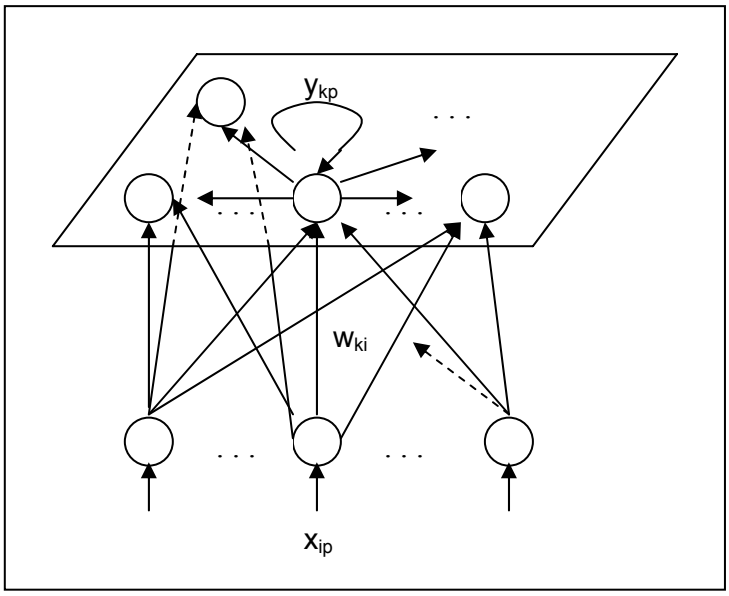
Al capítol anterior ens hem referit a la capacitat del cervell humà per a treballar formant mapes topològics. Dèiem que els records en la fase de consolidació no estaven organitzats de forma aleatòria sinó que s'estructuraven de manera que quedessin agrupats en funció de les seves similituds. De la mateixa manera, la informació captada pels sentits també és representada en forma de mapes bidimensionals. Seguint aquest model, el finlandès Teuvo Kohonen desenvolupà al 1982 un tipus de XNA que es coneix amb el nom de *mapes de Kohonen* o *SOM* (*Self Organizing Maps*)[7].

Els mapes de Kohonen disposen la informació d'entrada, inicialment en forma de vectors n-dimensionals, en un espai de dues dimensions en funció de la semblança entre les seves característiques, o formalment, en funció de la distància entre el valor de les seves components. El resultat és un *mapa de característiques*, on els patrons que estan propers tenen unes característiques similars i les diferències amb altres patrons augmenten a mesura que també ho fa la seva distància espacial. L'avantatge de disposar d'un instrument com aquest és el de poder establir relacions entre la informació d'entrada en funció de la seva disposició topològica, a la vegada que permet, donat un patró del que desconeixem el seu comportament, atribuir-li el d'un altre de conegut que estigui junt a ell en el mapa, ja que en els SOM es suposa que entrades amb característiques comuns donaran lloc a sortides també properes.

Un mapa de Kohonen és una XNA heteroassociativa; amb connexions *feedforward* entre el vector d'entrada i les neurones que formen el mapa o capa de sortida, connexions laterals entre una neurona de la capa de sortida i les seves veïnes, i autorrecurrents també en la capa de sortida. Esquemàticament es pot representar:

$$\text{NMSE}=\frac{\sum_{p\in P}\left(d_p - y_p\right)^2}{\sum_{p\in P}\left(d_p - \bar{d}_p\right)} = \frac{1}{\sigma_P^2}\frac{1}{N}\sum_{p\in P}\left(d_p - y_p\right)^2$$

essent σ^2 la variància estimada de les dades, $\bar{d}_p$ la mitjana i N el nombre d'elements del conjunt P.



Gràfic 2.3. Estructura d'un mapa de Kohonen.

L'aprenentatge de les SOM és *off line*, però a diferència de les xarxes *back-propagation*, l'aprenentatge és no supervisat. Aquesta propietat permet tractar únicament amb informació d'entrada de la que en desconeixem l'*output* corresponent, sent la pròpia xarxa la que n'estableix el valor de sortida en funció del grau de semblança dels patrons introduïts. És per aquest motiu que es diu que tenen la capacitat d'autoorganitzar-se.

Dins del no supervisat, els mapes de Kohonen utilitzen l'*aprenentatge competitiu*. Rep aquest nom perquè en ell les neurones competeixen, podent també cooperar, mitjançant els pesos de les connexions, per a que davant un vector d'entrada només quedi activada una neurona de la capa de sortida, neurona que rep el nom de *guanyadora*. Aquesta neurona és la representant del vector d'entrada en qüestió en la capa de sortida.

*

El mecanisme de competició i la resta de procediments necessaris per a implementar un mapa de Kohonen es resumeixen tot seguit:

[7] Existeixen dues versions: una xarxa d'una sola capa, que rep el nom de *Learning Vector Quantizer* (LVQ) i una de dues capes anomenada *Topology Preserving Map* (TPM) que és la que emprem en la tesi.

1º. La informació d'entrada $X_p = (x_{1p}, x_{2p}, ..., x_{Np})$ es propaga cap a les neurones de la capa de sortida, mitjançant les connexions de pes w_{ki} entre la neurona d'entrada *i* i la de sortida *k,* inicialment aleatoris.

2º. Juntament amb la informació rebuda provinent de les connexions laterals i autorrecurrents, les neurones de la capa de sortida emeten una senyal y_{mp} (t+1) = $f\left(\sum_{i=1}^{N} w_{ki} \cdot x_{ip} + \sum_{k=1}^{M} \varphi_{pk} \cdot y_{kp}(t)\right)$ on φ_{pk} és una funció que recull la interacció entre les neurones de la capa de sortida. Degut a que quan més a prop estiguin dues neurones, major serà la influència que l'una efectuï sobre l'altra, φ_{pk} és una funció de les anomenades de *barret mexicà*, prenent valors elevats entre neurones properes i disminuint a mesura que augmenta la distància entre elles. La funció f és la de transferència i pot ser qualsevol funció contínua, ja que la xarxa treballa amb valors reals, acostumant a emprar-se la lineal o la sigmoidal.

3º. S'estableix la competició entre les neurones de la capa de sortida, després de la qual només una restarà activa, la *neurona guanyadora*, i inactives tota la resta. La guanyadora serà la que presenti una menor distància[8] entre el patró d'entrada i els pesos que li són associats. Formalment,

$$y_{kp} \begin{cases} 1 \text{ si } \min ||X_p - W_K|| = \min \sqrt{\sum_{i=1}^{N} (x_{ip} - w_{ki})^2} \\ 0 \text{ per a la resta} \end{cases}$$

4º. Es modifiquen els pesos de la neurona guanyadora i de les que formen part de la seva zona de veïnatge.

$$w_{ki}(t+1) = w_{ki}(t) + \alpha(t) \cdot [x_{ip} - w_{k^*i}(t)] \quad \forall k \in V_{k^*}(t)$$

on $V_{k^*}(t)$ és l'àrea de veïnatge en torn a la neurona guanyadora k*. Pot ser una zona circular, quadrada, hexagonal, o qualsevol altra amb els requisits de que estigui centrada en la neurona k* i dibuixi entorn a ella un polígon regular. Està en funció de t, nombre d'iteracions, perquè l'àrea de veïnatge disminueix a mesura que la xarxa aprèn, per tal de garantir la convergència cap a l'equilibri. $\alpha(t)$ és el coeficient

[8] En els mapes de Kohonen s'acostuma a utilitzar la distància euclídea o una transformació d'ella, encara que altres distàncies podrien ser igualment vàlides.

d'aprenentatge i també disminueix amb el nombre d'iteracions. Una de les més usuals és l'expressió: $\alpha(t) = 1/t$.

*

Les aplicacions dels mapes de Kohonen són totes aquelles possibles per a XNA amb aprenentatge no supervisat, és a dir, el reconeixement de patrons, la resolució de problemes d'optimització, codificació de dades, etc.; havent-se també ja aplicat a problemes específics del món financer[9]. En particular, i degut a la seva estructura que reprodueix la capacitat humana de la memòria i el seu funcionament, és l'instrument que proposem per a l'anàlisi de la memòria col·lectiva d'un mercat financer.

2.3.3. Les xarxes ART (*Adaptive Ressonance Theory*).

En una emulació de la memòria biològica, podríem dir que les xarxes *back-propagation* i les de Kohonen pateixen d'*inhibició proactiva*, és a dir, el coneixement que tenen desat en la memòria, en els pesos de la xarxa, satura la seva capacitat per a adquirir nou coneixement. Si volem que emmagatzemin nova informació, hem de tornar a entrenar-les, però no únicament amb les noves dades sinó també amb tot el conjunt de patrons que ja coneixen, ja que en cas contrari, els oblidarien.

Aquest comportament no s'adiu amb el de l'ésser humà, que és capaç d'aprendre nova informació sense que per això comporti oblidar la que ja havia consolidat i formava part de la memòria a llarg termini. A partir d'aquesta base, Gail A. Carpenter i Stephen Grossberg (1987a), van dissenyar un nou model de XNA que reuneix la propietat de mantenir la informació apresa i retinguda en els pesos, *propietat d'estabilitat,* amb la capacitat de continuar aprenent nova informació, *propietat de plasticitat*. Aconseguiren conjugar ambdues propietats en una sola xarxa gràcies a la *teoria de la ressonància adaptativa* (*Adaptive Resonance Theory*), mitjançant un sistema de retroalimentació en base a connexions *feddforward/feedback*.

Degut a les seves característiques, l'aplicació fonamental de les ART és el reconeixement i agrupació de patrons. Així, nosaltres les apliquem a l'anàlisi de la memòria-patró i a la detecció de climes en un mercat financer.

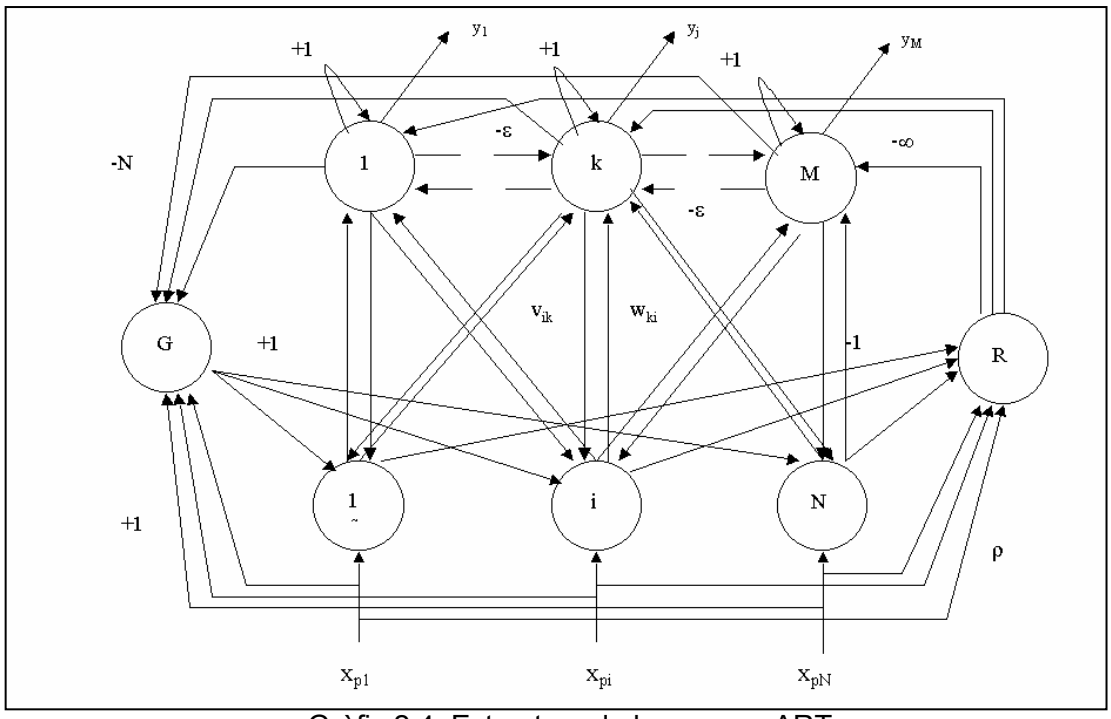
[9] Per exemple, vegi's Lee, K., D. Booth i A. Pervaiz (2003).

Les ART són xarxes amb aprenentatge no supervisat i competitiu. Estan formades per dos subsistemes: el *subsistema d'atenció* i el *d'orientació*. El primer té com a finalitat reconèixer i classificar la informació apresa, mentre que el segon és l'encarregat de detectar si *l'input* o patró que se li presenta al sistema pertany a una de les categories ja establertes o bé, si cal afegir-ne una de nova. La decisió sorgirà de la comparació entre la informació d'entrada i els representants dels grups que la xarxa ja ha creat, recollits pels seus pesos. A més, cal un paràmetre amb el que realitzar la comparació, anomenat *paràmetre de vigilància* (ρ), escollit per la persona que dissenya la xarxa. L'esquema gràfic de l'arquitectura de les ART està recollida en el gràfic 2.4.

A més de les connexions *feedforward/feedback* ja comentades, també hi ha connexions laterals i autorrecurrents en les neurones de la capa de sortida per a permetre la competició. Els pesos de cadascuna d'elles es detallen en el gràfic 2.4. Representem per ε un valor petit, mai superior a 1/M, per garantir la convergència de la xarxa, i amb el símbol $-\infty$ un pes suficientment gran i negatiu com per a inhibir completament una neurona.

Dos elements específics d'aquesta xarxa són les neurones G, pròpia del subsistema d'atenció, i R, que duu a terme la tasca del subsistema d'orientació. La neurona G fa la funció de *control de guany d'atenció*. Mitjançant els seus pesos s'aconsegueix augmentar el grau d'atenció o sensibilitat de les neurones de la capa d'entrada, evitant que s'anticipin a l'arribada d'informació del següent patró. Donat que les ART també disposen de connexions *feedback*, quan aquestes trameten la seva informació a les neurones de la capa d'entrada, es podria confondre la informació de dins del sistema amb informació exterior, com si es tractés d'un nou patró. El sistema de control de guany d'atenció, mitjançant el valor dels pesos i de les seves sortides, assegura que no es doni aquesta situació.

D'altra banda, la neurona R fa la funció del subsistema d'orientació. Determina si la informació d'entrada és prou semblant com per formar part d'algun dels grups ja establerts o se'n crea un de nou del qual la informació entrant en serà el representant. En el primer cas, R permetrà que la neurona guanyadora romangui activa i s'ajustin els pesos d'aquesta per tal de que el prototip del grup representi també d'alguna manera les característiques del nou patró incorporat. Si no hi ha prou semblança entre la informació entrant i la neurona guanyadora, la funció del subsistema d'orientació és desactivar aquesta última momentàniament, permetent la comparació amb la resta de neurones de la capa de sortida. Si no hi ha cap adient, s'afegirà una unitat més.



Gràfic 2.4. Estructura de les xarxes ART.

La forma que adopta la sortida de cadascuna de les dues neurones per tal de poder desenvolupar les seves funcions és:

$$y_G \begin{cases} 1 & \text{si } \sum_{i=1}^{N} x_{ip} - N\sum_{k=1}^{M} y_k \geq 0'5 \\ 0 & \text{si } \sum_{i=1}^{N} x_{ip} - N\sum_{k=1}^{M} y_k < 0'5 \end{cases}$$

$$y_R \begin{cases} 1 & \text{si } \sum_{i=1}^{N} \rho \cdot x_{ip} - \sum_{i=1}^{N} y_{x_{ip}} > 0 \\ 0 & \text{si } \sum_{i=1}^{N} \rho \cdot x_{ip} - \sum_{i=1}^{N} y_{x_{ip}} \leq 0 \end{cases}$$

Ens cal conèixer també els valors de sortida de les neurones de la capa d'entrada i de sortida.

Per a les neurones de la capa d'entrada:

$$net_{x_{ip}} = x_{ip} + \sum_{k=1}^{M} v_{ik} \cdot y_k + y_G \text{ i a partir d'aquí } y_{x_{ip}} = \begin{cases} 1 & \text{si } net_{x_{ip}} \geq 1'5 \\ 0 & \text{si } net_{x_{ip}} < 1'5 \end{cases}$$

Si es treballa amb dades binàries, on els valors d'entrada únicament poden ser 1 ó 0, l'expressió se'ns simplifica a:

$$y_{x_{ip}} = \begin{cases} x_{ip} \text{ si } y_k = 0 \ \forall k \\ x_{ip} \cdot \sum_{k=1}^{M} v_{ik} \cdot y_k \text{ en altre cas} \end{cases}$$

Al tractar-se d'una XNA amb aprenentatge no supervisat i competitiu, únicament una neurona de la capa de sortida restarà activada, amb valor de sortida $y_k = 1$, restant inactives les demés.

El procés per a determinar la neurona guanyadora i la posterior modificació dels pesos que comporta el procés d'aprenentatge es resumeixen a continuació.

*

1º S'introdueix un patró d'entrada $X_p = (x_{1p}, x_{2p}, ..., x_{Np})$ al sistema.

2º La informació es propagarà cap a la capa de sortida mitjançant uns pesos w_{ki}. Per a cada neurona de la capa de sortida s'obté:

$$y_k(t+1) = f\left[y_k(t) - \varepsilon \sum_{\substack{m=1 \\ m \neq k}}^{M} y_m(t) + \sum_{i=1}^{N} w_{ki} y_i(t) \right]$$

on f és la funció esglaó.

3º Després de la competició únicament romandrà activa una neurona, determinant-se de la següent manera:

$$y_k(t+1) = \begin{cases} 1 & \text{si } \max\left(\sum_{i=1}^{N} w_{ki} \cdot x_{ip}\right) \\ 0 & \text{per a la resta} \end{cases}$$

4º Mitjançant els mecanismes de retroalimentació, aquesta informació torna a la capa d'entrada:

$$x_{ip} = \sum_{k=1}^{M} v_{ik} \cdot y_k = v_{ik^*} \quad \text{ja que } y_k = \begin{cases} 1 & \text{si } k=k^* \\ 0 & \text{si } k \neq k^* \end{cases}$$

5º Es compara X_p amb V_{k^*}, on X_p és la informació d'entrada i el vector V_{k^*} el representant o prototip de la classe k*-éssima que té emmagatzemada la xarxa ART i el quocient

$$\frac{\left\|X_p \cdot V_{k^*}\right\|}{\left\|X_p\right\|} = \frac{\sum_{i=1}^{N} x_{ip} \cdot v_{ik^*}}{\sum_{i=1}^{N} x_{ip}}$$ és una relació de semblança.

Aquest resultat és el que es compara amb el paràmetre de vigilància $\rho \in [0, 1]$. Quan més proper a 1 es trobi ρ, més semblants hauran de ser diferents patrons per pertànyer a un mateix grup i per tant, més homogenis seran els elements que el formin. En els cas extrem de ρ=1, només s'admetrien en el mateix grup patrons idèntics. En l'extrem oposat, si ρ=0, tots els patrons s'agruparien en una única classe.

6º Si $\frac{\left\|X_p \cdot V_{k^*}\right\|}{\left\|X_p\right\|} < \rho$, s'activa la neurona R, que anul·la momentàniament la neurona guanyadora i reinicia el procés de comparació amb les restants. Si cap d'elles és adient per a representar el patró d'entrada , es crea un nou grup. En cas de que la relació de semblança sigui superior a ρ, els pesos de la neurona guanyadora – però només els d'ella, per a mantenir la informació dels restants grups intacta – es modificarien per a incorporar alguna de les característiques del nou element del grup.

7º La modificació dels pesos, l'aprenentatge, ve donat per l'expressió $\frac{dv_{ik}}{dt} = y_k\left(y_i - v_{ik}\right)$. Si la neurona k no és la guanyadora, $\frac{dv_{ik}}{dt} = 0$ i per tant els pesos no es modifiquen. Si ho és, $\frac{dv_{ik^*}}{dt} = y_i - v_{ik^*} = v_{ik^*}\cdot x_{pi} - v_{ik^*}$. Llavors,

$$v_{ik^*}(t+1) = v_{ik^*} + \frac{dv_{ik^*}}{dt} = v_{ik^*}(t)\cdot x_{ip} \text{ i}$$

$$w_{ik^*}(t+1) = \frac{v_{ik^*}(t)\cdot x_{ip}}{\gamma + \sum_{i=1}^{N} v_{ik^*}(t)\cdot x_{ip}}$$

☞ En relació als valors dels pesos, inicialment, v_{ik}= 1, i w_{ki}= $\frac{1}{1+N}$, ja que els pesos *feedforward* són els mateixos pesos *feedback*, però normalitzats[10], és a dir, $w_{ki} = \frac{v_{ik}}{\gamma + \sum_{i=1}^{N} v_{ik}}$.

El funcionament descrit correspon al model ART1, que treballa amb dades binàries. Posteriorment, s'ha desenvolupat el model continu ART2 (Carpenter i Grossberg, 1987b) que admet valors reals i el model que permet entrades en forma de nombres borrosos (*fuzzyART*). Les expressions són semblants a les exposades però modificant les relacions de semblança a la naturalesa dels patrons d'entrada. En el capítol 4 emprarem les xarxes ART2, degut a que treballarem amb patrons formats per rendibilitats, volums de negociació i indicadors que prenen valors reals. També mostrem, doncs, les expressions necessàries per a la seva implementació.

L'estructura és la mateixa que la representada en el gràfic 2.4. (pàg. 59), essent ara w_{ji} = v_{ij}, és a dir, els pesos de les connexions *feedforward* i *feedback* es suposa que tenen el mateix valor.

Els dos primers passos que cal seguir són compartits amb els de la xarxa ART1 descrita anteriorment. A partir d'aquell moment, comencen les diferències:

[10] γ pren un valor petit. Normalment es considera 1 en t=0 i després γ=0'5.

3º La neurona guanyadora (k*) és la que minimitza la distància entre el patró d'entrada i els pesos de les connexions entre aquesta i les neurones de la capa d'entrada. Formalment:

$$y_{kp} = \begin{cases} 1 & \text{si } \min \| X_p - W_k \| = \min\left(\sum_{i=1}^{N} \left| x_{ip} - w_{ki} \right|\right) \\ 0 & \text{per a la resta} \end{cases}$$

4º És el valor d'aquesta distància per a la neurona guanyadora la que determina la relació de semblança que es compararà amb el paràmetre de vigilància[11].

Si $\| X_p - W_{k^*} \| < \rho$, es desestima la neurona guanyadora i es torna a repetir el procés de competició sense ella. En cas de que cap neurona compleixi aquesta condició, es crearà un grup nou, afegint-se una unitat més a la capa de sortida.

Si $\| X_p - W_{k^*} \| > \rho$, la neurona guanyadora és la representant del patró d'entrada i els seus pesos es modificaran segons l'expressió:

$$w_{k^*i}(t+1) = v_{ik^*}(t+1) = \frac{x_{ip} + w_{k^*i}(t) N_{k^*}(t)}{N_{k^*}(t) + 1}$$

essent $N_{k^*}(t)$ el nombre de patrons en el moment *t* que s'agrupen en el prototip representat per la neurona k*.

*

Malgrat hem destacat l'avantatge que tenen les ART per la seva capacitat d'emmagatzemar nous patrons sense perdre informació ja apresa, també presenta algunes limitacions, com ara la seva dependència respecte l'ordre en el que s'introdueixen els patrons a la xarxa, podent obtenir agrupacions diferents únicament en funció de l'ordre d'entrada dels vectors *input.* La dependència de la composició dels grups al valor del paràmetre de vigilància també pot semblar un inconvenient, malgrat nosaltres no opinem d'aquesta manera. L'elecció del paràmetre per part del

dissenyador no és subjectiva, sinó que és una forma de controlar el nombre i grau d'homogeneïtat entre els elements de cada grup, és una elecció entre la simplicitat de pocs *clusters* o el rigor de grups molt homogenis. En els mercats financers, aquesta decisió pot significar prendre una posició davant el nivell de risc que es vol assumir.

2.4. Consideracions finals.

El present capítol, juntament amb l'anterior, formen el bloc dedicat a la presentació dels conceptes i metodologia necessaris per al desenvolupament de la tesi. Però no han estat concebuts per ser aplicats a un àmbit general, sinó que han estat enfocats cap a la temàtica pròpia de la nostra investigació. És sota aquesta perspectiva que hem d'entendre la nostra referència a les xarxes neurals artificials. Hem introduït breument la seva definició, elements i tipologia, centrant el nostre interès en les relacions entre l'aproximació connexionista i la memòria en cadascuna de les seves accepcions.

Així concloem que, degut a les característiques que defineixen a les XNA, podem aplicar-les a qualsevol definició de memòria als mercats financers. En primer lloc, proposem l'ús de les xarxes *back-propagation* en l'anàlisi de la memòria-dependència, degut a la seva capacitat per a detectar i ajustar relacions de caire no lineal entre diferents variables, o en el nostre cas, de la mateixa variable en diferents períodes de temps.

Les xarxes ART, capaces de resoldre el conflicte entre estabilitat i plasticitat, ens proporcionen una eina molt útil per al tractament de la memòria-patró, doncs tant les unes com l'altra estan basades en els fenòmens de ressonància i autoassociació, ajudats per mecanismes de retroalimentació. El reconeixement de patrons propi de les xarxes ART és el que aplicarem per a detectar "climes" i repeticions de gràfiques en l'evolució de cotitzacions històriques, recolzant i aportant nous matisos a les eines de l'anàlisi tècnica.

Finalment, els mapes de Kohonen són la nostra proposta per a l'estudi sobre la memòria col·lectiva als mercats financers. La seva capacitat per a autoorganitzar-se, provinent d'un sistema d'aprenentatge no supervisat, i la distribució dels patrons en un

[11] En aquest cas, el paràmetre de vigilància està comprés entre 0 i el valor màxim de les components dels patrons d'entrada. A efectes pràctics, com nosaltres tractarem amb dades normalitzades, el valor de ρ continuarà estant entre 0 i 1.

mapa topològic sobre el que realitzar comparacions intertemporals amb ell mateix o amb altres patrons en un mateix període, configura, al nostre entendre, l'instrument d'anàlisi més robust per a un concepte tan dinàmic, complex i ampli com és el de memòria col·lectiva.

En els tres casos, hem inclòs els passos necessaris per a la implementació de les XNA. La llista amb les sentències concretes depenen del llenguatge de programació, motiu pel qual no n'hem especificat cap. El *software* utilitzat en les aplicacions de la tesi es fa constar en els capítols corresponents.