4. REDES NEURAIS

Neste capítulo será apresentada uma visão global sobre as redes neurais e a sua utilização junto ao reconhecimento de padrões. O objetivo é apresentar, de uma maneira simples e resumida, os conceitos desta área de conhecimento, de modo a possibilitar ao leitor uma introdução ao assunto. São dadas desta forma condições para um maior entendimento do que são e para que servem as redes neurais, bem como são fornecidos subsídios para um maior aprofundamento no assunto. Após serem apresentados os principais tópicos a respeito das redes neurais, poderá ser justificada a escolha do modelo adotado na implementação do sistema N^2OCR, bem como ficará mais clara a forma de funcionamento e características deste sistema.

Serão apresentados um histórico, a evolução e os modelos de redes neurais de grande importância na atualidade, onde para cada modelo serão analisadas as suas características principais, vantagens e desvantagens, no que diz respeito ao reconhecimento de padrões. Os modelos abordados são: Perceptron, Adaline, Madaline, rede multinível, modelo de Hopfield, máquina de Boltzmann, BAM, modelo de Kohonen e ART.

O estudo das redes neurais é uma área de pesquisas multidisciplinar, envolvendo conhecimentos de diversas áreas como: Medicina, Biologia, Física, Computação, e outras. Isto gera certos "conflitos" a cerca da nomenclatura e dos diferentes enfoques na abordagem do assunto. Neste capítulo será abordado o assunto orientado à Ciência da Computação.

4.1 Conceitos Básicos

As redes neurais foram desenvolvidas a partir de uma tentativa de criar um modelo que descrevesse a estrutura e o funcionamento dos neurônios do cérebro humano.

Os modelos de redes neurais buscam definir novos computadores ou novos modelos de processamento de dados. Estes devem apresentar um comportamento baseado em modelos neurobiológicos ao invés de modelos baseados em "circuitos de silício" (portas lógicas, circuitos combinacionais, biestáveis, etc).

O sistema nervoso humano é o responsável pelas tarefas de automatismo, decisões, integração de idéias ou sensações, e de adaptação do organismo e do próprio ser, sendo esta última realizada através do aprendizado. O centro do sistema nervoso é constituído por células que são responsáveis pelo seu funcionamento, onde estas recebem, geram e transmitem os estímulos que chegam ou partem de nosso cérebro. Estas células são denominadas de *neurônios* e estão presentes em nosso cérebro em uma quantidade da ordem de 10^{11} com cerca de 10^{15} conexões entre elas [KOH 88a]. A figura 4.1 apresenta a estrutura biológica de um neurônio, a qual foi utilizada como base para o desenvolvimento da teoria sobre redes neurais.

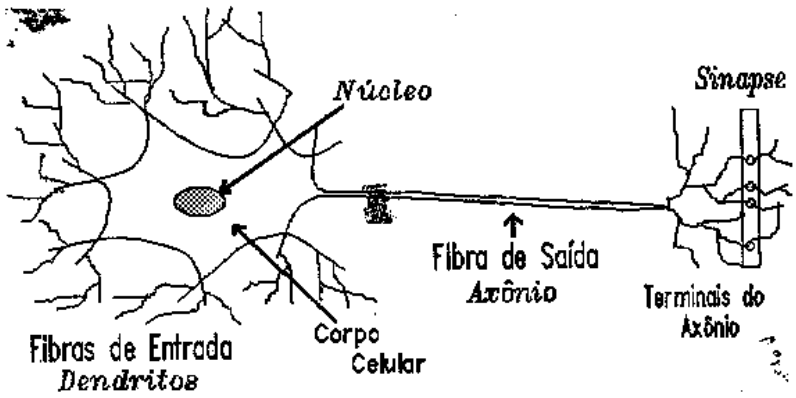


Figura 4.1 - Aparência externa de um Neurônio

A partir da observação do comportamento de sistemas reais (redes neurais de seres humanos), foi possível descrever de uma maneira muito simplificada o comportamento do neurônio. Este comportamento, de uma forma bem abstrata, será descrito a seguir. Os sinais (estímulos) de entrada, recebidos pelo neurônio através de seus *dendritos*, fazem com que este passe a gerar ou não um sinal de saída, o qual é enviado através do *axônio*. Cada neurônio é uma unidade independente de processamento de informações que está conectada a diversos outros neurônios através dos terminais do axônio. Estes terminais se conectam aos dentritos de outros neurônios, através de uma estrutura biológica denominada *sinapse* que possibilita a passagem dos sinais. A *sinapse* existente entre dois neurônios irá influenciar sobre o carater do sinal recebido, em função da atuação dos neurotransmissores, fazendo com que para um dado neurônio um sinal de entrada venha a funcionar como sinal excitador e para outro neurônio este mesmo sinal poderá atuar como inibidor.

Mas então surge a pergunta: o que tem a ver uma ciência exata, como é a computação, com estes conceitos biológicos? A resposta é simples, a partir do conhecimento da estrutura e do comportamento destes elementos básicos, é possível extrair suas características fundamentais e partir para a criação de modelos que imitem os sistemas reais. Estes modelos são utilizados na criação das chamadas *redes de neurônios artificiais*, também conhecidas como : redes neurais, neuroniais ou neuronais, modelos conectivistas (*connectionist models*), conectivismo (*connectionism*), inteligência natural ou processamento paralelo distribuído (PDP). O nome *rede de neurônios artificial* é o mais adequado, mas o nome de uso corrente, adotado na atualidade para denominar estes sistemas, é *redes neurais*.

A partir da definição de um modelo específico pode-se implementar programas de simulação, ou então implementar um hardware dedicado que implemente funcionalmente o modelo, utilizando-se de uma arquitetura de processamento paralelo.

Para se obter uma definição mais exata das redes neurais, é preciso determinar antes como funciona o neurônio (modelo comportamental). Ele possui várias entradas, as quais recebem estímulos de entrada (entrada ativa → valor positivo, entrada inativa → 0 ou valor negativo). De acordo com uma ponderação - *pesos sinápticos* ou *pesos de atuação* - sobre os sinais de entrada, estes terão um carater excitatório ou inibitório, fazendo com que o neurônio se "ative", enviando assim um sinal de saída, ou não se ative. Este sinal de saída será propagado de acordo com a topologia de interconexões da rede de neurônios.

Este é portanto um primeiro modelo de neurônio, formado por um integrador de sinais que avalia cada estímulo de entrada com o respectivo peso de atuação. Conforme a avaliação total das entradas, poderá ser gerado ou não um sinal de saída único, que será propagado pela rede através de suas interconexões. Existe um maciço paralelismo entre os elementos processadores (neurônios), todos avaliam em paralelo os seus estímulos de entrada.

O modelo descrito acima é uma descrição superficial do *Psychon* (figura 4.2), que foi o resultado do trabalho precursor de McCulloch e Pitts [MCC 43] na definição de modelos baseados no estudo do sistema nervoso. O *Psychon* é um dispositivo lógico de dois estados que, quando as entradas somadas ultrapassam um valor limite de excitação,gera um sinal binário de saída. Baseado apenas nestas informações já seria possivel especificar a implementação de um simulador para o mesmo.

O modelo do *Psychon*, apesar de representar razoavelmente bem o funcionamento de um neurônio, não previa a capacidade de adaptação. Não se pode formar novas memórias

com este modelo, uma vez que a ponderação dos estímulos é determinada externamente. Um trabalho muito importante que seguiu-se ao do *Psychon*, foi o do neurofisiologista D. O. Hebb [HEB 49], que influenciou a criação de muitos outros modelos.

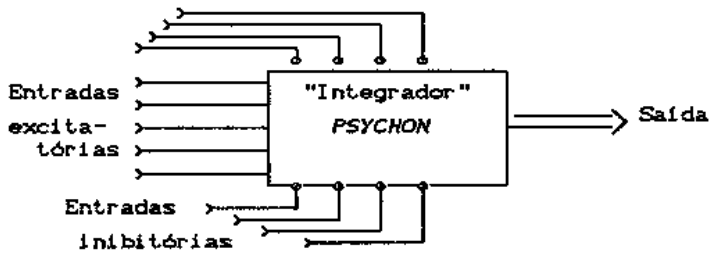


Figura 4.2 - Modelo do *Psychon* de McCulloch & Pitts

O trabalho de Hebb não resultou diretamente em um modelo específico e bem formalizado de uma rede neural, mas sim em uma análise das características e formas de comportamento que uma rede neural deveria possuir. Hebb propôs que a capacidade de aprendizado de uma rede neural reside na auto-organização das suas ligações sinápticas, ou seja, o aprendizado ou memória reside na distribuição de intensidades das ligações entre os neurônios. Afirmava Hebb que pares de neurônios que são ativados simultaneamente tornam-se mais fortes através de alterações em suas sinapses (pesos de atuação). Ele estabeleceu que as sinapses mais freqüentemente ativadas devem ter maior chance de se tornarem ativas novamente.

Baseando-se nos conceitos vistos anteriormente, é possível agora se determinar quais os itens que devem ser detalhados, visando a definição de um modelo específico de uma rede neural. Através do detalhamento destes itens, diferentes modelos de redes neurais são definidos, sendo que alguns destes modelos se destacam, e serão analisados

posteriormente neste capítulo. Os itens a serem especificados na definição de um modelo de rede neural são os seguintes [CAU 90]:

1) *Topologia da Rede* : Definição da forma de interconexão entre os neurônios, destacando-se as organizações de um nível linear, retro-alimentada, totalmente conectada, multinível, matricial ou livre (grafo). A figura 4.3 apresenta o esquema de algumas destas topologias de interconexão entre os neurônios.

2) *Definição do Neurônio Formal* : Especificação comportamental de um neurônio isolado. O que diferencia basicamente os neurônios é a denominada função de transferência. Esta função de transferência pode ser linear ou não, e determina o disparo (ativação da saída) ou não de um neurônio. A figura 4.6 apresenta as principais funções de transferência empregadas junto às redes neurais.

3) *Regra de aprendizado* : Determinação das formas pelas quais os pesos de atuação da rede serão modificados. Alterar os pesos de entrada dos neurônios implica alterar seu comportamento diante de um conjunto específico de estímulos de entrada.

Esta alteração de um comportamento diante uma dada situação (conjunto de estímulos) consiste de um aprendizado. A alteração dos pesos segue certas regras específicas, podendo ser um aprendizado do tipo supervisionado ou auto-aprendizado (*supervised* ou *self-learning*). Maiores detalhes a respeito dos itens vistos acima serão dados ao serem apresentados os modelos específico de redes neurais.

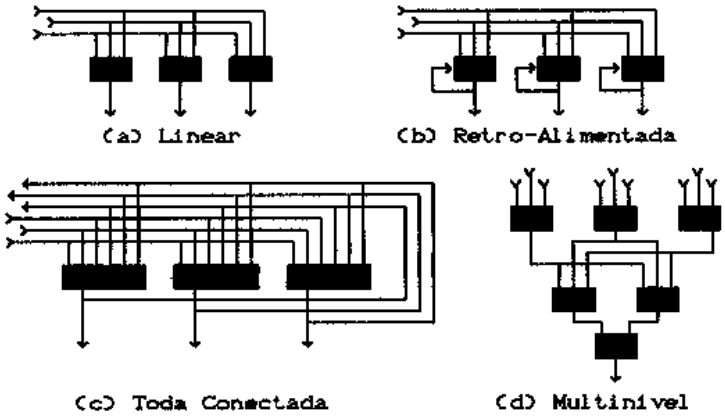


Figura 4.3 - Topologias de interconexões em redes neurais

4.2 Origem e Evolução

A origem e evolução dos estudos sobre redes neurais estão baseadas nos modelos que surgiram a partir dos anos 40s até a atualidade. Estes modelos, que se destacaram e tornaram-se famosos, foram sendo desenvolvidos ao longo deste período e a ordem cronológica dos fatos ajuda a compreender melhor esta evolução. Todos os modelos abordados neste item serão posteriormente descritos em maiores detalhes.

O estudo das redes neurais começou basicamente com a criação do *Psychon* em 1943 por McCulloch e Pitts [MCC 43], sendo que, em 1949, D. O. Hebb publicava a sua importante obra *"The organization of Behavior"* [HEB 49], que veio no futuro a influenciar vários modelos de redes neurais.

Em 1959, Frank Rosenblatt criou o *Perceptron* [ROS 59] que, como será visto, tem até hoje uma grande influência nos estudos sobre as redes neurais. Alguns outros modelos similares ao Perceptron foram também desenvolvidos nesta época, como é o caso do *Adaline*, criado por Bernard Widrow em 1962 [WID 62]. Estes modelos são baseados na correção de erros e formam uma importante classe de redes neurais.

Foi em 1969 que os modelos baseados no Perceptron receberam uma dura crítica feita por Minsky e Papert, através de sua obra "Perceptrons: An introduction to computational geometry" [MIN 69]. Através deste livro, ficou provado matematicamente que os modelos de redes neurais de apenas um nível baseados no Perceptron (aqueles que eram utilizados até o momento), não eram capazes de aprender uma simples função lógica "ou-exclusivo" (XOR - *Exclusive OR*). A função XOR possui um padrão de valores de entrada e saída que não podiam ser aprendidos pelos modelos de redes neurais propostos até aquela época.

O Madaline, criado também por Widrow, podia de certa forma resolver o problema, mas o aprendizado não podia ser realizado de uma forma muito "natural", como será visto posteriormente. Desta forma as redes neurais ficaram esquecidas por um certo tempo ...

Somente em 1982, surgia um novo modelo de rede neural de destacada importância. Criado por J. Hopfield [HOP 82], este novo modelo fez os estudos sobre as redes neurais ganharem um novo impulso. O modelo de Hopfield propunha um tipo de rede neural da classe das redes baseadas em sistemas competitivos, utilizando-se dos conceitos definidos por Hebb. Outros modelos similares ao de Hopfield surgiram logo após, todos pertencentes à classe de modelos com aprendizado competitivo. Estes novos modelos eram: a máquina de Boltzmann [HIN 84] e a BAM [KOS 87].

A década de 80 ficou marcada pelo aparecimento de um grande número de modelos importantes de redes neurais. Isto se deve principalmente à disponibilidade de novos computadores, mais velozes e com maiores potencialidades, o que permitiu a criação de um número maior de novos simuladores para as redes neurais, e também devido ao pouco sucesso da Inteligência Artificial em relação a uma série de aplicações (percepção visual, reconhecimento de fala, etc). Esta década foi muito importante, pois além do modelo de Hopfield foi desenvolvido o modelo multinível, que se utiliza da

regra de aprendizado de *Back-Propagation*. Este modelo foi desenvolvido por D. Parker [PAR 82] e D. Rumelhart [RUM 85] sendo baseado no Perceptron com correção de erros, mas implementando redes de múltiplos níveis. Estas redes podem inclusive resolver o problema do aprendizado da função XOR.

Além dos modelos de Hopfield e do modelo multinível, outro importante modelo também surgiu na década de 80, o modelo de Teuvo Kohonen, criado também em 1982 [KOH 82]. O modelo de Kohonen, é muito interessante pois permite um aprendizado competitivo e uma auto-organização da rede neural, criando os chamado "mapas de atributos de auto-organização".

Por fim o último modelo de destaque que será abordado é o modelo do ART, criado em 1983 por G. Carpenter e S. Grossberg [CAR 83]. Este modelo possui um aprendizado do tipo não-supervisionado. O modelo do ART teve várias versões posteriores, sendo que só será abordada neste trabalho a primeira versão.

Os estudos sobre as redes neurais sofreram uma grande evolução a partir dos anos 80, conforme foi demonstrado acima. E, a partir desta época, cada vez mais, esta área de estudos tem se destacado, seja pelas promissoras características apresentadas pelos modelos de redes neurais propostos ou pelas condições tecnológicas atuais de implementação de redes neurais em arquiteturas paralelas dedicadas.

4.3 Características e Aplicações das Redes Neurais

As redes neurais possuem a característica de serem muito apropriadas ao reconhecimento de padrões. Como já foi visto, as redes neurais podem sofrer um aprendizado, modificando seu comportamento frente a um conjunto de estímulos de entrada (padrão de entrada). Portanto, a rede pode aprender a dar uma resposta específica para um determinado conjunto de estímulos fornecidos. Isto será obtido através da alteração dos pesos de atuação das entradas.

Logo as redes neurais são muito adequadas para o reconhecimento e classificação de padrões, pois podem se adaptar para responder a um padrão específico.

As redes neurais possuem também uma alta velocidade de processamento, devido ao seu maciço paralelismo interno, possibilitando o desenvolvimento de certos tipos de aplicações complexas que necessitam operar em tempo real.

Devido às características inerentes às redes neurais, estas podem realizar alguns tipos de tarefas que não são executadas de uma forma satisfatória em sistemas computacionais tradicionais, mas que para o ser humano são tarefas triviais. Elas possuem a característica de se adequar perfeitamente às seguintes aplicações :

- Reconhecimento e síntese contínua da fala [KOH 88, WAI 89 e SEJ 86];
- Reconhecimento visual de padrões, OCR [LIP 87, BRO 87 e CAR 90] e padrões em geral [FUK 88 e REI 89]. Reconhecimento e classificação de imagens como, por exemplo: textos, assinaturas, impressões digitais, objetos, etc;
- Processamento adaptativo de sinais [WID 88] e eliminação de ruídos [DOW 89 e KLI 89];
- Aplicações onde os dados fornecidos são incompletos e os resultados produzidos são aproximados (*fuzzy*).

Além destas aplicações, também encontram-se referências de uso das redes neurais em outras aplicações como: memórias associativas, veículos autônomos, previsão do tempo, análise de finanças, gerenciamento de bases de dados e sistemas especialistas.

Apesar desta capacidade de realizar o reconhecimento de padrões de uma forma eficiente, as redes neurais, assim como os seres humanos, já não são tão rápidas e precisas no tratamento de dados numéricos.

O homem ao realizar cálculos aritméticos está sujeito a muitos erros possíveis, sendo também muito lento na realização de cálculos complexos e que necessitam de uma grande precisão. Estes cálculos exigem a capacidade de manipulação simbólica, sendo difícil a realização do mapeamento deste tipo de funções para uma rede de neurônios artificiais.

As redes neurais, devido as suas características, são utilizadas para trabalhar com dados aproximados, tanto de entrada como de saída. Um sistema baseado em redes neurais não pode ser considerado muito eficiente para aplicações como uma folha de pagamento. Por isso, o uso das redes neurais deve ser considerado como uma extensão para os sistemas tradicionais, por exemplo, com o uso de placas de redes neurais co-processadoras acopladas a estes. Estas placas podem servir para executar as tarefas que não são facilmente implementáveis em sistemas convencionais, como por exemplo, reconhecimento e síntese de fala; enquanto os sistemas convencionais continuam a realizar aquelas tarefas típicas onde obtêm um bom desempenho.

4.4 Modelos de Redes Neurais

As redes neurais trabalham basicamente sobre uma estrutura como a apresentada na figura 4.4. Este modelo de neurônio serve de base para os demais que serão apresentados nesta seção. Serão vistos apenas oito modelos, mas estes são os que possuem uma destacada importância em termos de histórico, evolução e aplicações na atualidade [LIP 87 e WID 90].

Existe uma série de outros modelos possíveis de serem implementados, mas que não serão abordados aqui porque isto tornaria este trabalho excessivamente extenso. A maioria destes modelos não abordados são variações dos modelos básicos apresentados neste capítulo ou modelos que foram desenvolvidos e não possuem uma grande repercussão junto aos principais grupos de pesquisa sobre redes neurais.

Alguns destes modelos levam em consideração, por exemplo, o tempo de propagação dos sinais ou tentam modelar curvas de resposta o mais similares possíveis do comportamento real dos neurônios. Na opinião do autor, alguns fatores, como os citados acima, podem ser desprezados no caso dos sistemas de reconhecimento visual, lembrando-se que um modelo de simulação é uma simplificação da realidade que permite, em termos funcionais, uma reprodução de um comportamento de um sistema real. A reprodução exata da realidade pode não ser a solução mais adequada para uma certa aplicação.

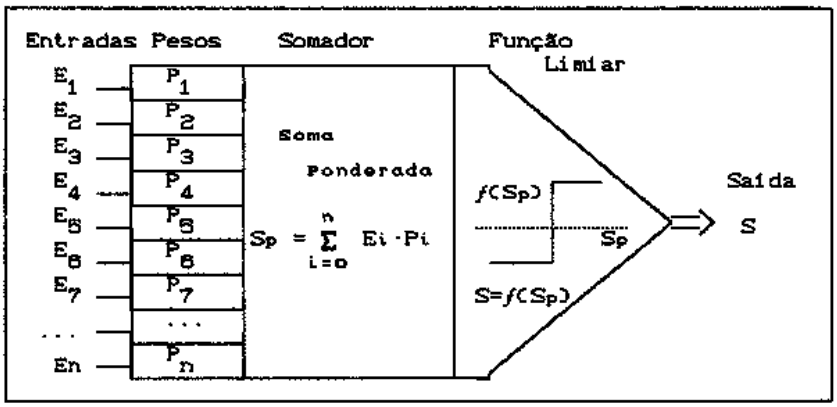


Figura 4.4 - Estrutura de um neurônio artificial

4.4.1 Classificação dos Modelos

Os diferentes modelos de redes neurais podem ser classificados em diversos grupos, conforme as suas características e propriedades. Estes grupos dividem as redes neurais em categorias de acordo com:

- *Treinamento*: auto-aprendizado (*self-learning, unsupervised learning*) ou aprendizado supervisionado (*supervised learning*) [LIP 87];

- *Regra de Aprendizado*: algoritmo competitivo (*competitive learning*) ou algoritmo adaptativo por correção de erros (*adaptive error correction*) [WID 90];

- *Interconexões dos Neurônios*: redes com neurônios sem realimentação (*feed-forward*) ou com realimentação (*feedback*) [LAW 90];

- *Organização da Rede*: um nível único ou em diversos níveis (*multi-layer*) [LIP 87];

- *Distribuição das Memórias*: localizada (*grand-mothering*) ou distribuída (*generalization*) [CAU 90];

- *Classificação*: separador linear ou separador não linear [LAW 90];

Os diferentes tipos de treinamento (auto-aprendizado ou aprendizado supervisionado) estão diretamente relacionados com estes mesmos dois tipos de treinamento utilizados em sistemas de reconhecimento de padrões, abordados no capítulo 2. Em resumo, o treinamento supervisionado consiste em apresentar à rede um padrão a ser reconhecido, juntamente com a resposta que a rede deve fornecer ao reconhecer novamente este mesmo padrão. O treinamento com auto-organização ou auto-aprendizado, consiste em apresentar apenas os padrões para a rede e esta se encarrega de agrupar aqueles que possuem propriedades similares (*clusterization*).

Uma importante categorização das redes neurais é em relação à regra de aprendizado, onde existem dois tipos bem distintos de redes neurais: as redes com regras de aprendizado competitivas e as redes baseadas na correção de erros. As redes neurais que se utilizam de regras de aprendizado do tipo competitivo são as redes de Hopfield e Kohonen, caracterizando-se pelas conexões laterais dos neurônios com os seus vizinhos e estabelecendo assim uma "competição entre os neurônios", que levará a rede a um estado estável. Os algoritmos de agrupamento (*clustering*) como o de Kohonen são exemplos típicos de redes com regra de aprendizado do tipo competitivo.

As redes neurais com aprendizado do tipo correção de erros são baseadas no princípio de adaptação e correção dos pesos de atuação de cada neurônio, até que este responda da maneira desejada. Este tipo de aprendizado é utilizado em redes baseadas em neurônios como o Adaline, utilizando-se de regras como a *Delta Rule*, que será apresentada posteriormente. A correção de erros está diretamente ligada ao aprendizado do tipo supervisionado ou tutorado.

Quanto à interconexão dos neurônios, costuma-se separar os modelos de redes neurais em redes do tipo *feed-forward* ou *feedback*. As redes do tipo *feed-forward* possuem neurônios cujas saídas conectam-se apenas com os níveis inferiores da rede, não havendo realimentação de suas entradas. Neste tipo de redes, a saída de um neurônio não influenciará a sua própria resposta. Em uma rede do tipo *feedback*, um neurônio pode se conectar a uma de suas próprias entradas ou à entrada de um neurônio em um nível superior da rede. Desta forma, a rede do tipo *feedback* possui neurônios cuja saída é capaz de influenciar de maneira direta ou indireta o seu próprio comportamento. Um exemplo típico desta rede é o modelo de Hopfield.

Uma rede pode ter uma topologia de interconexões, cuja organização seja em uma única camada (nível) ou em múltiplas camadas. As figuras 4.3 (a), (b) e (c) apresentam redes de um único nível e a figura 4.3 (d) apresenta uma rede do tipo multinível.

Uma rede neural forma memórias, ou seja, é capaz de guardar informações ou, no que se refere a este trabalho, a rede neural pode memorizar padrões. A memória de uma rede neural fica localizada nos pesos de atuação de suas entradas, que são responsáveis pelo seu aprendizado. Um único neurônio pode ser treinado para aprender um dado padrão, e desta forma, todos os pesos de suas entradas estão ligados a um único padrão, ocorrendo assim a criação de uma memória localizada (*grandmothering*). Um único neurônio pode não ser capaz de memorizar um padrão único, mas este

neurônio associado a vários outros neurônios pode ser capaz. Desta forma, a informação não estará localizada em um único neurônio, estando distribuída entre os diversos neurônios da rede neural. Sendo assim, os pesos de diferentes neurônios são responsáveis pela formação da memória e não se pode dizer que a informação está localizada em um único ponto da rede neural.

Um último tipo de distinção entre diferentes tipos de modelos de redes neurais é feito em relação à capacidade de classificação destes modelos. Os modelos que implementam um classificador linear são modelos onde basicamente se tem apenas um tipo de padrão associado a cada classe e desta forma ocorre uma associação linear de padrões. Um classificador não linear permite que tipos bastante distintos sejam associados a uma mesma classe. Este tipo de classificador será bem discutido posteriormente, uma vez que envolve o problema clássico do reconhecimento de um padrão que represente uma função XOR. Os classificadores lineares não são capazes de aprender uma função XOR, enquanto um classificador não linear terá a capacidade de aprender este tipo de função.

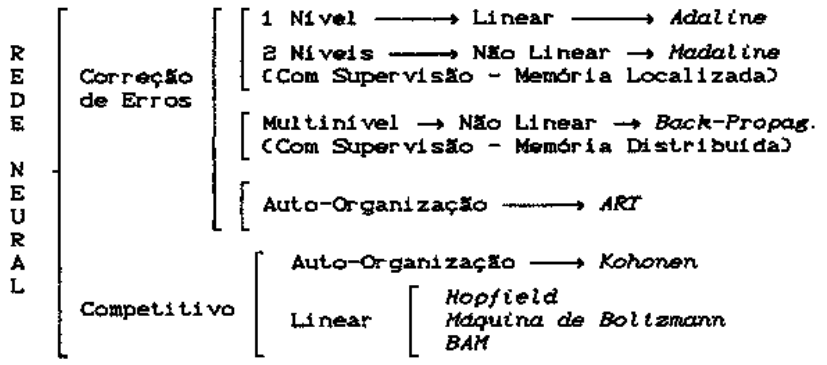


Figura 4.5 - Taxonomia das redes artificiais de neurônios

As categorias citadas acima servem para classificar diferentes tipos de modelos de redes neurais, mas estes modelos estão distribuídos de uma forma irregular nestas categorias. Devido a este fato, o esquema apresentado na figura 4.5 aborda apenas os itens de destaque na classificação de cada um dos modelos de redes neurais, não estando relacionadas todas as categorias.

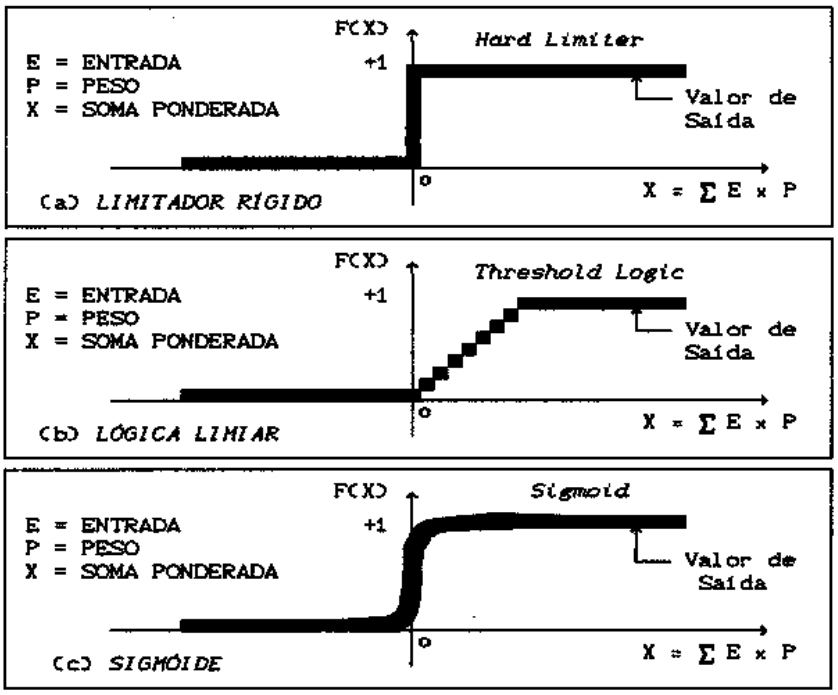


Figura 4.6 - Funções de transferência

4.4.2 Modelos Adaptativos

Os modelos de redes neurais adaptativos são modelos baseados no *Perceptron*. Estes modelos possuem em geral um aprendizado do tipo supervisionado, onde é aplicado um algoritmo de correção de erros.

4.4.2.1 Perceptron

O modelo básico do Perceptron foi idealizado por Frank Rosenblatt [ROS 59]. Este modelo possui muitas variações, onde será analisado aqui o modelo de apenas um nível, o mais simples possível. A estrutura dos neurônios se apresenta exatamente igual a da figura 4.4, sendo que sua função de transferência típica, é uma função do tipo *hard limiter* - figura 4.6 (a).

Este modelo consiste de uma rede linear de neurônios, onde os valores de entrada alimentam simultaneamente todos os neurônios. O neurônio que, após a sua ativação, possuir o maior valor de saída ou que atingir o limiar para a ativação indicará a resposta correta. Neste modelo cada neurônio é treinado para "responder" (identificar) a um padrão específico.

Será dada uma demonstração de como funciona uma rede de Perceptrons, através de um exemplo prático. Tomando-se um padrão de aprendizado o qual se deseja ensinar à rede, altera-se os pesos de um determinado neurônio que aprenderá este padrão. Os pesos são alterados de modo que, ao ativar a rede com um padrão de entrada similar ao do aprendizado, se obtenha um valor máximo na soma ponderada.

No exemplo da figura 4.7, tem-se uma entrada que é o padrão a ser reconhecido (linha superior), a qual ao ser fornecida a rede é multiplicada pelos pesos de cada um dos neurônios (linhas 1 a 5). Estes pesos refletem um treinamento previamente realizado. Para este padrão de entrada, o conjunto de pesos correpondentes a linha três fará com que a saída deste respectivo neurônio alcance um valor máximo e será portanto aquele que reconheceu o padrão de entrada apresentado. O padrão reconhecido é aquele que se parece mais com o padrão original para o qual este neurônio foi treinado. A função de transferência pode fazer com que apenas um neurônio seja ativado, aceitando um baixo grau de distorção no padrão de teste. Caso a função de transferência

seja menos rígida ou se for utilizada uma função do tipo *threshold logic*, pode ocorrer de mais de um neurônio ter sua saída ativa. Neste caso, é necessário selecionar o maior valor de ativação, onde no final só um neurônio deve se manter ativo. Isto pode ser feito usando-se uma rede tipo MAXNET [LIP 87] que seleciona o neurônio com a maior saída.

Padrão→		-1	+1	-1	+1	+1	+1	-1	-1	+1	-1	+1	Saída = $\sum$ E.P
Conjunto de pesos →	1	+1	-1	+1	-1	+1	+1	-1	+1	-1	+1	-1	Saída = -5
	2	-1	-1	-1	-1	-1	+1	+1	+1	+1	+1	+1	Saída = -1
	3	-1	+1	-1	+1	+1	+1	-1	-1	+1	-1	+1	Saída = +11
	4	+1	+1	+1	+1	+1	-1	-1	-1	-1	-1	-1	Saída = +1
	5	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	Saída = +1

Figura 4.7 - Exemplo de uso do Perceptron

Outro exemplo prático de aplicação é o emprego de uma rede de Perceptrons com a saída normalizada [KIN 89], o que facilita bastante a definição dos limiares e até o ajuste dos pesos de atuação pode ser feito de forma bastante simples. Nesta rede, os pesos são determinados da seguinte forma:

NE = Número total de elementos (pixels) do padrão,
P_{xK} = Peso da entrada x do neurônio K da rede,
PA_{ix} = Elemento x do padrão i de aprendizado e
Valores de entrada = -1 (apagado) ou +1 (ligado)

Para o neurônio K que será responsável pelo reconhecimento do padrão i, faz-se:

Se $PA_{ix} = +1$ então $P_{xK} = +1/NE$
senão $P_{xK} = -1/NE$

O ajuste de pesos descrito acima é realizado uma única vez, através de uma inicialização (não existe o procedimento de adaptação). Com o algoritmo acima o valor máximo de saída de um neurônio será +1, caso o padrão de teste seja exatamente igual ao padrão aprendido; e -1 caso o padrao de teste seja totalmente diferente do padrão para o qual este dado neurônio foi treinado.

O exemplo acima é interessante pois ele possui embutida, diretamente sobre o neurônio, a noção de similaridade de um padrão. Pode-se dizer que a saída da rede deste modelo, indica de uma forma imediata o número de pontos do padrão de teste que são exatamente iguais ao padrão aprendido. Outra vantagem é o fato da faixa de valores de saída ser extremamente bem definida, variando apenas entre -1 e +1, e dos pesos serem sempre iguais a uma constante com uma precisão pré-estabelecida, podendo valer apenas: -1/NE ou +1/NE.

Este modelo de rede tem uma arquitetura extremamente simples, como a da figura 4.3 (a), não correspondendo à organização real dos neurônios do sistema nervoso, o qual possui uma alta interconetividade entre seus componentes. Este modelo é excessivamente simples e possui uma grande limitação: não possui a capacidade de generalização! Logo, estas redes neurais ficarão enormes pois cada neurônio aprende apenas um único padrão.

Partindo-se do modelo básico do Perceptron, pode-se chegar a outros modelos mais completos, que possibilitarão um melhor aprendizado. Um destes modelos é o do Adaline.

4.4.2.2 Adaline

Este modelo é uma alteração do Perceptron, onde utiliza-se a *Regra Delta* de aprendizado (*Delta Rule*), ou como também é conhecido, algoritmo de aprendizado LMS (*Least Mean Square*). O Adaline foi criado por Widrow & Hoff

[WID 62] e seu nome vem da abreviação do nome dado aos elementos processadores : ADAptive LINear Element. A arquitetura é de uma rede de uma única camada disposta linearmente com cada neurônio possuindo uma *retro-alimentação* de sua própria saída. A figura 4.8 apresenta o esquema do modelo de neurônio utilizado no Adaline.

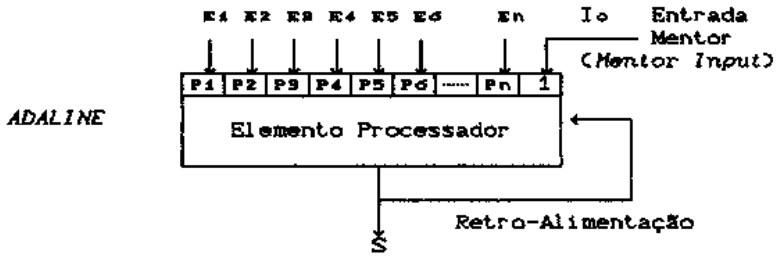


Figura 4.8 - Esquema do neurônio tipo Adaline

O Adaline é um neurônio que utiliza uma regra de aprendizado que permite uma adaptação, fazendo com que este se ajuste a novos padrões. O aprendizado é do tipo supervisionado, por correção de erro, através do uso de uma rede com retro-alimentação de apenas um nível. Este tipo de rede realiza uma classificação linear dos padrões.

O Adaline tem uma entrada denominada de *mentor input*, a qual possui uma ponderação sempre igual a 1. Esta entrada apresenta a verdadeira resposta que deveria ser dada por um determinado neurônio diante de um determinado conjunto de estímulos (padrão de teste).

A Regra Delta altera o vetor de pesos de atuação de forma que, para um determinado conjunto de entrada, a saída se aproxime da resposta desejada, indicada através da *mentor input*. Isto é obtido através da minimização do erro médio quadrático (erro LMS), que com o uso da Regra Delta é feito da maneira mais eficiente possível [CAU 90]. A Regra Delta nada mais é do que a regra de otimização de funções empregando-se o método do gradiente [CAR 90].

A figura 4.9 descreve em detalhes o funcionamento da Regra Delta de aprendizado que é utilizada pelo Adaline. Esta regra é muito importante, sendo empregada inclusive em modelos sucessores do Adaline.

$$E_R = I_O - S$$

S → Saída Atual da Rede
I_O → Saída Desejada (*Mentor Input*)
E_R → Erro Estimado

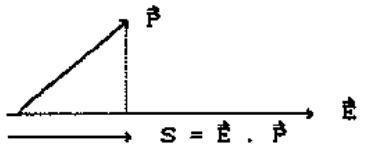
Regra Delta: $$\vec{P}_{NOVO} = \vec{P}_{ANTIGO} + \frac{\beta * E_R * \vec{E}}{|\vec{E}|}$$

Onde : β = Fator de Ajuste de Convergência (0 à 1)
E_R = Erro Estimado
$\vec{E}$ = Vetor de Valores de Entrada
$\vec{P}$ = Vetor de Pesos

Figura 4.9 - Algoritmo de aprendizado *Delta Rule*

A representação dos valores de entradas e pesos de atuação através de vetores possibilita obter uma interpretação geométrica e matemática mais clara do processo de aprendizado. Abaixo é apresentada esta interpretação para um vetor de duas componentes apenas. Estes conceitos são também válidos para *N* componentes, mas neste caso se terá um espaço *N*-dimensional ao invés de um plano.

Neurônio : 2 entradas
Vetor de Entrada : E1, E2 ($\vec{E}$)
Vetor de Pesos : P1, P2 ($\vec{P}$)
Soma Ponderada : $S = \vec{E} \cdot \vec{P}$



- A soma ponderada é o produto interno entre os vetores $\vec{E}$ e $\vec{P}$.

- O produto interno entre dois vetores tem como resultado a projeção de um vetor sobre o outro. Quanto maior for a projeção (valor de saída da soma ponderada), mais próximos estão os dois vetores. Caso os vetores sejam ortogonais, a projeção será igual a 0 (zero).

- O aprendizado deve deslocar o vetor de pesos de forma a aproximá-lo do vetor de entradas que representa o padrão a ser ensinado a um dado neurônio.

- O reconhecimento consiste em identificar qual o vetor de pesos, de um dado neurônio, que está mais próximo do vetor de entrada de teste.

Pode-se considerar também os vetores de pesos como variáveis de uma equação da reta do tipo:

$$S = E1 \,.\, P1 + E2 \,.\, P2 \longleftrightarrow C = A \,.\, X + B \,.\, Y$$

Esta equação se transforma em uma inequação devido ao uso da função de transferência. Desta forma o plano, denominado de espaço de decisão, é dividido em dois semi-planos no caso de equações a duas variáveis, ou em dois semi-hiperplanos para *N* variáveis.

Se for considerado o exemplo dado abaixo, pode-se visualizar claramente os conceitos que foram descritos anteriormente.

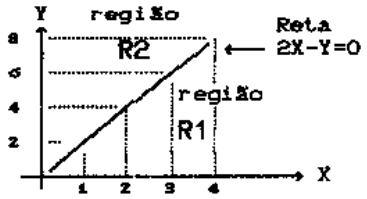
$S \leq P_1E_1 + P_2E_2$ → É uma inequação que divide o plano em duas sub-regiões

$C \leq A.X + B.Y$

A,B → São os pesos

X,Y → São as entradas.

X e Y estão localizados em uma das duas sub-regiões formadas



Supondo-se :

$P1 = A = 2$ Vem ... $0 \leq 2 . X - 1 . Y$ (Região R1)
$P2 = B = -1$ $0 > 2 . X - 1 . Y$ (Região R2)

Observação : o limiar adotado é zero - 0 (*hard limiter*)

Portanto :

Se a soma ponderada S for: $S \geq 0 \rightarrow$ Padrão na região R1
$S < 0 \rightarrow$ Padrão na região R2

Na interpretação dada acima, percebe-se que um dado conjunto de pesos de atuação de um neurônio terá como função dividir o espaço de decisão em duas sub-regiões. Cada neurônio se ativará de acordo com a posição do padrão de entrada dentro do espaço de decisão, se ativando ou não, conforme a região onde se encontra este padrão.

Esta divisão do espaço de decisão em duas sub-regiões bem definidas e isoladas é o que determina a característica de classificação linear dos modelos baseados no Adaline. Este tipo de interpretação também serve para que se possa compreender porque o Adaline não é capaz de aprender os padrões que representam a função XOR.

4.4.2.3 Madaline

Uma extensão proposta por Widrow ao modelo do Adaline é a da criação de um novo elemento, chamado de Madaline. O Adaline é capaz de aprender a reconhecer apenas um padrão simples e, a partir deste Adaline, surgiu a idéia de criar um elemento composto, chamado Madaline (*Many Adaline*). O Madaline apenas agrupa a resposta de dois ou mais Adalines, realizando um "e lógico" (AND), "ou lógico" (OR) ou pela obtenção de maioria (*Majority Voteker*) dos valores obtidos nas saídas de um conjunto de neurônios do tipo Adaline [WID 88]. Com isto, é possível fazer com que

uma saída da rede seja ativada para mais de um tipo de padrão de entrada. Assim, é possível treinar a rede de Madaline para reconhecer, por exemplo, um padrão transladado ou até rotacionado. Para isto, basta treinar cada elemento interno do Madaline para reconhecer o padrão em uma dada posição ou orientação e fazer o "ou" das saídas destes elementos [CAU 90].

Os modelos de um único nível baseados no Perceptron sofreram um grande abalo após a publicação de Minsky [MIN 69] que demonstrou uma grande limitação na sua capacidade de aprendizado.

A pergunta crítica feita por Mynsky é: como uma rede de neurônios de um único nível, pode aprender a reconhecer o padrão de uma função lógica "ou exclusivo" (XOR)? O problema das redes de um nível baseadas no Perceptron é que elas dividem o espaço de soluções em duas regiões distintas, mas em uma função XOR não se consegue abranger as duas respostas em uma só região. O diagrama da figura 4.10 apresenta a relação entre diferentes modelos de redes e a formação das regiões de decisão.

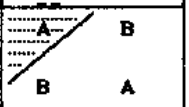
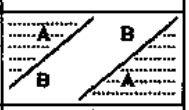
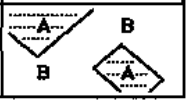
TIPO REDE	TIPOS DE REGIOES	XOR
1 Nível	Criação de um plano ou hiper-plano de divisão	
2 Níveis	Regiões convexas abertas ou fecha-das	
3 Níveis	Regiões arbitrá-rias de formato livre	

Figura 4.10 - Níveis de uma rede *versus* regiões de decisão

O modelo anterior, o Adaline, que é uma rede de um nível, não pode resolver o problema de um XOR.

Será dado aqui uma breve demonstração do motivo que levou a procura de uma nova solução, chegando-se então às redes neurais de mais de um nível, como o Madaline. Pode-se comparar uma rede de um nível com um sistema de equações lineares onde, se for considerado um neurônio com duas entradas, obtem-se:

$$\text{Saída} = P_1 \cdot E_1 + P_2 \cdot E_2 \longrightarrow \begin{array}{l} P = \text{Vetor de Pesos} \\ E = \text{Vetor de Entradas} \end{array}$$

Para a função XOR, as equações obtidas serão:

E1	E2	S
-1	-1	-1
-1	+1	+1
+1	-1	+1
+1	+1	-1

Equações : P1, P2 = Pesos

$$\left.\begin{array}{l} -P1 - P2 = -1 \\ -P1 + P2 = 1 \\ P1 - P2 = 1 \\ P1 + P2 = -1 \end{array}\right] \text{SISTEMA INSOLÚVEL}$$

Portanto em uma rede de um único nível não há um conjunto de pesos que possibilite a solução do problema. A idéia é se criar um nível a mais que possibilite a união das respostas do nível superior. A solução do problema do XOR em uma rede de dois níveis é imediata:

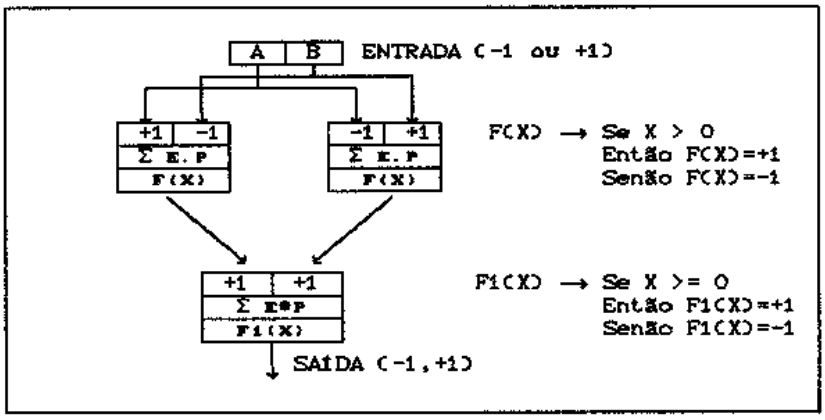


Figura 4.11 - Solução do problema do XOR

Uma rede de mais de um nível poderá aprender inclusive padrões como o da função XOR. O Madaline, que é uma rede neural de dois níveis (como a da figura 4.11), resolve em parte o problema, mas ainda apresenta um inconveniente, o usuário é quem realiza a "junção" dos Adalines.

Até então, não havia um algoritmo que fizesse o aprendizado automaticamente de uma forma completa, faltando uma regra de aprendizado mais genérica para as redes de múltiplos níveis. Além disto, no Adaline e mesmo no Madaline ainda não se tem um conhecimento distribuído na rede, sendo que cada neurônio tem que ser treinado para reconhecer um dado padrão. O Madaline apenas possibilita a junção de dois Adalines de forma a "aparentemente" fazer com que quaisquer dois padrões possam ser identificados como pertencentes a uma mesma classe.

4.4.2.4 Redes Multinível

A técnica de *Back-Propagation* conseguiu dar uma nova vida às redes neurais, pois permite o aprendizado das redes neurais baseadas no Perceptron, com qualquer número de níveis, servindo inclusive para o caso do XOR. Este algoritmo de aprendizado foi desenvolvido por Parker [PAR 85] e Rumelhart [RUM 86].

Uma rede neural que utiliza o modelo de *Back-Propagation* é uma rede com neurônios dispostos em três ou mais níveis: um nível de entrada (*input layer*), um ou mais níveis intermediários ou ocultos (*hidden layers*) e um nível de saída (*output layer*) - vide figura 4.3 (d). A função de transferência utilizada é uma função do tipo função sigmóide (tem a forma de um 'S') e a regra de aprendizado é derivada da Regra Delta, mas aplicável a uma rede multinível. A seguir são apresentadas as funções utilizadas nas redes multinível.

* Ajuste dos Pesos :

Nível de Entrada : Todas as entradas possuem pesos igual a 1.

Nível de Saída : Adaptação dos pesos através da Regra Delta original.

Nível Intermediário :

- Calcula-se o ERRO de cada neurônio do nível intermediário;
- Aplica-se o ERRO (calculado conforme descrição abaixo) na equação principal da Regra Delta;
- Usa-se como função de transferência a função sigmoidal.

* Cálculo do Erro :

$$ERRO_i = F'(E) \cdot \left\{ \sum P_{ij} \cdot ERRO_j \right\}$$

→ Onde :

F'(E) - Derivada da função de ativação para a entrada E

i - Número do neurônio no nível intermediário

j - Número do neurônio no nível de saída

* Função de Transferência :

$$F(X) = \frac{1}{(1 + e^{-(X + T)})}$$

$$F'(X) = F(X) \cdot (1 - F(X))$$

→ Onde :

T - Limiar simples da função de transferência (Constante definida pelo projetista da rede)

X - Valor de entrada da função de transferência

e - Número de Euler (constante)

O aprendizado por *back-propagation* baseia-se na propagação retrógrada do erro para os níveis anteriores da rede, de acordo com o grau de participação que cada neurônio teve no erro do nível posterior. O modelo multinível utilizado normalmente é o composto por três níveis, onde o número de neurônios do nível de entrada e de saída podem ser estimados de acordo com as características do problema. Já o número de neurônios do nível intermediário, assim como o número de níveis intermediários tem que ser estabelecidos de uma maneira intuitiva, não havendo uma técnica precisa para a estimativa destes dados [CAU 90]. Isto se deve ao fato de que neste modelo há uma generalização do conhecimento, onde agora não existe mais um neurônio encarregado de aprender especificamente um padrão e, desta forma, o número de neurônios da rede não pode ser estabelecido de forma exata. Se o nível intermediário tiver um pequeno número de neurônios, a rede pode não ter capacidade de memorização para todos os padrões e, se o nível intermediário tiver muitos neurônios, pode ocorrer o efeito de *grandmothering*, ou seja, a criação de neurônios especializados para reconhecer cada padrão.

As redes neurais do tipo multinível com *back-propagation* são um dos modelos mais estudados e adotados na atualidade pela sua grande capacidade de adaptação a qualquer tipo de padrão. É claro que o custo em termos de implementação é um pouco mais elevado que o Adaline mas, dependendo da aplicação e principalmente do tipo de padrões a serem reconhecidos ou classificados, pode ser vantajosa a utilização deste modelo.

4.4.2.5 ART

O modelo ART ou teoria da ressonância adaptativa (*Adaptive Resonance Theory*) foi desenvolvido por Carpenter e Grossberg [CAR 83]. Este modelo implementa uma rede neural, a qual é treinada sem uma supervisão, formando agrupamentos

(*clusters*) e podendo ser completamente descrita usando equações diferenciais não lineares [LIP 87].

O modelo ART é basicamente um modelo onde é desenvolvida uma técnica que permite uma evolução da rede, fazendo com que esta possa "crescer" à medida que novos padrões lhe são ensinados. O algoritmo principal seleciona a primeira entrada como um exemplar para o primeiro *cluster*. A próxima entrada é comparada com este primeiro *cluster* onde, caso a distância do novo exemplar seja menor que um certo limite, faz-se apenas uma adaptação de forma que o *cluster* represente ambos os padrões. Caso a distância entre o novo exemplar e o exemplar já armazenado seja superior ao limite estabelecido, é criado um novo *cluster* que servirá para representar este novo exemplar. Este processo é repetido para todas as entradas seguintes. O número de *clusters* crescerá com o tempo e dependerá tanto do limite como da métrica usado no cálculo da distância na comparação das entradas com os exemplares armazenados na rede.

O modelo ART possui uma propriedade muito interessante, que é a possibilidade de expansão do número de padrões aprendidos pela rede. Através da criação de novos *clusters*, este modelo de rede neural pode ir se expandindo à medida que seja necessário acrescentar novos exemplares. É interessante notar que o limite de comparação de distância entre o exemplar de entrada e os exemplares armazenados pode ser fixado de acordo com cada aplicação. Este limite determina a flexibilidade do reconhecimento, ou seja, um valor muito pequeno indica que os exemplares serão muito parecidos para serem colocados juntos, representando um único *cluster*. Mas, quando o limite é um valor maior, haverá uma maior flexibilização e os exemplares de entrada serão agrupados em um conjunto menor de *clusters*.

O algoritmo completo do comportamento de um rede do tipo ART é descrito por Lippmann [LIP 87]. A figura 4.12 apresenta um exemplo do comportamento de uma rede do modelo ART para o algoritmo descrito por Lippmann [LIP 87].

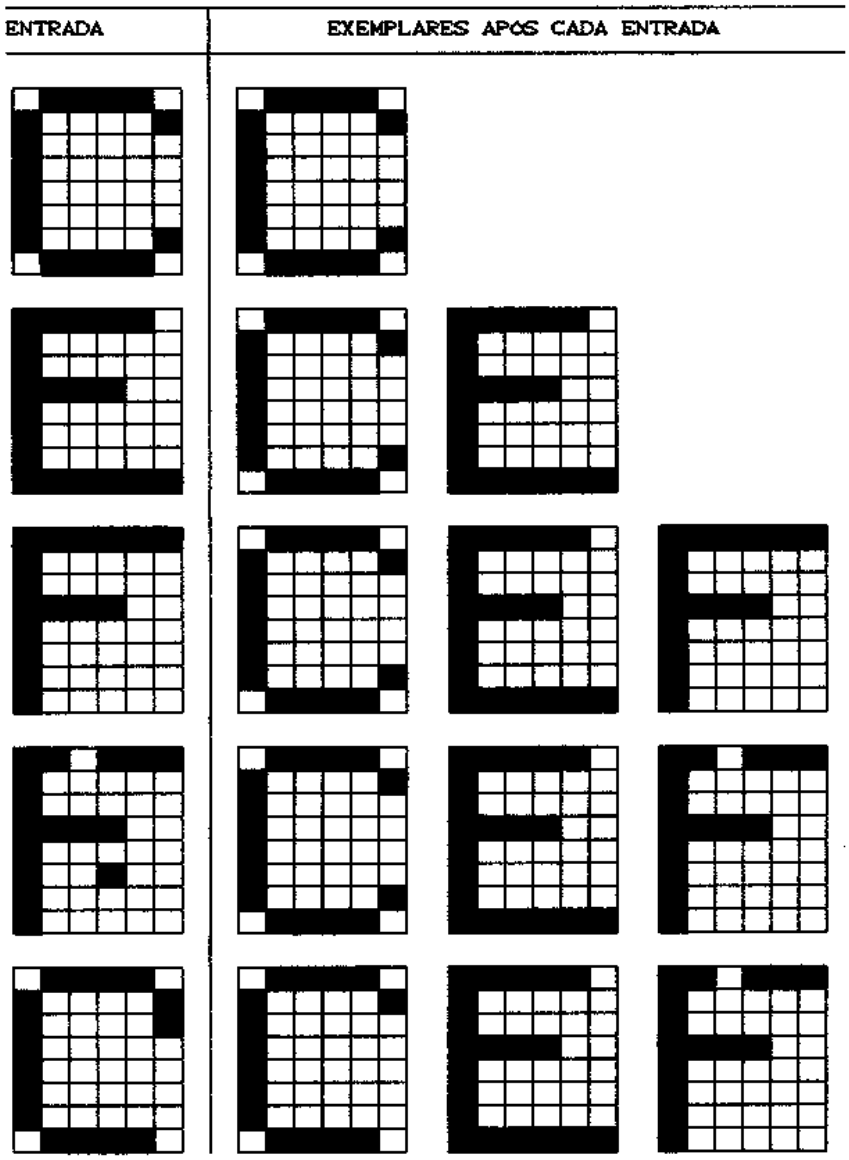


Figura 4.12 - Funcionamento do modelo ART

O modelo ART sofreu várias evoluções, sendo que existem diversos modelos, denominados de ART-1, ART-2 e ART-3 [WID 90]. Estes modelos no entanto implicam uma sobrecarga de trabalho muito pesada em simulações, sendo considerados mais como elementos de pesquisa do que modelos de aplicação prática [CAU 90].

4.4.3 Modelos Competitivos

Os modelos competitivos baseiam-se na competição entre os neurônios, onde existem modelos com alguma das seguintes características: conexões laterais entre neurônios vizinhos, formação de *clusters* ou auto-organização. Nesta classe de modelos de rede neurais, encontra-se os modelos de Hopfield, máquina de Boltzmann, BAM e modelo de Kohonen.

4.4.3.1 Modelo de Hopfield

O modelo original de Hopfield [HOP 82] surge utilizando uma rede onde se dá destaque ao comportamento coletivo. O conjunto possui mais importância do que os elementos, estando os neurônios totalmente interconectados entre si. Um neurônio só não conecta a sua saída às suas próprias entradas.

O modelo de Hopfield é utilizado como uma memória associativa ou para resolver problemas de otimização. Este modelo é estudado em Física, onde se reduz ao problema do vidro de *spin* infinito numa rede de interconexão total [PEN 90]. O modelo de Hopfield é muito empregado na restauração de padrões, sendo que ele trabalha com valores de entrada e saída binários (+1,-1). A figura 4.13 apresenta a topologia de uma rede de Hopfield, a qual pode utilizar a regra de aprendizado somente uma vez para inicializar a rede - figura 4.16. A função de transferência empregada é do tipo *hard limiter*, conforme é apresentada na figura 4.6 (a).

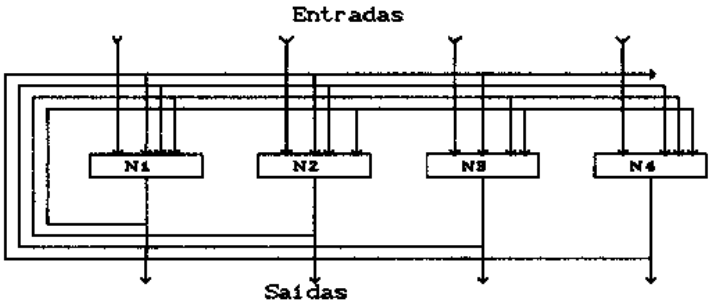


Figura 4.13 - Organização de uma rede de Hopfield

A rede de Hopfield é utilizada para o reconhecimento de padrões como : caracteres, objetos e voz. A figura 4.14 apresenta o exemplo de uso do modelo de Hopfield.

Figura 4.14 - Utilização da rede de Hopfield

O funcionamento do modelo proposto por Hopfield é o seguinte:

- Cada neurônio é responsável por um elemento (pixel) do padrão de entrada, podendo assumir somente dois estados (+1 = Ligado) ou (-1 = Desligado);

- Ao aprender um padrão, o neurônio se comportará de forma que sua saída ajude a ativar algum outro neurônio que faça parte do padrão aprendido e que deveria estar ativo; ou inibir algum outro neurônio que faça parte do padrão e que deveria estar inativo (figura 4.15);

- A rede será ativada e a medida que os neurônios vão interagindo uns com os outros, a saída da rede vai convergindo para um estado estável, que deve ser um dos padrões aprendidos;

- Para adicionar novos padrões (novas memórias - estados estáveis) para os quais a rede deve convergir, basta adicionar os pesos já existentes aos valores novos obtidos através do mesmo cálculo utilizado na inicialização.

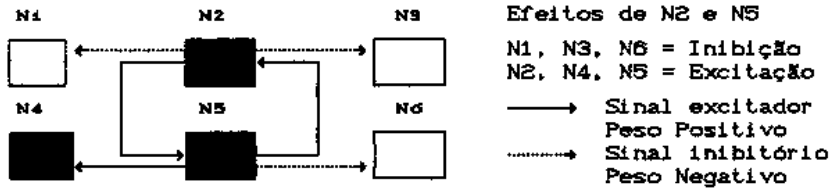


Figura 4.15 - Interação entre os neurônios no modelo de Hopfield

O modelo de Hopfield possui algumas limitações. Uma delas é quanto ao tamanho da rede em termos do número de neurônios, o qual deve respeitar a equação [HOP 82]:

Nro. de neurônios x 0,15 > Nro. de padrões armazenados

Devido a esta restrição, as redes deste modelo tem um tamanho muito grande. Por exemplo, para armazenar 10 padrões é preciso de mais de 70 neurônios com um total de aproximadamente 5000 conexões (pesos de atuação). Caso esta limitação não seja respeitada, a rede pode convergir para os chamados "*estados espúrios*", que são estados imperfeitos, uma mistura entre os padrões aprendidos.

Outro problema é o aprendizado de padrões muito similares que também podem levar à obtenção de um estado espúrio ou podem convergir para um padrão incorreto. A solução neste caso é realizar alguns procedimentos de ortogonalização [WAL 88], visando diferenciar mais os padrões.

Além do modelo de Hopfield, existem dois outros modelo que são muito parecidos a este, atuando também como memórias associativas, são eles: a máquina de Boltzmann e o modelo BAM.

Aprendizado/Inicialização

$$P_{ij} = \begin{cases} \sum_{p=0}^{M-1} E_i^p E_j^p \\ 0 \text{ se } i=j \end{cases}$$

Onde :

P_{ij} = Peso p/entrada i do neurônio j

p = Padrões
Existem M padrões

E_i^p = Elemento de entrada i do padrão p

Reconhecimento

1) $\mu(0)$ = Padrão a ser reconhecido

$\mu_i(0) = E_i^{Teste}$ para $0 \le i \le N-1$

2) $\mu_j(t+1) = F\left(\sum_{i=0}^{N-1} P_{ij} \; \mu_i(t)\right)$ para $0 \le j \le N-1$

3) Repetir o passo 2 até que o padrão de saída $\mu(t)$ fique estável

Onde :

N = Nro. de elementos do padrão de teste (entradas)

$\mu_i(t)$ = Saída do neurônio i no tempo t

$t \longrightarrow$ Número da iteração

F(X) = Função de Transferência

Figura 4.16 - Algoritmos da rede de Hopfield

4.4.3.2 Máquina de Boltzmann

O modelo da máquina de Boltzmann foi desenvolvido por Hinton, Sejnowski e Ackley [HIN 84, HIN 86]. Este modelo é composto por uma rede neural totalmente conectada, que possui uma representação distribuída do conhecimento, sendo formado por uma rede de elementos de processamento estocástico. A máquina de Boltzmann pode ser considerada, de uma maneira simplificada, como uma rede do modelo de Hopfield, que estabelece as soluções por um processo de *simulated annealing* governado pela estatística de Boltzmann [WID 90].

O *simulated annealing* (resfriamento simulado) [KIR 83] é uma técnica de pesquisa que tem sido aplicada a diversos problemas de otimização. Este tipo de mecanismo é empregado, de forma a tentar evitar os chamados *estados espúrios* do modelo de Hopfield. O princípio para o uso desta técnica é a adição de um componente aleatório ao processo de decisão de cada neurônio, de forma a tentar escapar dos estados espúrios, também denominados de mínimos locais de energia [BAL 89].

Este modelo segue, de uma forma geral, os princípios de funcionamento do modelo de Hopfield. Por este motivo e, devido ao fato deste tipo de rede sofrer problemas de perda de precisão, não serão apresentados maiores detalhes sobre o mesmo. Para maiores informações, sugere-se a leitura de [HIN 84, HIN 86, ACK 83 e BAL 89].

4.4.3.3 BAM

A BAM - memória associativa bidirecional (*Bidirectional Associative Memory*), foi desenvolvida por Kosko [KOS 87], baseada em algumas idéias de Hopfield e Grossberg. Esta rede serve como uma memória associativa capaz de associar pares de padrões. Ao ser apresentado um dos componentes de algum destes pares de padrões, a rede gerará automaticamente na saída a representação do outro padrão associado ao padrão apresentado.

A rede BAM, assim como a rede de Hopfield, se utiliza de uma técnica de minimização de energia. A diferença é que, no modelo de Hopfield, um padrão é associado a ele mesmo e, portanto, na saída, o padrão é apresentado apenas reconstituído. Na BAM, um padrão pode ser associado a um outro padrão e, desta forma, ao apresentar algum destes dois padrões associados, mesmo distorcidos, a rede apresentará na saída o padrão associado ao apresentado na entrada da rede.

O modelo BAM associa pares de padrões (A_i, B_i), onde $A_i \in \{-1, +1\}^N$ e $B_i \in \{-1, +1\}^P$. Para estes padrões A e B, compostos por vetores de dimensões N e P respectivamente, será necessária uma matriz de pesos M de dimensões N x P para codificar tais padrões.

A codificação, que é a forma como se denomina o aprendizado em uma rede BAM, é realizada simplesmente através do somatório das matrizes de correlação de cada um dos pares de padrões [BLU 90]. Desta forma, a matriz M que codifica K pares de padrões é simplesmente:

$$M = \sum_{i=1}^{K} A_i^T \; B_i$$

K = Nro. de pares de padrões
M = Matriz de pesos
A_i^T = Trasnposta do vetor A de padrões
B_i = Vetor B de padrões

O reconhecimento é denominado de *recall* ou decodificação. Ele consiste apenas da multiplicação do vetor de teste (padrão de teste - codificado em valores +1, -1) pela matriz M, seguida da aplicação de uma função de transferência, resultando um outro vetor. Este vetor resultante será o padrão associado ao vetor de teste. A rede é dita bidirecional, pois A.M resultará no padrão B, assim como $B.M^T$ resultará no padrão A. Outra caraterística interessante deste modelo é a possibilidade de retirar um par de padrões da memória, "esquecendo-os". A figura 4.17 apresenta os processos de codificação e decodificação.

A BAM, assim como o modelo de Hopfield, também possui problemas de aprendizado em relação ao número máximo de padrões aprendidos e de reconhecimento incorreto de padrões similares (necessidade de ortogonalização da entradas). O limite de memorização em uma BAM será da ordem de M onde M é o menor valor entre as dimensões N e P dos pares de padrões codificados na rede [KOS 87a].

Codificar:

$X_1 = (101010)$ e seu par $Y_1 = (1100)$

$X_2 = (111000)$ e seu par $Y_2 = (1010)$

Recodificação dos Padrões: $(0,1 \rightarrow -1,+1)$

$A_1 = (1\ -1\ \ 1\ -1\ \ 1\ -1)$ $\qquad B_1 = (1\ \ 1\ -1\ -1)$

$A_2 = (1\ \ 1\ \ 1\ -1\ -1\ -1)$ $\qquad B_2 = (1\ -1\ \ 1\ -1)$

Matrizes:

$$A_1^T B_1 = \begin{bmatrix} 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 \end{bmatrix} \qquad A_2^T B_2 = \begin{bmatrix} 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 \\ -1 & 1 & -1 & 1 \\ -1 & 1 & -1 & 1 \end{bmatrix}$$

$$M = \begin{bmatrix} 2 & 0 & 0 & -2 \\ 0 & -2 & 2 & 0 \\ 2 & 0 & 0 & -2 \\ -2 & 0 & 0 & 2 \\ 0 & 2 & -2 & 0 \\ -2 & 0 & 0 & 2 \end{bmatrix}$$

Função de Transferência:

$F(X) = -1$ se $X < 0$

$F(X) = 1$ se $X >= 0$

$A_1 \,.\, M = (8\ \ 4\ -4\ -8) \rightarrow F(X) \rightarrow (1\ \ 1\ -1\ -1)$

$A_2 \,.\, M = (8\ -4\ \ 4\ -8) \rightarrow F(X) \rightarrow (1\ -1\ \ 1\ -1)$

$B_1 . M^T = (4\ -4\ \ 4\ -4\ \ 4\ -4) \longrightarrow (1\ -1\ \ 1\ -1\ \ 1\ -1)$

$B_2 . M^T = (4\ \ 4\ \ 4\ -4\ -4\ -4) \longrightarrow (1\ \ 1\ \ 1\ -1\ -1\ -1)$

Figura 4.17 - Processo de codificação em uma BAM

Uma solução para problemas de limitação de aprendizado em uma BAM, é a criação dos sistemas BAM [BLU 90].

Um sistema BAM é composto por várias redes BAM, mas a sua grande vantagem é a possibilidade, ao se determinar que uma dada rede não consegue aprender um novo padrão, de codificar-se este padrão em outra rede. Para determinar se a rede pode ou não aprender um padrão, basta codificá-lo na matriz M e, em seguida, testar para ver se a rede responde corretamente à apresentação deste padrão. Caso a rede não se comporte da maneira esperada, o padrão é removido da rede e codificado em uma outra rede, que tenha a capacidade de armazená-lo corretamente.

4.4.3.4 Modelo de Kohonen

Este modelo recebe o nome de seu criador Teuvo Kohonen [KOH 84]. Sendo um modelo baseado no auto-aprendizado, ou seja, a rede neural do modelo de Kohonen possui a capacidade de auto-organização. Este modelo também é conhecido pela criação dos mapas de atributos de auto-organização (*Self-Organization Feature Maps*). A rede neural utilizada no modelo básico de Kohonen é uma rede de um único nível, onde os neurônios possuem conexões com as entradas e conexões laterais com outros neurônios, até um certo grau de vizinhança pré-estabelecido. As entradas externas estão conectadas em paralelo a todos os neurônios simultaneamente. A figura 4.18 apresenta um esquema de organização do modelo de Kohonen para neurônios conectados apenas aos seus oito vizinhos mais imediatos.

A propriedade de auto-aprendizado é muito interessante, pois reproduz o aprendizado humano de certas funções, de uma maneira muito natural. Considerando-se um recém-nascido com exemplo, encontra-se um auto-aprendizado na função de foco dos olhos, a qual é adquirida de forma automática, sem a necessidade de uma "supervisão do aprendizado". Outra característica muito interessante do modelo de Kohonen é a formação de regiões da rede que tem seus neurônios ativados quando determinado padrão é apresentado.

Cada padrão provocará a ativação de uma região determinada da rede, onde a identificação de um dado padrão se dá através da determinação de qual área de rede foi ativada. Estudos neurofisiológicos demonstram que o cérebro humano atua de uma forma similar a esta, onde regiões diferentes do cérebro são ativadas para cada tipo de estímulo externo recebido.

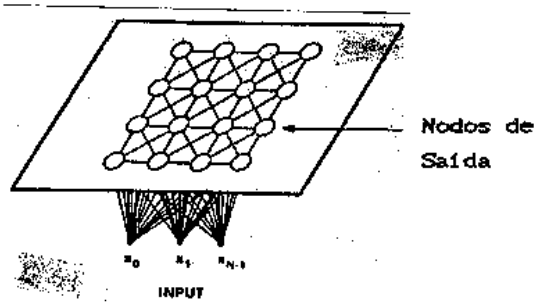


Figura 4.18 - Organização de uma rede neural de Kohonen

Esse fenômeno torna a rede um mapa de classificação, ou seja, para um determinado vetor de entrada, uma determinada região do mapa bidimensional será ativada. Além disso, os pesos serão organizados de forma que neurônios topologicamente próximos sejam sensíveis a entradas fisicamente similares. A figura 4.19 apresenta um exemplo de saída de ativação de uma rede de Kohonen, onde os pontos correspondem aos neurônios e seus tamanhos correspondem à sua atividade (valor de ativação na saída). Na figura da direita é apresentado o resultado da lenta e gradual mudança das entradas, e o movimento do aglomerado é indicado pela sua "cauda".

O tamanho do aglomerado também pode ser ajustado através da determinação do grau de interconexões de um neurônio, em relação aos seus vizinhos. Pode-se dizer então que o raio do aglomerado é proporcional à extensão da vizinhança dos neurônios. Também foi descoberto

empiricamente que, para bons resultados de auto-organização da rede neural, o raio do aglomerado deve decrescer monotonicamente com o decorrer do tempo. A figura 4.20 demonstra exemplos de vizinhança possíveis de serem adotados, bem como a possível evolução da rede em termos de grau de vizinhança das conexões.

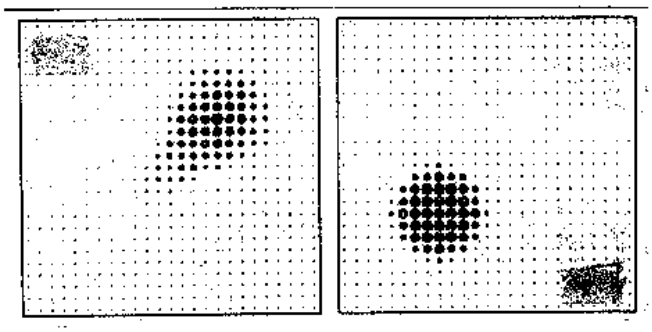


Figura 4.19 - Agrupamento da atividade das saídas de uma matriz bidimensional de neurônios

O algoritmo de ajuste dos pesos e adaptação da rede (algoritmo de *clustering*) pode ser encontrado na literatura [LIP 87, BAL 89 e KOH 84]. A rede de Kohonen constitui-se uma interessante ferramenta para reconhecimento de padrões, sendo inclusive muito empregada em sistemas de reconhecimento de fala, como o citado por Kohonen [KOH 88]. Uma única desvantagem da aplicação deste tipo de modelo no reconhecimento de padrões é a necessidade de identificação das regiões de ativação, ou seja, existe a necessidade de posteriormente associar cada região de ativação a um dado padrão.

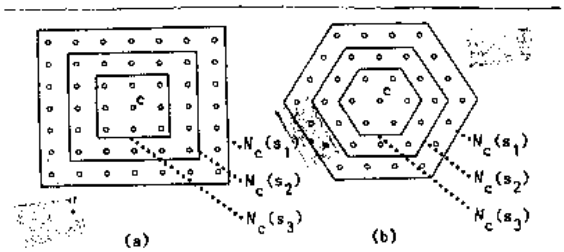


Figura 4.20 - Exemplos de topologias de vizinhança

4.5 Aplicações das Redes Neurais Junto aos Sistemas OCR

As redes neurais tem sido muito empregadas junto ao reconhecimento de padrões e principalmente em sistemas OCR. Uma vez que a entrada de uma rede já se constitui de um padrão, do tipo de um *bit-map*, e sua saída representa a identificação, classificação ou associação de um padrão a outro, logo, é natural que as redes neurais sejam utilizadas em sistemas OCR. Em quase todos os artigos e exemplos de aplicações sobre as redes neurais encontra-se sempre uma demonstração de seu comportamento junto ao reconhecimento de caracteres.

Outro motivo que leva a sua utilização junto aos sistemas de OCR é devido ao maciço paralelismo embutido nas redes neurais. Como os sistemas de OCR são sistemas que precisam processar muitas informações, é interessante que seja buscada uma solução que permita a obtenção de um desempenho satisfatório (sistema eficiente).

Além destes motivos, a principal característica inerente às redes neurais que as tornam especiais é o fato destas poderem sofrer uma adaptação. A adaptação e o aprendizado são fortes razões para que se implemente um sistema de OCR através da utilização de redes neurais. Como já foi discutido no capítulo 3, esta capacidade de adaptação/aprendizado é extremamente desejável em sistemas OCR, permitindo a criação de sistemas mais flexíveis.

Por último, as redes neurais são uma tentativa de recriar funcionalmente o comportamento do cérebro humano. Se o objetivo final de um sistema OCR é o de reconhecer caracteres e textos, de uma forma a imitar o ser humano em sua tarefa de leitura, nada mais óbvio que as redes neurais sejam utilizadas como uma forma de recriar estes mecanismos que permitem ao homem realizar este tipo de tarefas.

Devido a estes fatores e, inserido no contexto que foi apresentado nos capítulos 2, 3 e 4, deste trabalho, resultou o desenvolvimento de um sistema de OCR, o sistema N^2OCR, que se utiliza de um simulador de uma rede neural para executar a tarefa de reconhecimento. O capítulo 5 descreverá a proposta do sistema N^2OCR, a implementação de um protótipo deste sistema, uma análise dos modelos de rede neural passíveis de utilização e o modelo que foi adotado.