



Universidade Federal Fluminense
Instituto de Computação
Pós-Graduação em Ciência da Computação

Redes Neurais Artificiais: Rede Hopfield e Redes Estocásticas

Juliana Mendes N. Silva

Estudo Orientado
Lúcia Drummond
Roseli Suzi Wedemann

Niterói, RJ
Julho de 2003

Sumário

1. Introdução	3
2. Redes Neurais Artificiais.....	4
2.1 Introdução	4
2.2 Histórico	5
2.3 O modelo neural biológico.....	7
2.4 O cérebro – Processamento Paralelo	8
2.5 O Modelo de McCulloch e Pitts	9
2.6 Arquiteturas das RNAs	10
2.7 Aprendizado da rede	13
2.7.1 Aprendizagem por correção de erros	14
2.7.2 Aprendizado Hebbiano	14
2.7.3 Aprendizado de Boltzmann.....	15
2.7.4 Aprendizado competitivo	15
2.8 Aplicações	16
2.8.1 Aplicações Práticas	16
3. Rede Hopfield.....	18
3.1 O uso da Rede Hopfield como uma memória associativa.....	18
3.2 O modelo de Hopfield.....	19
3.3 Armazenando padrões na rede	20
3.4 Capacidade de Armazenamento.....	22
3.5 A energia funcional	23
3.6 Estados espúrios	25
4. Algoritmos de Simulação e Implementação.....	27
4.1 Algoritmos de simulação para Redes Hopfield	27
4.2 Implementação.....	29
5. Redes Estocásticas	33
5.1 Mecanismos estatísticos de Sistemas magnéticos	33
5.2 A temperatura.....	34
5.3 A Rede Hopfield como uma rede estocástica	36
6. Conclusão.....	38
7. Bibliografia	39
8. Apêndice	40

1. Introdução

O cérebro humano é um sistema cognitivo composto de neurônios interconectados por uma rede de sinapses e que cooperam entre si para processar informações de maneira distribuída. As informações são armazenadas no cérebro pela forma com que os neurônios estão interligados, ou seja, pela topologia de interconexão sináptica. Existe na literatura um modelo computacional esquemático implementado por um algoritmo sequencial de redes neuronais artificiais que simula o cérebro em alguns conceitos associados à patologia mental relacionada ao funcionamento da memória, conhecida como neurose [1]. O modelo de simulação consiste de uma rede neuronal artificial baseada na Máquina de Boltzmann, composta por N nós (neurônios) totalmente conectados, ou seja, existe pelo menos uma ligação sináptica associada a cada par de nó.

Contudo, busca-se uma melhor aproximação dos sistemas neuronais biológicos, onde o número de neurônios é grande e estes se conectam de maneira hierarquizada, podendo ser uma rede não totalmente conectada.

Esta monografia tem como principal objetivo apresentar alguns estudos preliminares que são base para o desenvolvimento de uma versão distribuída do algoritmo de simulação da rede neuronal sendo esta mais próxima do sistemas neuronais biológicos.

O estudo baseou-se em redes neurais artificiais com ênfase no modelo Hopfield e nas principais propriedades das redes estocástica que auxiliam no estudo e na implementação da Máquina de Boltzmann.

Esta monografia está dividida da seguinte maneira. No segundo capítulo é apresentada a técnica das redes neurais, bem como seu histórico, características, modelo computacional e como a rede é capaz de aprender. No terceiro capítulo são discutidas as características do modelo proposto por Hopfield, onde a rede neural é vista como uma rede endereçável pelo conteúdo. O quarto capítulo descreve os algoritmos de simulação sequencial e distribuído e implementação sequencial da Rede Hopfield. O quinto capítulo apresenta uma rede neural como uma rede estocástica, destacando suas características e semelhança com o Modelo Ising. Por fim, no último capítulo é feita uma conclusão do estudo. O apêndice contém o código fonte a alguns dos resultados obtidos.

2. Redes Neurais Artificiais

Numerosos avanços têm ocorrido no desenvolvimento de sistemas inteligentes, alguns consistem em tentar simular as redes neurais biológicas como é o caso das redes neurais artificiais (RNA's). Este tipo de rede está sendo projetada por muitos pesquisadores para resolver uma variedade de problemas em padrões de reconhecimento, predição, otimização, memória associativa e controle.

2.1 Introdução

Qualquer pessoa pode perceber que o cérebro humano é superior a um computador digital. Um bom exemplo no processamento de informação visual é que uma criança consegue reconhecer objetos e faces muito melhor e mais rápido do que um sistema de Inteligência Artificial rodando em um supercomputador.

O cérebro possui muitas características que são desejadas em sistemas artificiais, o que justifica o estudo de computação neural, apesar de que uma grande RNA pode ter centenas ou milhares de unidades de processamento, já o cérebro dos mamíferos pode ter bilhões de neurônios.

RNAs são técnicas computacionais que apresentam um modelo matemático inspirado na estrutura neural de organismos inteligentes. Estas redes adquirem conhecimento através da experiência e podem ser utilizadas na resolução de uma grande classe de problemas encontrados em várias áreas aplicadas: classificação, identificação, diagnóstico, análise de sinais e de imagens, otimização e controle.

De forma análoga ao cérebro, uma RNA é composta por várias unidades de processamento, correspondentes aos neurônios. Estas unidades são interconectadas através de pesos, que são valores numéricos, representando as sinapses. Elas são responsáveis por determinar uma saída que servirá de entrada para outra unidade.

O que difere este modelo de outros é que a informação está armazenada na forma que os neurônios estão conectados e não em endereços de memória. Portanto, podemos destacar que a definição do estado e a forma de atualização dos neurônios, e a topologia de uma rede caracteriza a diversidade de modelos existentes.

As redes neurais têm-se mostrado de fácil implementação, robusta no tratamento de dados com ruído e eficientes especialmente nos problemas em que não se tem uma formulação analítica ou um conhecimento explícito acessível, ou se os dados "estão contaminados" por ruído ou o próprio problema modifica-se com o tempo. O emprego de uma rede neural depende da habilidade em adaptá-la ao problema considerado, por meio de mudanças nos pesos sinápticos (aprendizagem), de modo a aumentar sua eficiência.

O estudo das redes neurais é um dos ramos da Inteligência Artificial que mais se desenvolve, atraindo pessoas de diversas áreas do conhecimento.

2.2 Histórico

A história destas idéias originou-se na psicologia com Aristóteles e a base computacional ou modelo neural no artigo de McCulloch e Pitts [1943].

McCulloch e Pitts descreveram um modelo simplificado, porém poderoso, do neurônio real, que é utilizado na maioria dos modelos de RNAs atualmente. O modelo, chamado MCP, é baseado na neurofisiologia de um neurônio e desempenha uma função de soma e threshold, onde o peso das conexões entre os nós corresponde as sinápses de um neurônio real.

Em 1949 surgiu a primeira teoria neurofisiológica para modificação de sinápses em neurônios reais (determinação do processo de aprendizagem do cérebro) por Donald Hebb. De acordo com sua teoria, "Se um neurônio A é repetidamente estimulado por um neurônio B , ao mesmo tempo em que ele está ativo, A ficará mais sensível aos estímulos de B , e a conexão sináptica de B para A será mais eficiente. Deste modo, B achará mais fácil estimular A para produzir uma saída". Esse procedimento de ajuste de pesos, freqüentemente chamado de regra de Hebb, tem sido adotado de formas diferentes em algoritmos de aprendizado para modelos de RNAs baseados no modelo MCP.

Na década de 60 foi focado, por muitos pesquisadores, o problema de como encontrar os pesos apropriados para particulares tarefas computacionais. Frank Rosenblatt (1962) criou o Modelo Neural Perceptron de uma única camada e conseguiu provar a convergência de um algoritmo de aprendizagem. Seu modelo de rede, baseado no neurônio MCP, era capaz de classificar os padrões aprendidos do ambiente através do ajuste gradual dos pesos entre os neurônios.

Marvin Minsky e Seymour Papert [1969] publicavam um livro onde demonstravam as limitações do Perceptron de uma única camada, contrariando o conceito defendido por Rosenblatt. Seu modelo era incapaz de resolver problemas não linearmente separáveis, como o "Ou Exclusivo".

Rosenblatt estudou estruturas com mais camadas temendo que uma única camada limitava o Perceptrons. Mas, de qualquer forma, não existiam algoritmos de aprendizagem capazes de determinar os pesos sinápticos para implementar alguns cálculos. Com isto a comunidade de ciência da computação deixou o paradigma de redes neurais por mais de 20 anos.

O surgimento de algoritmos de aprendizado mais eficientes para redes multiníveis, como a descrição do modelo de Hopfield (Hopfield, 1982; 1984), das Máquinas de Boltzmann (Ackley, Hinton e Sejnowsky, 1985) e do algoritmo Backpropagation (Rumelhart, Hinton e Williams, 1986a, 1986b), contrariaram um pouco as idéias de Minsky e Papert sobre Perceptrons e provocaram novamente um forte interesse pela área com publicações de outros modelos importantes de Redes Neurais (Willshaw, Buneman e Longuet-Higgins, 1969; Anderson 1972; Kohonen, 1982; 1972; Fukushima 1975; Carpenter e Grossberg 1987a; 1987b; Kanerva, 1984).

Destes novos algoritmos podemos destacar os seguintes. Caianiello [1961] acreditava existir uma memória no sistema e construiu uma teoria estatística, utilizando a idéia de mecanismos estatísticos, incorporando a idéia de aprendizado do cérebro proposta por Hebb. Hopfield [1982] também utilizou a mesma idéia introduzindo uma energia funcional, enfatizando a noção de memória como atratores dinamicamente estáveis. Em [1983,1986] Hilton e Sejnowski construíram formulações usando unidades estocásticas, mas cometeram um engano na probabilidade análoga à temperatura em mecanismos estatísticos. Amit et al [1985,1989], usaram métodos desenvolvidos na teoria randômica de sistemas magnéticos chamados spin glasses [2].

2.3 O modelo neural biológico

O sistema nervoso é constituído basicamente por um conjunto extremamente complexo de células nervosas (neurônios) que se interconectam de forma específica e precisa, formando os chamados circuitos neurais.

O neurônio é o principal componente do sistema nervoso, estima-se que no cérebro humano exista aproximadamente 15 bilhões destas células, que são responsáveis por todas as funções do sistema.

O neurônio [fig 2.1] destaca-se das outras células por ser capaz de processar informações. Sua estrutura morfológica consiste do corpo do neurônio (soma) constituído de núcleo, que dá suporte metabólico à toda célula, o axônio (fibra nervosa) prolongamento único e grande que aparece no soma e é responsável pela condução do impulso nervoso para o próximo neurônio e os dendritos que são prolongamentos menores em forma de ramificações que emergem do soma e do final do axônio, sendo, na maioria das vezes, responsáveis pela comunicação entre os neurônios [6].

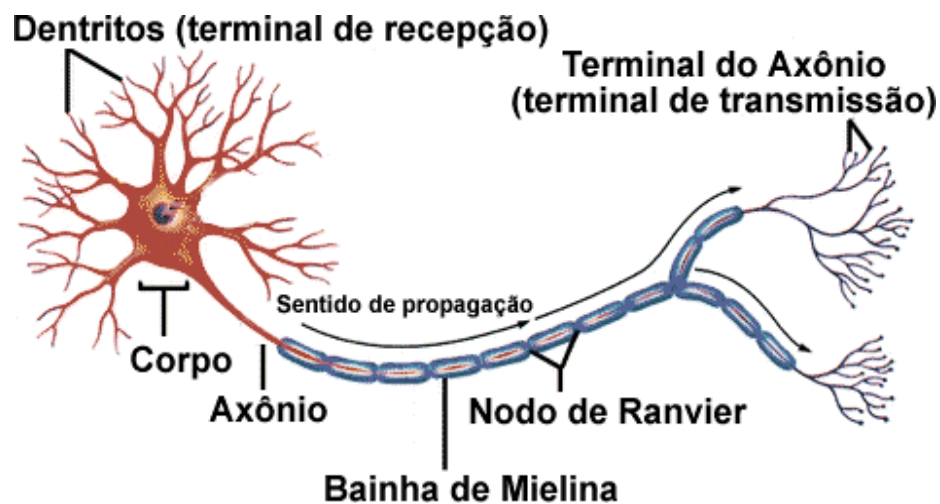


Figura 2.1 - A célula nervosa (neurônio)

Ao receber um estímulo, a membrana do neurônio se despolariza, acontecendo a inversão das cargas elétricas: externamente, ficam as cargas negativas, e internamente, as positivas. Uma vez que um ponto em qualquer lugar do nervo se torne despolarizado, um

impulso nervoso propaga-se a partir dele em cada direção, e cada impulso é assim mantido, propagando-se até atingir as extremidades da fibra.

O percurso do impulso nervoso no neurônio é sempre no sentido dendrito → corpo celular → axônio [fig 2.1]. A região de passagem do impulso nervoso de um neurônio para a célula adjacente chama-se sinapse.

O comportamento de uma sinapse química pode ser resumido da seguinte maneira: quando um potencial de ação chega a um terminal pré-sináptico, a despolarização da membrana pode fazer com que um pequeno número de vesículas sinápticas secrete, na fenda sináptica, um composto químico denominado neurotransmissor, que por sua vez, irá atuar nas proteínas receptoras do neurônio seguinte produzindo alterações na permeabilidade da membrana neural pós-sináptica, podendo excitá-lo, inibi-lo ou modificar de alguma forma sua intensidade [6].

A sinapse pode ser ajustada através das atividades que ela participa, assim pode-se dizer que uma rede neural pode “aprender”.

Os neurônios se comunicam através de uma sequência curta de pulsos e a mensagem é modulada na frequência destes pulsos. A frequência é um milhão de vezes mais lenta do que a maior velocidade de chaveamento de um circuito elétrico. Todavia, tarefas complexas são realizadas em poucos milissegundos devido a forma paralela do cérebro [3].

2.4 O cérebro – Processamento Paralelo

Um córtex cerebral contém 10^{11} neurônios e no total 10^{14} a 10^{15} interconexões complexas, ou seja, a grosso modo o cérebro é como um sistema paralelo de 10^{11} processadores. Cada processador tem um programa muito simples: ele computa a soma de pesos dado de entrada por um outro processador e tem como saída um único número. Esta saída é enviada a um outro processador que faz o mesmo tipo de cálculo. O peso sináptico e a função são diferentes para cada neurônio o que pode ser entendido como um armazenamento local de dados pelo processador.

A alta conectividade da rede não permite que a soma de poucos termos errados cause alguma consequência no resultado, tornando o sistema totalmente robusto.

Fazendo uma breve comparação entre o processamento do cérebro e uma situação convencional da computação, cada processador executa programas complicados e a sequência computacional pode ser arruinada por um único bit errado [2].

“...Inúmeras pesquisas comprovam que uma vida intelectual ativa é fundamental para manter a cabeça funcionando. Explica-se: o cérebro humano possui 100 bilhões de neurônios, ou células nervosas, que regem o movimentado tráfego elétrico e químico do corpo humano. Toda vez que algo chama a atenção - uma visão, um som, uma idéia - um grupo de neurônios é ativado e deflagra milhares de impulsos elétricos nas brechas existentes entre eles, as chamadas sinapses, transformando essa elaborada rede em uma tempestade de atividades sinalizadoras. A memória humana trabalha com esse gigantesco número de informações distribuídas por todo o cérebro e está em contínua mudança...” [7]

2.5 O Modelo de McCulloch e Pitts

McCulloch e Walter Pitts desenvolveram um modelo que propõe elementos computacionais, abstraídos das propriedades fisiológicas dos neurônios biológicos e de suas conexões.

O neurônio de McCulloch-Pitts, MPC [fig 2.2], é um dispositivo binário onde suas várias entradas possuem um estado e pesos nas conexões (pesos sinápticos que podem ser excitatórios ou inibitórios) e sua saída representa um dos dois estados possíveis (1 - pulso ou 0 – não pulso).

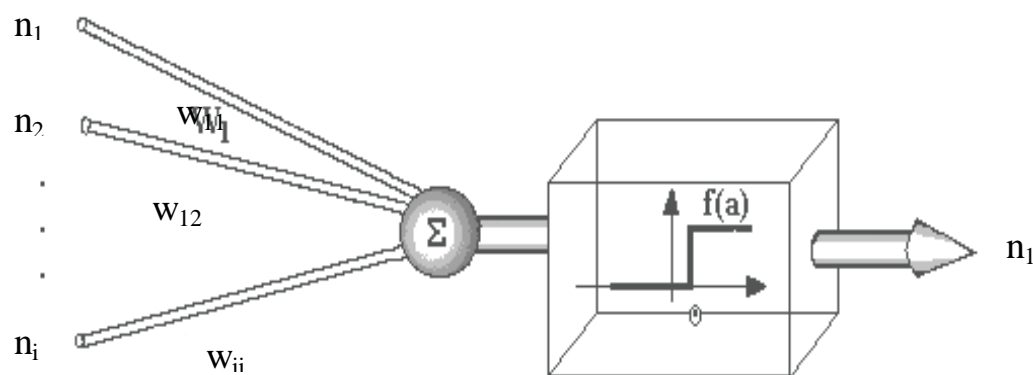


Figura 2.2 – Neurônio de McCulloch e Pitts

Para determinar o estado de um neurônio faz-se a integração sináptica, ou seja, calcula-se a soma ponderada do estado das entradas com seus respectivos pesos sinápticos, positivos nos casos excitatórios e negativos nos casos inibitórios, menos o potencial limite

do neurônio. Se a soma for maior ou igual a um certo limiar (θ) inicialmente fixado, o neurônio torna-se ativo e a saída é um pulso, se não permanece inativo.

Esta soma pode ser vista na equação 2.1, onde n_i é o estado do neurônio i , n_j ($j = 1, 2, \dots, n$) é o estado dos vizinhos, w_{ij} , corresponde ao peso sináptico que conecta i em j e μ_i é o valor do limite potencial do neurônio i que será assumido zero em toda monografia.

$$n_i = \theta \left(\sum_{j=1}^n w_{ij} n_j - \mu_i \right)$$

Equação 2.1 – Função de ativação

Diante do comportamento desta equação, o discriminador linear pode ser utilizado como classificador de padrões, ou seja, através de uma rede formada por simples neurônios, é possível modelar algumas funções ou fenômenos que podem ser representados por funções lógicas capazes de classificar padrões.

McCulloch e Pitts provaram que em princípio, uma escolha adequada dos pesos sinápticos permite que um funcionamento síncrono da rede realize computações distribuídas.

Apesar do MPC ser muito simples quando comparado com um neurônio real, a analogia com um neurônio biológico é clara: cabos e interconexões modelam os dendritos, os pesos das conexões são as sinapses e o potencial limite aproxima a atividade no corpo celular.

2.6 Arquiteturas das RNAs

A forma de organização dos neurônios em uma rede neural (topologia) está intrinsecamente ligada ao problema no qual se deseja solucionar e é fator importante na definição do algoritmo de aprendizagem a ser utilizado.

RNAs podem ser vistas como grafos direcionados onde os neurônios são os nós e as arestas a conexão entre os neurônios. A cada aresta está associado um peso que é correspondente ao peso sináptico.

A arquitetura das redes neurais pode ser dividida em dois grupos:

a. *Redes feed-forward*, onde o grafo não possui laços. Dentro desta arquitetura podemos destacar:

a.1. *Monocamada com alimentação unidirecional* [fig 2.3]: nesse caso existe somente uma camada de neurônios ligados diretamente aos nós de entrada e eles mesmos já fornecem os dados de saída.

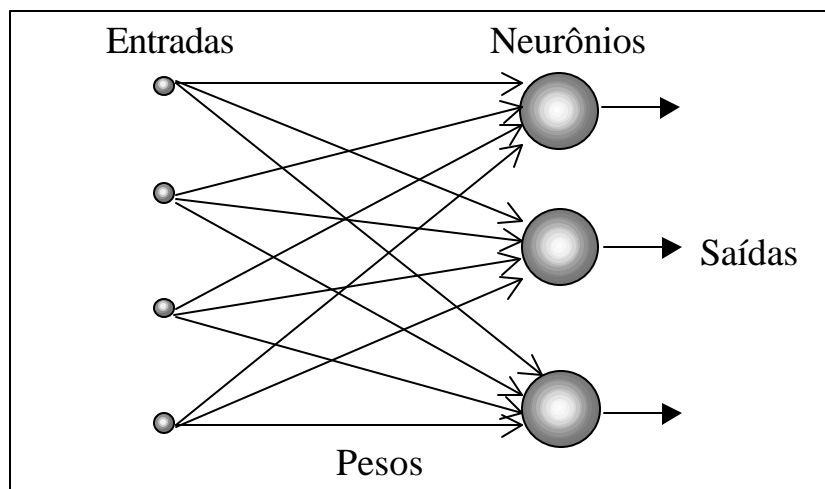


Figura 2.3 - Monocamada com alimentação unidirecional

a.1. *Multicamada com alimentação unidirecional* [fig 2.4]: esse tipo possui uma ou mais camadas de neurônios escondidos. Sua função é processar os sinais de entrada antes de enviá-los aos neurônios de saída. Apesar da maior complexidade, essa arquitetura possibilita uma melhor qualidade de treinamento, pois há maior interação entre os neurônios. Quando todos os neurônios estão conectados a todos os outros neurônios essa rede é dita completamente conectada do contrário é chamada parcialmente conectada.

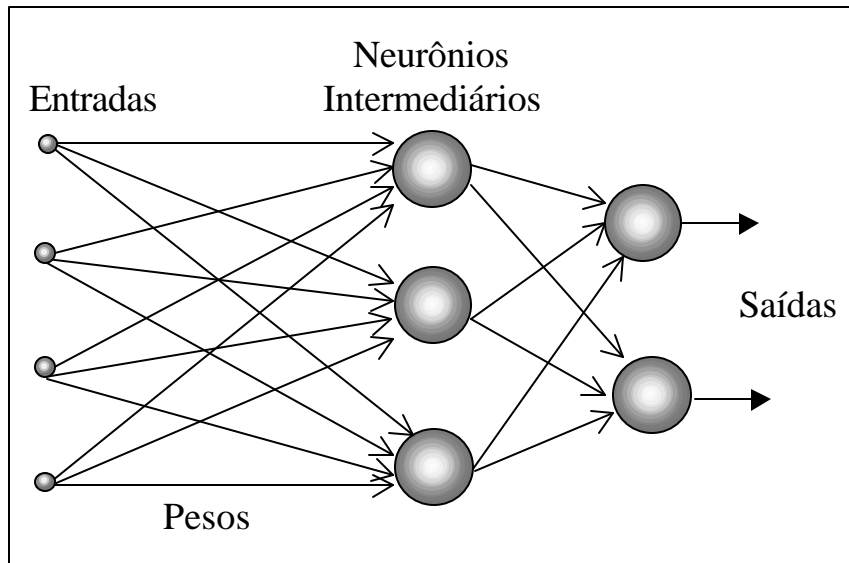


Figura 2.4 - Multicamada com alimentação unidirecional

b. *Redes Recorrentes* [fig 2.5] são arquiteturas neurais que possuem laços de realimentação. Desta forma, produzem um único conjunto de valores de saída. Pode ser monocamada ou multicamada.

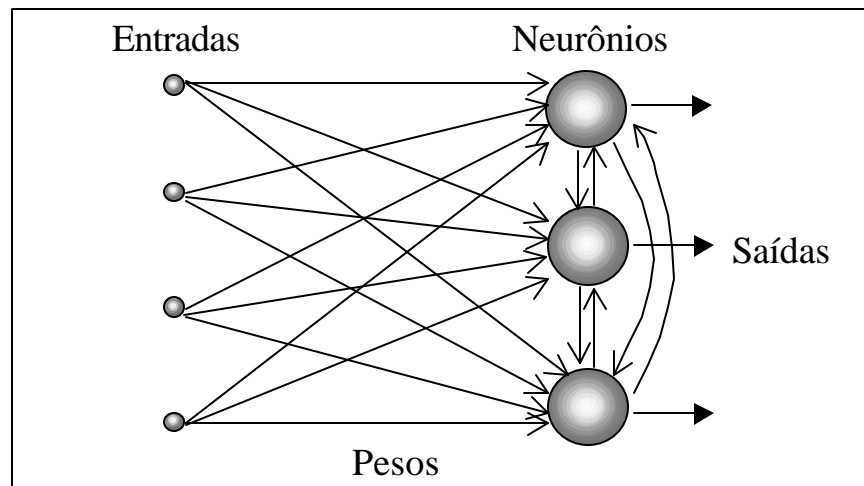


Figura 2.5 - Redes Recorrentes

Conectividades diferentes levam a diferentes comportamentos da rede. As redes *feed-forward* são estáticas, pois produzem um único conjunto de valores de saída. Pode-se dizer que não possui memória já que sua resposta a uma entrada não depende do estado

prévio da rede. Redes recorrente, por outro lado, são sistemas dinâmicos. Quando a entrada é apresentada à rede o neurônio de saída é computado.

A definição da arquitetura utilizada depende das características do sistema a ser desenvolvido [8].

2.7 Aprendizado da rede

Redes neurais “aprendem” por experiência, da mesma forma que uma criança aprende a andar, falar, associar objetos e nomes, através de exemplos ou tentativa e erro. Assim, após a escolha de uma representação para os dados do problema, deve-se montar um conjunto de treinamento para que, através dele, a rede aprenda. Este conjunto é gerado a partir de dados históricos, ou seja, a partir de experiências e fatos ocorridos no passado.

O aprendizado, na maioria das vezes, constitui no ajuste do conjunto de pesos de modo que a rede consiga executar uma tarefa específica. Para se construir um processo de aprendizado é necessário conhecer quais informações estarão disponíveis à rede e como os pesos serão atualizados. Existem três paradigmas de aprendizado:

a. *Aprendizado supervisionado*: é aquele que utiliza um conjunto de pares (entrada - saída), em que para cada padrão de entrada é especificado um padrão de saída desejado (resposta desejada). O aprendizado ocorre no momento em que a saída gerada pela rede, a partir dos cálculos efetuados com o padrão de entrada e os pesos correntes. Se a saída for diferente da saída desejada, o algoritmo de treinamento, segundo alguns critérios, ajusta os pesos da rede de forma a reduzir o erro. Essa dinâmica é repetida inúmeras vezes para todo conjunto de dados (entradas e saídas) até que a taxa atinja uma faixa considerada satisfatória.

b. *Aprendizado não-supervisionado*: este tipo de aprendizado também é conhecido como aprendizado auto-supervisionado, e classifica os padrões similares sem utilizar conjunto de pares, isto é, no treinamento da rede são usados apenas valores de entrada. A rede trabalha essas entradas e se organiza de modo a classificá-las mediante algum critério de semelhança. Esse tipo de rede utiliza os neurônios como classificadores, e os dados de entrada como elementos de classificação.

c. *Aprendizado híbrido*: combina as duas técnicas apresentadas. Parte dos pesos são determinados pelo aprendizado supervisionado e a outra parte através do aprendizado não-supervisionado.

Entre as questões práticas e fundamentais a respeito da aprendizagem pode-se destacar, a capacidade de armazenamento da rede e limite de decisões que a rede deve tomar, a complexidade dos exemplos, onde são determinados o número de padrões de treinamento necessários para treinar a rede e qual será o tempo computacional gasto para que a rede “aprenda”.

Existem quatro tipos básicos de regra de aprendizado.

2.7.1 Aprendizagem por correção de erros

Neste modelo a correção dos pesos sinápticos é efetuada através da comparação da saída desejada com a saída atual, ambas medidas no mesmo instante de tempo. O erro é então definido com sendo a diferença do valor da saída desejado e a saída atual.

A fim de se estimar o erro da rede define-se uma função custo baseada na função erro. Esta função indica o erro instantâneo da rede e procura-se em adaptar os pesos sinápticos a fim de minimizar essa função.

É importante notar que a rede só aprende quando ocorre um erro. Um valor pequeno proporcionará uma aprendizagem lenta e pode acontecer da rede demorar a convergir para a solução proposta. Por outro lado, uma taxa alta pode comprometer a estabilidade do sistema, pois a rede pode divergir da solução.

2.7.2 Aprendizado Hebbiana

Este processo é baseado na teoria de Hebb [seção 2.2]. Hebb observou uma característica dos neurônios que consiste no reforço ou enfraquecimento do sinal sináptico de acordo com certos pré-requisitos. A aplicação desta regra às redes neurais pode ser escrita da seguinte forma. Se dois neurônios, um em cada lado de uma sinapse são ativados simultaneamente, então a força daquela sinapse é seletivamente aumentada. Todavia, se dois neurônios, um em cada lado de uma sinapse, são ativados assincronamente então a força daquela sinapse é seletivamente enfraquecida ou eliminada.

Uma importante característica desta regra é que o aprendizado é feito localmente, ou seja, a mudança dos pesos sinápticos depende somente da atividade de dois neurônios conectados.

O modelo Hebbiano possui uma base neurológica importante: há fortes indícios de que as atividades cerebrais das células do hipocampo responsáveis pela aprendizagem e memória funcionam de forma semelhante [2].

2.7.3 Aprendizado de Boltzmann

Esse método de aprendizagem tem suas origens na teoria da computação além de contar com algumas considerações da teoria da termodinâmica.

Chamada de Máquina de Boltzmann, a rede neural é formada por uma rede recorrente simétrica operando em somente dois estados, -1 e 1 , e é caracterizada por uma função de energia cujo valor é determinado pelos estados dos neurônios. A atualização dos estados dos neurônios é feita de forma aleatória e acontece até que um mínimo da função de energia seja alcançado. Cada neurônio é, na verdade, uma unidade estocástica que gera uma saída de acordo com a distribuição de Boltzmann dos mecanismos estatísticos.

Existem dois subconjuntos de neurônios nesta estrutura, um subconjunto formado por neurônios visíveis e outro formado por neurônios escondidos, sendo que este último não possui contato com o exterior.

2.7.4 Aprendizado competitivo

No aprendizado competitivo os neurônios são projetados de tal forma que somente um deles está ativo após a computação de determinada tarefa. Cada neurônio possui um mecanismo de neutralização dos demais, logo, se um neurônio for ativado, ele é o único a fornecer uma saída do sistema, pois ele anula os demais.

Existem os seguintes componentes em uma rede neural competitiva:

- a. Um conjunto de neurônios está interligado a diversas sinapses aleatoriamente distribuídas e com pesos diferentes de forma que cada um deles possui uma resposta única a um determinado conjunto de sinais de entrada;
- b. Um limite de força a cada neurônio;

c. Um mecanismo que permite os neurônios competirem pelo direito de responder a um determinado conjunto de entradas.

Esta estrutura possui a característica de que cada neurônio se torna especialista em reconhecer determinados padrões de entrada.

2.8 Aplicações

Através da computação paralela a rede neural atua em todos os dados simultaneamente. Isto lhe dá a capacidade de realizar algumas tarefas desejadas por muitos sistemas.

a. *Memória Associativa Distribuída*: a memória de uma rede neural é distribuída e ao mesmo tempo associativa. Diante de um padrão de entrada, a rede é capaz de recuperar um padrão armazenado em sua memória, mais próximo daquele que lhe foi apresentada.

b. *Reconhecimento de padrão*: habilidade de verificar simultaneamente uma grande quantidade de informação e gerar uma saída categórica ou generalizada.

2.8.1 Aplicações Práticas

a. *Redes Neurais Lineares*

a.1. *Telecomunicações*: modems de alta velocidade de transmissão utilizam equalizadores de linha adaptativos e canceladores de eco adaptativos. Cada um desses sistemas adaptativos utiliza uma rede neural.

a.1. *Controlador de acelerador de feixe de partículas*: o Centro de Aceleração de Partículas Lineares de Stanford utiliza técnicas adaptativas para cancelar perturbações que diminuam a precisão de feixes de pósitrons e elétrons durante a colisão.

b. *Redes neurais não-lineares*

b.1. *Reconhecimento de caracteres*: existem diversos dispositivos comerciais disponíveis. Um deles reconhece caracteres da língua japonesa, possui 10 milhões de sinapses e seu desempenho é superior a dos sistemas convencionais.

b.1. *Controle de qualidade em fábricas*: existem diversas redes neurais sendo utilizadas no controle de qualidade na indústria em geral. São aplicações sobre detecção de nível de contaminação, classificação de erros em alto-falantes e até avaliação do grau de pureza do suco de laranja.

b.1. *Controle no processo de fabricação de semicondutores*: a utilização de uma rede neural para determinar o momento de corte de dopagem e correção de erros de densidade promoveu uma melhora por um fator acima de dois outros métodos convencionais.

3. Rede Hopfield

Hopfield usou uma função de energia como ferramenta para desenhar redes recorrentes [seção 2.6] e entender seu comportamento dinâmico. Assim, popularizou o uso desta arquitetura como uma memória associativa e para resolver problemas de otimização.

A Rede Hopfield emprega um princípio chamado de armazenamento de informação como atratores dinamicamente estáveis. A recuperação de informação acontece via um processo dinâmico de atualização dos estados dos neurônios, sendo que, o neurônio a ser atualizado, é escolhido randomicamente.

Dois modelos de redes foram apresentados por Hopfield: o analógico e o binário. Nesta monografia será apresentado o modelo binário.

3.1 O uso da Rede Hopfield como uma memória associativa

A memória humana funciona como uma memória endereçada pelo conteúdo, por exemplo, muitas vezes uma pessoa pode lembrar de fatos inteiros tendo inicialmente uma pequena lembrança do acontecimento. A Rede Hopfield funciona exatamente como uma memória endereçada pelo conteúdo também chamada de memória associativa.

Na área computacional uma memória associativa ou endereçada pelo conteúdo pode ser entendida como um dispositivo que permite armazenar um conjunto p de padrões \hat{x}_i^1 , (com $i = 1, 2, 3, \dots, p$ e $i = 1, 2, 3, \dots, N$, onde N é a quantidade de neurônios da rede) de tal modo que quando apresentado um novo padrão x_i , a rede responde produzindo o padrão armazenado que possui a menor distância (ou seja, o mais parecido) de x_i .

A menor distância entre dois padrões é medida pela distância de *Hamming* [eq 3.1], que é o número de bits que um padrão difere do outro.

$$\sum_i [\hat{x}_i^1 (1 - x_i) + (1 - \hat{x}_i^1) x_i]$$

Equação 3.1 – Distância de Hamming

Como será mostrado nas próximas seções, a evolução da rede (atualização dos neurônios) leva à estados estáveis, então, cada padrão \hat{t}_i^j pode ser associado a um destes estados. Quando a rede for inicializada α , sendo este próximo de \hat{t}_i^j , a rede evoluirá para este estado estável e o padrão poderá ser recuperado. Por isto, a rede corrige erros [seção 2.7.1] no sentido que anula informações inconsistentes apresentadas a ela.

3.2 O modelo de Hopfield

O estado de um neurônio i , denotado, a partir de agora, por S_i , pode assumir o valor 1 (quando estão ativos) ou -1 (quando não estão ativos). Um padrão é, na verdade, um vetor de estados, onde o tamanho do vetor é igual ao número de neurônios da rede. Por exemplo, um vetor $\hat{t}^j = [1, -1, 1]$ consiste de uma rede com três neurônios, onde o estado do primeiro neurônio é igual a 1, o segundo igual a -1 e o terceiro igual a 1.

A dinâmica da rede (ou seja, atualização dos neurônios) é dada pela seguinte função:

$$S_i := \text{sgn}(\sum_j w_{ij} S_j)$$

Equação 3.2 – Atualização do neurônio i

Onde a função $\text{sgn}(x)$ é uma função *signum* definida por:

$$\text{sgn}(x) = \begin{cases} 1 & \text{se } x \geq 0; \\ -1 & \text{se } x < 0; \end{cases}$$

Equação 3.3 – Função $\text{sgn}(x)$

Existem dois modos de atualização dos neurônios especificados como síncrona e assíncrona. A primeira é apropriada para simulação e existe um relógio global além de ser sensível a erros de tempo. Já a segunda pode ser feita escolhendo-se randomicamente um neurônio i a ser atualizado ou o próprio neurônio escolhe se atualizar segundo a regra em uma probabilidade por unidade de tempo.

Os efeitos da maneira de como são realizadas as atualizações serão discutidas no próximo capítulo.

3.3 Armazenando padrões na rede

Para que um padrão \hat{i}_i seja armazenado, é necessário que ele seja estável. Um padrão \hat{i}_i é estável quando

$$\hat{i}_i = \text{sgn}(\sum_j w_{ij} \hat{i}_j)$$

Equação 3.4 – Estabilidade

A condição de estabilidade é verdadeira porque a regra 3.2 não produz mudanças nos estados dos neurônios, portanto o estado \hat{i}_i é estável. O valor da sinapse w_{ij} entre i e j é igual a

$$w_{ij} = 1/N \hat{i}_i \hat{i}_j$$

Equação 3.5 – Peso sináptico

A constante de proporcionalidade $1/N$ é utilizada para simplificar a descrição matemática de uma recuperação de informação.

Quando um padrão de entrada é apresentado a rede, mesmo que com ruído (alguns bits diferentes do padrão armazenado), o padrão armazenado que for mais próximo do padrão de entrada poderá ser recuperado. Este fato pode ser comprovado substituindo a equação 3.5 na equação 3.4., onde h_i é o estado estável do neurônio i . Pelo desenvolvimento da equação 3.6 tem-se que se S_j (estado do neurônio j) é igual \hat{i}_j (padrão \hat{i}_j armazenado), $\hat{i}_j * S_j$ é positivo. Se S_j é diferente de \hat{i}_j , o valor de $\hat{i}_j * S_j$ é negativo. Por fim, pela somatória, se a maioria de bits são iguais a equação se reduz para $h_i = \text{sgn}(\hat{i}_i)$, ou seja, o estado estável do neurônio é igual ao estado estável do padrão armazenado, assim o padrão é então recuperado.

Equação 3.6 – Estado estável de um neurônio i

Logo, uma configuração inicial perto (em distância de Hamming) de \hat{t}_i relaxará para \hat{t}_i . A rede é capaz de corrigir erros como esperado.

Além das afirmações anteriores, pelas definições das equações, pode-se perceber que a saída de um neurônio na rede é *feed-back* para os outros neurônios, não existe auto *feed-back* (ou seja, $w_{ii} = 0$) e que a matriz de pesos sinápticos é simétrica, pois $w_{ij} = w_{ji}$.

Até o momento, foi discutido o armazenamento de um único padrão. Mas o interessante é poder armazenar uma quantidade maior de padrões $\hat{t}_i^{\hat{i}}$, onde $\hat{i} = 1, 2, \dots, p$. Para tanto a equação 3.5 será modificada para que o cálculo dos pesos sinápticos não seja em relação a um único padrão, mas em relação a p padrões.

$$w_{ij} = 1/N \sum_{\hat{i}=1}^p \hat{t}_i^{\hat{i}} \hat{t}_j^{\hat{i}}$$

Equação 3.7 – Peso sináptico em relação a p padrões

Esta regra de cálculo de pesos é chamada regra de Hebb por causa da similaridade com a teoria desenvolvida por Hebb [seção 2.2]. Quando aplicamos o conjunto de padrões a serem armazenados durante a fase de treinamento e ajustamos os pesos sinápticos chegamos na equação 3.7.

Para facilitar a análise de estabilidade, considere padrão estável $\hat{t}_i^{\mathbf{n}}$. Para que a condição de estabilidade seja satisfeita é necessário, como no caso de um único padrão armazenado, que $\text{sgn}(h_i^{\mathbf{n}}) = \hat{t}_i^{\mathbf{n}}$. Para avaliar se esta condição realmente é satisfeita, os pesos sinápticos são substituídos na equação 3.4, como feito na equação 3.6, com a diferença que os termos onde \hat{i} é igual a \mathbf{n} e \hat{i} é diferente de \mathbf{n} são escritos separadamente

$$h_i^{\mathbf{v}} = \text{sgn}(\hat{t}_i^{\mathbf{v}} + 1/N \sum_{\hat{i} \neq \mathbf{v}} \sum_{\hat{j} \neq \mathbf{v}} \hat{t}_i^{\hat{i}} \hat{t}_j^{\hat{i}} \hat{t}_j^{\mathbf{v}})$$

Equação 3.8 – Estabilidade de p padrões

Se o segundo termo da equação, chamado de *Crosstalk*, for igual a zero, pode-se dizer, logicamente, que o estado do neurônio i pertencente ao padrão \mathbf{n} é estável (pois $h_i^{\mathbf{n}} = \hat{i}_i^{\mathbf{n}}$) e o padrão $\hat{i}_i^{\mathbf{n}}$ é então recuperado. Mas isto é ainda verdadeiro se este termo for menor que 1 e isto acontece quando o número de padrões armazenados é bastante pequeno. Então, quando a rede é inicializada com um padrão com poucos bits diferentes de $\hat{i}_i^{\mathbf{n}}$ ela relaxa para $\hat{i}_i^{\mathbf{n}}$. É exatamente por este fato que a rede de Hopfield é chamada memória endereçada pelo conteúdo.

Cada padrão armazenado \hat{i}_i^j está associado a um estado estável (um mínimo local) e é chamado atrator. A rede quando inicializada dentro de um ponto da base de atração é influenciada por um atrator e tende a se estabilizar, dinamicamente, neste atrator.[fig 3.1].

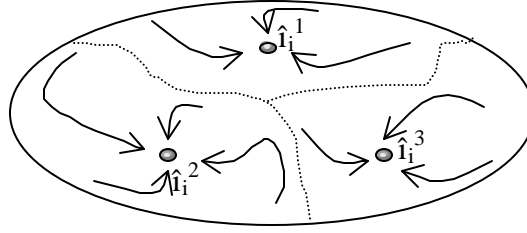


Figura 3.1 – Configuração de um espaço contendo três atratores

3.4 Capacidade de Armazenamento

Na seção anterior, foi discutido que dado um padrão próximo de $\hat{i}_i^{\mathbf{n}}$ a rede estabilizará em $\hat{i}_i^{\mathbf{n}}$ dependendo do número p de padrões. Nesta seção, será apresentado qual é a capacidade da rede na tentativa de se armazenar p padrões.

O valor da capacidade p_{\max} da rede está ligado a aceitabilidade de erros durante a recuperação de informação.

De acordo com [1], o erro, P_{erro} , é proporcional a probabilidade da capacidade ser maior que 1. A medida que tenta-se aumentar o número de padrões armazenados, P_{erro} aumenta, veja tabela 3.1.

P_{erro}	p_{max}/N
0,001	0,105
0,0036	0,138
0,01	0,185
0,05	0,37
0,1	0,61

Tabela 3.1 – Capacidade da rede

A tabela mostra os valores de p/N para vários valores de P_{erro} admitidos. Por exemplo, se o critério suporta que o erro seja, menor ou igual a 0.01 (ou 1%), o número máximo de padrões armazenado, p_{max} , é igual a $0.185N$. Em um caso real, se aceitável é ainda de 1% uma rede de 50 neurônios armazenaria no máximo 9 padrões.

É importante notar que este erro refere-se ao estado estável inicial dos padrões, ou seja, inicialmente, poucos bits estarão errados. Na próxima iteração estes bits podem alterar o valor dos demais bits corretos. E por fim, no pior caso, pode ocorrer um fenômeno avalanche e o padrão poderá não ser recuperado.

Por outro lado se não é aceitável qualquer probabilidade de erro a capacidade da rede é proporcional a $N/\log N$. [1]

3.5 A energia funcional

A energia funcional, H , é uma função de configuração dos estados S_i dos neurônios e é dada por:

$$H = -1/2 \sum_{ij} w_{ij} S_i S_j$$

Equação 3.9 – Energia funcional

com i é diferente de j . Quando i igual a j não produz consequência no valor de H , pois $S_i^2 = 1$ e portanto contribuem apenas com uma constante em H .

A principal propriedade desta energia é que ela sempre decresce (ou permanece constante) à medida que o sistema desenvolve segundo a equação 3.2. Os atratores (ou seja, os padrões armazenados) são “mínimos” desta função. Esta característica da rede é explorada por pesquisadores da área de otimização. A idéia básica é transformar a energia funcional em uma função objetivo.

Para que exista uma energia funcional em um sistema as conexões devem, obrigatoriamente, ser simétricas, apesar de que esta afirmação não é verdadeira nas redes neurais verdadeiras.

É fácil mostrar que a energia da rede decresce sempre que um bit S_i é invertido. Suponha que S_i' é o novo valor de S_i ($S_i' = -S_i$) e o valor da energia funcional de S_i' igual a H' e a de S_i igual a H e fazendo $H' - H$ tem-se:

$$H' - H = -\sum_{j \neq i} w_{ij} S_i' S_j + \sum_{j \neq i} w_{ij} S_i S_j$$

$$H' - H = \sum_{j \neq i} w_{ij} S_i S_j + \sum_{j \neq i} w_{ij} S_i S_j$$

$$H' - H = 2S_i \sum_j w_{ij} S_j - 2w_{ii}$$

Equação 3.10 – Decrescimento da energia.

O somatório do primeiro termo da equação é sempre negativo, uma vez que $S_i' = \text{sgn}(\sum w_{ij} S_j)$ e S_i' é negativo. O segundo termo também será negativo pela equação 3.7, onde, $w_{ii} = p/N$. Este último termo é omitido da equação, pois ele afeta a dinâmica da rede e pode levá-la a estados espúrios que não corresponde a nenhum dos padrões armazenados [seção 3.6].

Vale ressaltar que o comportamento da energia funcional é garantido ocorrer se, e somente se, dois neurônios, que são interconectados por um peso sináptico diferente de zero, têm seus estados atualizados concorrentemente [4].

3.6 Estados espúrios

Além dos estados estáveis armazenados existem alguns outros estados, que também são mínimos, em que a rede se estabiliza. Dentre estes destacam-se: estados reversos, estados misturados e *spin glass*.

Os estados reversos ocorrem por causa da simetria existente entre a dinâmica da rede e a energia funcional. Estes estados $-\hat{i}^i$ são mínimos e são recuperados (a rede se estabiliza nestes estados) quando o número de bits diferentes entre o padrão de entrada e o padrão armazenado é igual a mais da metade dos bits. Veja o exemplo abaixo considerando $\hat{i} = [1, -1, 1, -1, 1]$ o padrão armazenado na memória. As setas indicam em qual padrão a rede estabilizará.

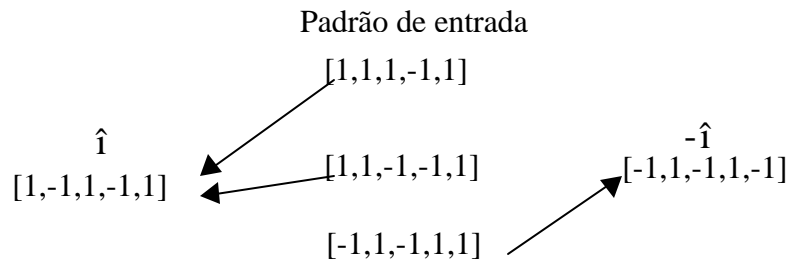


Figura 2.2 – Estados espúrios

Quando é apresentado o padrão $[1, 1, 1, -1, 1]$ a rede estabiliza no padrão recuperado pois a distância de Hamming entre o padrão de entrada e o padrão armazenado é igual a 2. Quando o padrão de entrada possui uma distância de Hamming igual a 4 ($[-1, 1, -1, 1, 1]$) a resposta da rede é um padrão igual ao inverso do padrão armazenado pela rede, ou seja é igual a $-\hat{i}$.

Os estados misturados, \hat{i}_i^{mix} , não são iguais a nenhum padrão armazenado, eles correspondem a somatória linear de um número ímpar de padrões estáveis. \hat{i}_i^{mix} tem o mesmo sinal de \hat{i}_i^{i1} $\frac{3}{4}$ da vezes em média. Portanto a distância de Hamming de \hat{i}_i^{mix} para os padrões pertencentes a somatória (\hat{i}_i^{i1} , \hat{i}_i^{i2} , \hat{i}_i^{i3} , por exemplo) é de $N/4$. A prova da

estabilidade deste estado é feita do mesmo modo que na equação 3.6, substituindo o valor de \hat{t}_i^n por \hat{t}_i^{mix} para se descobrir o valor de h_i .

O sistema não escolhe um número de padrões par, pois na somatória um termo pode anular o outro, assim o estado não será estável.

Por último, para um grande valor de p , existem mínimos locais que não são associados com qualquer número finito dos padrões originais, estes estados são chamados de *spin glass* em analogia aos modelos *spin glass* do mecanismo estatístico.

Uma forma de evitar estes estados é executar a dinâmica da rede até que um padrão válido seja recuperado [seção 4.2]. Outra forma é estabelecer uma distância de *Hamming* mínima segura entre o padrão e os padrões misturados. Se a distância é menor que o mínimo estabelecido os estados dos neurônios não são atualizados.

4. Algoritmos de Simulação e Implementação

Baseado nas características da Rede Hopfield apresentadas no Capítulo 3 serão apresentados neste capítulo dois algoritmos. O primeiro para uma implementação sequencial e o segundo para uma implementação paralela.

4.1 Algoritmos de simulação para Redes Hopfield

De acordo com as características da Rede Hopfield descritas nas seções anteriores, a energia que envolve a rede não pode aumentar depois de atualizações sucessivas dos neurônios. A energia deve decrescer até que uma nova atualização não lhe cause efeito, ou seja, ela permaneça constante.

No algoritmo proposto deve-se considerar a frequência com que os estados dos neurônios são atualizados, ou seja, um neurônio deve ter a chance de atualizar seu estado com uma determinada frequência, e a concorrência desta atualização, sendo que, a concorrência de atualização de dois neurônios requer que o peso sináptico entre eles seja diferente de zero.

A cada iteração todos os neurônios têm a chance de atualizar seus estados (de acordo com a equação 3.4) em uma ordem pré-fixada até que os estados atualizados de todos os neurônios sejam iguais aos seus estados anteriores.

Sendo v_i^0 um vetor que contém o estado inicial dos neurônios (v_i^0 é igual ao padrão de entrada, ou seja o padrão a ser recuperado pela rede), v_i um vetor que contém o estado atual dos neurônios e v_i^- um vetor que contém os estados anteriores do neurônio. No primeiro passo do algoritmo os estados atuais dos neurônios terão valores iguais à v_i^0 , posteriormente cada neurônio terá seu estado atualizado até que tenham seus estados constantes ($v_i = v_i^-$). A figura 4.1 descreve os passos do algoritmo.

```

Algoritmo Seq_Hopfiel
para i de 1 até n
     $v_i = v_i^0$ 
fim_para

repita
    para i = 1 até n
         $\tilde{v}_i = v_i$ 
         $v_i = \text{sgn}(\sum_j w_{ij} v_j)$ 
    fim_para
    até  $v_i = \tilde{v}_i$  para todo  $n_i \in N$ 
fim

```

Figura 4.1 – Algoritmo Seqüencial

Este algoritmo é totalmente determinístico, porém isto talvez não seja de interesse para uma rede que funciona como memória associativa, onde dado um padrão de entrada aproximadamente igual a vários mínimos locais de energia a rede desenvolverá para qualquer um dos mínimos da energia e não aquela mais próxima do padrão de entrada.

Uma forma de evitar que a produza este efeito é cada neurônio ter uma probabilidade pré-definida para atualizar seu estado. Esta probabilidade dever ser a mesma para todos os neurônios, fazendo com que eles sejam atualizados com igual frequência.

Na versão paralela deste algoritmo deve-se considerar a concorrência de atualização dos neurônios.

A convergência da simulação ocorre quando nenhuma atualização futura tem efeito sobre o estado atual do neurônio. O processador p_0 participa como um detector desta convergência.

O algoritmo paralelo possui os procedimentos *Inicialize_i*, *Update_i(MSG_i)*, e *Notify_i* para cada processador p_i . As mensagens trocadas entres os processadores são os estados dos neurônios que são utilizadas na atualização.

Algoritmo Dist_Hopfiel

```

Inicializei
  1.  $v_i = v_i^0$ 
  2. Envie  $v_i$  para  $p_0$ 
  3. Envie  $v_i$  para todos os vizinhos  $p_i$  de  $p_j$ , com arestas saindo
    de  $p_i$  para  $p_j$ 

Updatei (MSGi)
  1. Seja  $v_j \in \text{MSG}_i$  o estado de  $n_j$  para todo  $n_j \in V(n_i)$ 
  2.  $v_i = \text{step}(\sum_j w_{ij} v_j)$ 

Notifyi
  1. Envie  $v_i$  para  $p_0$ 
  2. Envie  $v_i$  para todos  $n_j$  tais que  $n_j \in V(n_i)$ 
fim

```

Figura 4.2 – Algoritmo Dist_Hopfield

4.2 Implementação

O algoritmo sequencial [fig 4.1] foi implementado para melhor entendimento do modelo desenvolvido por Hopfield. Além de aplicar de modo prático a teoria desenvolvida até aqui através dos resultados obtidos pôde-se observar e verificar as características que envolvem o dinamismo da rede.

O primeiro passo da implementação consistiu em armazenar os padrões na rede, ou seja, definir os pesos sinápticos, w_{ij} , segundo a regra de Hebb [eq 3.7]. Os padrões são lidos de um arquivo de entrada passado como parâmetro e os pesos sinápticos só são calculados se o número de padrões a serem armazenados está de acordo com a capacidade da rede como visto na seção 3.4. A porcentagem de erro aceitável foi de 0.36%, portanto a capacidade da rede é proporcional a $0.138N$.

Depois dos padrões armazenados a rede pode ser utilizada para recuperar padrões. Para tanto, foi criada uma função capaz de gerar padrões de entrada aleatórios. Para facilitar os testes, esta função possibilita o usuário escolher se o padrão de entrada deve ser igual a

um dos padrões armazenados (distância de *Hamming* = 0 ou N), ou se ele deve ser gerado totalmente aleatório.

Como na figura 4.1, um laço permite que todos os neurônios pertencentes à rede tenham seus estados atualizados até que, perante uma atualização, nenhum neurônio tenha seu estado alterado. A cada atualização, ou seja, todas as vezes que um neurônio tem seu estado atualizado a energia da rede é calculada pela equação 3.9. A energia alcança seu valor mínimo quando a condição de parada do algoritmo é satisfeita e os estados dos neurônios não sofrem mais alterações.

Os resultados obtidos nos testes mostraram que quando a rede armazena padrões que possuem pequena distância de *Hamming* uns dos outros a rede alcança um mínimo que não corresponde ao mínimo do padrão mais perto em distância de *Hamming* do padrão de entrada. Por exemplo, em uma rede com 30 neurônios foram armazenados os seguintes padrões:

[illegible]

[illegible]

[illegible]

Se apresentado o padrão $\hat{1}$ como entrada a rede recuperará o padrão $\hat{2}$. Isto acontece porque o padrão dado de entrada é muito próximo de todos os padrões armazenados na rede, portanto a rede alcança uma energia mínima porém diferente do mínimo desejado.

Para tentar resolver este problema os neurônios não são mais atualizados sequencialmente, mas como sugerido na seção 4.1, os neurônios têm seus estados atualizados em uma sequência aleatória. Como resultado, o problema descrito acima foi eliminado. Porém a rede muitas vezes estabiliza em estados espúrios. Este fato acontece, na maioria das vezes, quando o padrão de entrada é próximo em distância de Hamming de todos os padrões armazenados. O resultado da execução do algoritmo descrito abaixo exemplifica esta situação.

Padrões armazenados – entrada30.txt

$$\hat{1}^l = \{1-1\ 1-1\ 1-1\ 1-1\ 1-1\ 1-1\ 1-1\ 1-1\ 1-1\ 1-1\ 1-1\ 1-1\}$$

[illegible]

[illegible]

O numero de bits diferentes do padrao de entrada e de $\hat{1}$ e 16

O numero de bits diferentes do padrao de entrada e de \hat{r} e 15

O numero de bits diferentes do padrao de entrada e de \hat{r}^3 e 13

⇒ Estado Espúrio encontrado depois de 500 atualizações

***** Padrao Recuperado *****

Para a entrada: -1-1 1 1 1-1-1 1 1 1 1 1-1-1-1-1 1 1-1-1-1 1-1 1 1 1 1 1-1

O padrao recuperado e:-1 1-1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1-1 1-1 1

A energia da rede e: -22.099918

Como pode-se perceber o padrão que deveria ter sido recuperado é o 3, portanto, o padrão recuperado não corresponde a nenhuma dos padrões armazenados. Ele é um estado espúrio que também faz parte do conjunto de mínimos da função de energia.

Para tentar evitar a recuperação de estados espúrios foi permitido ao algoritmo que executasse um número x de iterações. A cada iteração a saída da rede é dada como entrada para que os estados sejam novamente atualizados tentando-se assim “fugir” destes estados que provocam uma certa instabilidade na rede. Bons resultados foram alcançados. Em muitos dos casos, depois de um determinado número de iterações a rede conseguia alcançar um novo estado estável, retornando um dos padrões armazenados na rede. Por muitas vezes este retorno é exatamente o padrão que possui a menor distância do padrão de entrada.

Uma outra análise pode ser feita em cima dos estados reversos. Estes estados são simétricos aos padrões armazenados, portanto quando a rede é inicializada com mais da metade de bits diferentes de um determinado padrão ela estabiliza em seu estado reverso. Por exemplo, considere os seguintes padrões armazenados:

$$\begin{aligned}\hat{\mathbf{i}}^1 &= \{-1 \ -1 \ -1 \ 1 \ 1 \ 1 \ 1 \ 1 \ -1\} \\ \hat{\mathbf{i}}^2 &= \{-1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ 1 \ 1 \ -1\}\end{aligned}$$

Dado um padrão de entrada igual a $\{1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ -1\}$, a distância de *Hamming* do padrão de entrada para o padrão \hat{t}^1 é maior do que a distância do padrão de entrada do estado reverso do padrão \hat{t}^2 , portanto o padrão recuperado será o $-\hat{t}^2$.

Como o padrão $-\hat{t}^2$ é simétrico ao \hat{t}^2 esta recuperação não é considerada um erro, mas uma particularidade da Rede Hopfield devido a simetria de suas conexões.

5. Redes Estocásticas

Quando uma unidade da rede (um neurônio) é considerada uma unidade estocástica e é adicionado a idéia de temperatura, a Rede Hopfield é análoga a um modelo simples de material magnético em física estatística.

Uma descrição simples do material magnético consiste de um conjunto de *spins*, chamados de magnetos atômicos, distribuídos em uma latência regular. O *spin* pode apontar para várias direções, sendo que, o número de direções possíveis depende do tipo atômico considerado. Em um caso mais simples possui duas direções. Estas direções são representadas pela variável S_i , onde cada estado i pode assumir o valor igual a 1, se o *spin* está apontado para cima e -1 se apontado para baixo.

Neste capítulo, serão discutidas as particularidades deste material magnético chamado de *Modelo Ising* e comparado com o modelo estudado até agora. Novos conceitos serão inseridos e estes servirão de base no entendimento da Máquina de Boltzmann.

5.1 Mecanismos estatísticos de sistemas magnéticos

Em um material magnético um *spin* é influenciado por um campo magnético que consiste de um campo externo, h^{ext} , aplicado pela experiência e um campo interno produzido por um outro *spin*. A influência de um *spin* em um campo interno é proporcional a sua localização em relação a outro *spin*, ou seja, o estado de um *spin* depende do estado de todos os outros vizinhos, como acontece na Rede Hopfield. Todavia, podemos utilizar a equação 3.2 para descrever o estado do *spin*, onde w_{ij} é a influência de um *spin* S_j sobre um *spin* S_i e h_i é o estado (chamado de campo magnético) de S_i sob a influência do campo interno (dos outros *spins*) mais do campo externo.

$$h_i = \sum_j w_{ij} S_j + h^{ext}$$

Equação 5.1 Estado de spin

O campo magnético h_i no *spin* S_i corresponde ao estado estável de S_i , como na Rede Hopfield. A estabilidade de um campo acontece quando os *spins* alinham-se ao campo magnético h_i que age sobre eles, ou seja, $S_i = \text{sgn}(h_i)$. Isto normalmente acontece quando a temperatura que influencia o sistema é baixa.

Por fim, a energia funcional descrita na equação 3.9 também pode ser utilizada para especificar a interação.

5.2 A temperatura

Na seção anterior, foi destacado que a estabilidade é alcançada a baixas temperaturas. Porém o sistema é influenciado por flutuações térmicas e quando está sob temperaturas altas, os *spins* alternam seu estado, corrompendo, assim, o seu alinhamento. Se a temperatura for suficientemente alta as oscilações dominam e os *spins* se opõem aos seus campos.

Devido esta propriedade, a dinâmica determinística da equação 3.3 é substituída pela dinâmica de Glauber:

$$S_i = \begin{cases} 1 & \text{com a probabilidade } g(h_i); \\ -1 & \text{com a probabilidade } 1-g(h_i); \end{cases}$$

Equação 5.3- Dinâmica de Glauber

Onde a função sigmóide $g(h)$ é:

$$g(h) = f_{\hat{a}}(h) = \frac{1}{1 + \exp(-2\hat{a}h)}$$

Seja \hat{a} a temperatura absoluta T e k_B a constante de Boltzmann igual a 1.38×10^{-16} erg/K. O valor de \hat{a} é:

$$\hat{a} = \frac{1}{k_{\hat{a}} T}$$

A regra dinâmica de uma rede pode ser escrita em sua forma simétrica como:

$$\text{Prob}(S_i = \pm 1) = f_{\hat{a}}(\pm h_i) = \frac{1}{1 + \exp(\mp 2 \hat{a} h)}$$

Equação 5.4 – Dinâmica de Glauber

Desta equação pode-se perceber que a altas temperaturas (valor \hat{a} pequeno) $f_{\hat{a}}(h_i)$ passa de 0 para 1 quase em linha reta, ou seja, a subida do sigmóide de 0 para 1 é muito pequena. A baixa temperatura a subida do sigmóide de 0 para 1 é repentina, ou seja, sua distância do eixo h é pequena e proporcional a $k_B T$.

Contudo, no limite T tendendo para 0 e \hat{a} tendendo para infinito, $f_{\hat{a}}(h_i)$ se reduz a uma função determinística, $S_i = \text{sgn}(h_i)$. Caso contrário, o valor de S_i é completamente randômico.

Para aplicar esta regra, considere um *spin* único influenciado pelo campo externo e para descobrir a probabilidade (média de magnetização) do seu estado $\langle S \rangle$ é utilizada a equação 5.4. Note que o estado de um *spin* é igual a probabilidade de seu estado ser igual a 1 ou -1, pois, devido as flutuações térmicas, o estado não é mais determinístico.

O valor de $\langle S \rangle$ é igual a tangente hiperbólica de $\hat{a} h$ ($\tanh(\hat{a} h)$), que tem o mesmo tipo de $f_{\hat{a}}(h_i)$. Portanto, entre os valores do intervalo $] +1, -1[$ ($k_B T > 0$), o *spin* pode assumir o valor 1 ou -1 (ou seja, o estado do *spin* é inconstante).

Considerando um conjunto de *spins* e lembrando que para o valor de um *spin* h_i (a evolução de S_i) depende da evolução dos outros *spins* (as evoluções dos S_j 's). Para calcular o valor de h_i aplica-se a teoria do campo médio, que consiste em substituir a flutuação verdadeira de h_i por um valor médio.

A equação 5.5 descreve o valor médio do estado de um *spin*:

$$\langle S_i \rangle = \tanh(\hat{a} \langle h_i \rangle) = \tanh(\hat{a} \sum_j w_{ij} \langle S_j \rangle + h^{\text{ext}})$$

Equação 5.5 – Teoria do campo médio

O modelo de um sistema magnético pode ser exemplificado pelas propriedades do ferromagnético, onde todos os w_{ij} 's são positivos fazendo com que os *spins* alinhem para cima uns com os outros, enquanto flutuações térmicas, que ocorrem quando a temperatura está maior do que uma temperatura crítica T_c , tendem a corromper este alinhamento fazendo $\langle S \rangle = 0$.

À temperatura igual a zero, portanto, os valores de $\langle S_i \rangle$ são iguais (o sistema está em equilíbrio), correspondendo diretamente a um padrão armazenado em Hopfield ($\hat{t}_i = I$ para todo i). Para vários padrões armazenados, cada padrão corresponde à magnetização estável local.

5.3 A Rede Hopfield como uma rede estocástica

Como o modelo descrito na seção anterior é análogo a Rede de Hopfield, pode-se, considerar uma Rede Hopfield como uma rede estocástica, utilizando a idéia de temperatura e fazendo com que cada unidade da rede corresponda a uma unidade estocástica (com seu comportamento correspondente ao do *spin* [eq 3.5] e a escolha da unidade a ser atualizada é feita randomicamente).

Nas redes neurais biológicas, os neurônios ativam com pesos diferentes e ocorrem flutuações na liberação dos neurotransmissores atrasando a sinapse. Este fenômeno é equivalente as flutuações térmicas.

A variável \hat{a} (que a partir de agora é igual a $1/T$) nas redes estocásticas controla estas flutuações que também são conhecidas como ruído além de não permitir que a equação 3.1 seja violada.

A utilização deste tipo de rede é justificada, pois elas conseguem eliminar os estados espúrios que impedem que um padrão seja recuperado com sucesso.

Uma exigência das redes estocástica é que exista um valor mínimo de energia da rede. Todavia, isto já foi mostrado na seção anterior através do comportamento da energia funcional que sempre chega em um mínimo, se a rede tiver conexões simétricas. Logo, agora não se tem mais uma configuração do estados de todas as unidades i estáveis da rede, mas sim uma média dos estados em equilíbrio.

Para um número $p \ll N$, temos uma equivalência direta com a equação 5.5. Por hipótese, pela similaridade com o ferromagnético, pode-se analisar a média dos estados das unidades estocásticas, produzindo:

$$\hat{m}_i^v = \tanh \left(\frac{1}{N} \sum_{j, \hat{i}} w_{ij} \hat{i}^{\hat{i}} \hat{i}_j^{\hat{i}} \hat{m}_j^v \right)$$

Equação 5.6 – Média dos estado S_i

Considerando \hat{m}_i^v o estado de $\langle S_i \rangle$ em $T = 0$ ($m = 1$), esta equação pode ser analisada separando os termos onde $\hat{i} = \pm 1$ como em foi feito na equação 3.8. Se o número de padrões é pequeno, o *crosstalk* pode ser eliminado e temos que $m = \tanh(\hat{a}m)$ garantindo que os estados de memória serão estáveis a temperaturas menores que 1, ou seja, $T_c = 1$ para $p \ll N$.

6. Conclusão

Neste trabalho pode-se perceber o funcionamento da Rede Hopfield como uma memória associativa, pois, devido a seu dinamismo, a rede sempre alcança um valor mínimo da função de energia, que representa um padrão armazenado.

Como mostrado ao longo da monografia, a Rede de Hopfield é tolerante a falhas, ou seja, dado um padrão de entrada parecido com o padrão armazenado, ela consegue recuperar este padrão armazenado. Pode-se dizer que a rede artificial estudada possui características do cérebro, pois o cérebro também funciona como uma memória endereçada pelo conteúdo.

O que se deve destacar em relação ao seu funcionamento é a quantidade de estados espúrios recuperados pela rede no lugar de um padrão armazenado. Baseando-se nas características da rede estes não são estados errados, o que acontece é que a rede não desenvolve em direção do padrão desejado. Estes estados também fazem parte do conjunto de mínimos da função de energia da rede, portanto este fato deve ser considerado e contabilizado perante a aplicação para que a rede será projetada.

Os estudo das redes neurais prosseguirá até o entendimento minucioso da Máquina de Boltzmann que possui características das Redes de Hopfield associadas com as características dos mecanismos estatísticos apresentadas no capítulo 5.

Além do estudo das redes neurais será feito em paralelo um estudo do algoritmo distribuído apresentado e qual seriam as modificações deste algoritmo para que os neurônios tenham uma hierarquia em suas conexões contando com as características da Máquina de Boltzmann.

7. Bibliografia

- [1] Hertz J., Krogh A. e Palmer R. G. “*Introduction to the Theory of Neural Computation*”. Addison – Wesley Publishing Company, 1991.
- [2] Jain A. K. e Mao J. “*Artificial Neural Networks : A tutorial*”. Computer, volume 29 número 3, Mar 1966.
- [3] Langkjer A. S. e Marins E. R. “*Estudo de Redes Neurais na Implementação de Memórias Associativas*”. Departamento de Informática, UERJ.
- [4] Babosa V. C. “*Massively Parallel Models of Computation: Distributed Parallel Processing in Artificial Intelligence and Optimization*”. Ellis Horwood Limited, West Sussex, Inglaterra 1993.
- [5] Wedmann, R.S. et al. *Memory Functioning in Psychopathology*. IME, Universidade do Estado do Rio de Janeiro.
- [6] Vilela A. L. M. “*Anatomia e Fisiologia Humana – Coordenação e Regulação*”. Disponível em: <<http://www.afh.bio.br/basicos/Nervoso1.htm>>, Junho de 2004.
- [7] Ludeig A. e Canpolim S. “*Esquecer é tão bom quanto lembrar*”. Revista Época/Nacional, 25 de janeiro de 1999.
- [8] Menezes C. C. “*Redes Neurais*”. Instituto de Informática, Universidade do rio Grande do Sul, outubro de 2001.

8. Apêndice

O código desenvolvido durante o estudo foi escrito na linguagem C e é apresentado no ANEXO I.

Os arquivos de testes possuem a seguinte forma: a primeira linha contém o número de neurônios da rede, a segunda o número de padrões a serem armazenados e por fim os padrões a serem armazenados. Um script de algumas execuções (de uma rede com 30 neurônios) é apresentado no ANEXO II. Analisando os resultados pode-se entender melhor o dinamismo da rede discutido ao longo desta monografia. Apesar de nem sempre a rede responder com o padrão esperado ela sempre alcança um mínimo da função de energia.