

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/228588719>

Sistemas Inteligentes baseados em redes neurais artificiais aplicados ao processamento de imagens

Article

CITATION

1

READS

2,763

3 authors, including:



Fernando Osorio

University of São Paulo

151 PUBLICATIONS 581 CITATIONS

SEE PROFILE

I WORKSHOP DE INTELIGÊNCIA ARTIFICIAL
UNISC – Universidade de Santa Cruz do Sul
Departamento de Informática- Junho 2000

Seminário

**Sistemas Inteligentes baseados em
Redes Neurais Artificiais aplicados ao
Processamento de Imagens**

Autores: Fernando Osório & João Ricardo Bittencourt

Apresentador:

Prof. Dr. Fernando Santos Osório

E-mail: osorio@exatas.unisinos.br

Web : <http://www.inf.unisinos.br/~osorio/>



Universidade do Vale do Rio dos Sinos
Centro de Ciências Exatas e Tecnológicas – Centro 6/6
Curso de Informática / Mestrado em Computação Aplicada
[Http://www.inf.unisinos.br/](http://www.inf.unisinos.br/)

1. Introdução

Este trabalho tem por objetivo apresentar uma introdução aos principais conceitos sobre o uso de Sistemas de Inteligência Artificial baseados em Redes Neurais Artificiais, destacando-se a sua aplicação em problemas ligados ao Processamento de Imagens, tais como: criação de filtros inteligentes e classificação automática de padrões.

As Redes Neurais Artificiais (RNAs) são ferramentas de Inteligência Artificial que possuem a capacidade de se adaptar e de aprender a realizar uma certa tarefa, ou comportamento, à partir de um conjunto de exemplos dados. A aplicação das RNAs junto a tarefas de processamento de imagens torna-se bastante atrativa dadas as características deste tipo de ferramentas, tais como: robustez, generalização, paralelismo e tolerância ao ruído.

Vamos analisar neste trabalho alguns modelos de redes neurais e suas propriedades, sendo então demonstradas algumas das principais aplicações destas redes neurais junto ao Processamento de Imagens em tarefas como:

- Reconhecimento de padrões: Sistemas OCR (Optical Character Recognizer), reconhecimento de assinaturas e de impressões digitais;
- Classificação de padrões: controle de qualidade, classificação e agrupamento de cenas/imagens com propriedades similares;
- Tratamento de imagens: filtros inteligentes usados na melhoria da qualidade visual das imagens (ressaltar elementos, detectar contornos, corrigir o foco/distorções);
- Visão artificial: tarefas de alto nível, como o reconhecimento de pessoas, de expressões faciais, de gestos, de elementos e de objetos presentes em um ambiente, etc.

Por fim, vamos analisar as principais vantagens do uso das RNAs neste tipo de aplicações, bem como as futuras tendências nesta área de pesquisas.

2. Inteligência Artificial e Aprendizado

Para podermos falar em Inteligência Artificial e Aprendizado Artificial, precisamos antes saber o que é “*inteligência*” e o que entendemos por “*aprendizado*”? Apesar de termos uma noção básica do que significam estas duas palavras, inteligência e aprendizado, temos uma grande dificuldade de defini-las em termos práticos e de forma bastante precisa.

2.1. O que é Inteligência?

O que é inteligência? O que é um ser inteligente? Estas duas questões devem ser respondidas (ao menos de forma superficial) se quisermos então partir para a implementação dos sistemas inteligentes. A idéia de implementação dos sistemas inteligentes parte do pressuposto que iremos “copiar” a inteligência humana e colocá-la a nosso serviço através de implementações que automatizem este tipo de processo/comportamento.

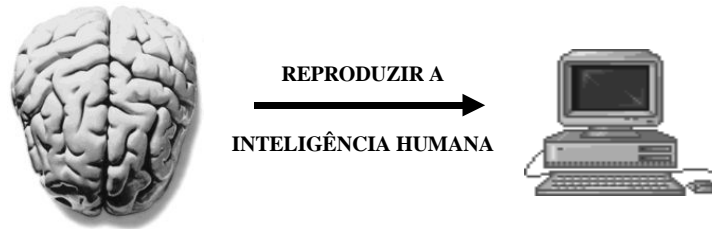


Figura 2.1 – Inteligência: Natural e Artificial

O termo “*Artificial Intelligence*” (A.I. – I.A. em português) foi usado pela primeira vez em 1956 por McCarthy (e desenvolvido por grande pesquisadores como Marvin Minsky e Herbert Simon), e nada mais é do que uma tentativa de formalizar o eterno sonho da criação de um “cérebro eletrônico” (termo muito usado na ficção científica e mesmo na época inicial do desenvolvimento dos computadores). Encontramos também algumas definições interessantes na literaturas, tais como:

- (1) Definição de Inteligente: Dotado de inteligência, capaz de compreender, esperto, habilidoso [Larousse 99];
- (2) Definição de Inteligência: Faculdade de conhecer, de aprender, de conceber, de compreender: a inteligência distingue o homem do animal [Larousse 99];
- (3) Definição de Inteligência Artificial: Conjunto de teorias e de técnicas empregadas com a finalidade de desenvolver máquinas capazes de simular a inteligência humana [Larousse99];
- (4) Definição de Inteligência Artificial: A Inteligência Artificial é uma área de estudos da computação que se interessa pelo estudo e criação de sistemas que possam exibir um comportamento inteligente e realizar tarefas complexas com um nível de competência que é equivalente ou superior ao de um especialista humano [Nikolopoulos 97].

Destas definições acima, podemos fazer os seguintes comentários:

- As definições são usualmente recursivas, ou seja, um ser inteligente é aquele que possui inteligência, e a inteligência é a característica dos seres inteligentes;
- As definições possuem contradições, onde aparece que a inteligência distingue o homem do animal, pois só o homem é capaz de aprender, conhecer, compreender. Isto não é muito preciso/correto, pois os animais também podem aprender (a reconhecer o dono), conhecer (o caminho de volta para casa) e compreender (um comando de seu mestre). E podemos mesmo dizer que estas tarefas realizadas por animais são tipicamente tarefas consideradas na atualidade como “tarefas inteligentes” que os computadores ainda não realizam com um desempenho satisfatório.

- Uma pessoa com limitações (analfabeta, deficiente mental e/ou físico, com QI reduzido) poderá ser qualificada, pelos critérios acima, como inteligente? E no entanto ela é um ser inteligente com capacidades superiores as de um computador, ou você acha que, por exemplo, um computador conseguiria ir até uma farmácia comprar um remédio? (isto inclui tarefas de: planificação de trajetória, reconhecimento do caminho, tratamento de imprevistos, etc).

Esta discussão poderia se prolongar aqui de forma indefinida, mas acredito que um ponto importante a ser considerado é: se queremos reproduzir a inteligência humana, devemos ter consciência de quais aspectos da inteligência que gostaríamos que fossem reproduzidos, e que em nossa opinião, permitiriam o desenvolvimento de um sistema com propriedades ditas “inteligentes”. Esta pergunta foi feita em sala de aula para um conjunto de alunos da Unisinos, e o resultado foi bastante interessante, onde reproduzo aqui as respostas de forma a complementar esta discussão: O que é Inteligência?

- Associação de idéias e conceitos;
- Concluir coisas;
- Reconhecer e classificar coisas;
- Capacidade de aprendizado;
- Acúmulo de conhecimentos;
- Raciocínio: lógico, abstrato, dedução, analogia, indução, inferência, síntese, análise, ...;
- Uso de experiências e conhecimentos passados;
- Uso prático de conhecimentos e experiências;
- Tomada de decisões;
- Criar coisas novas (criatividade);
- Saber o que eu sei (capacidade de explicar algo);
- Saber que sei / Saber que não sei;
- Interação;
- Comunicação;
- Capacidade de interagir com o mundo através dos cinco sentidos humanos: *visão, audição/fala, paladar, olfato e tato.*

... (espaço reservado para **você** continuar esta reflexão!) ...

Vamos aqui usar a definição de inteligência (que nos convêm) que é baseada no seguinte ditado: “*Errar é humano, mas repetir o erro é burrice*”. Logo o conceito de inteligência (por oposição à burrice, e devido ao fato desta ser uma propriedade humana), indica que o ser humano (ou não) inteligente deve ser capaz de: adaptar-se, aprender, evoluir, corrigir seus erros. E assim estamos passando da questão “o que é um ser inteligente” para a questão seguinte “o que é aprendizado”.

2.2. O que é Aprendizado?

O *aprendizado natural* é também um conceito difícil de ser definido, mas também de grande importância para que possamos partir para a construção de sistemas inteligentes dotados da capacidade de aprendizado. Sugerimos ao leitor que faça aqui o uso de sua *inteligência natural* a fim de definir o que você entende por *aprendizado natural* e quais são as propriedades que estão associadas a este conceito. Depois sugerimos ao leitor que, de posse dessa sua definição/descrição, passe a analisar os próximos itens abordados neste texto, com um olhar crítico sobre o mesmo no que diz respeito a presença ou não destas propriedades associadas aos métodos que iremos apresentar.

Vamos tentar definir aqui brevemente o que entendemos (ou melhor, o que o **autor** entende) por aprendizado. Aprendizado é a capacidade de se adaptar, de modificar e melhorar seu comportamento e suas respostas, sendo portanto uma das propriedades mais importantes dos seres ditos inteligentes, sejam eles humanos ou não. A capacidade de aprender está ligada diretamente aos seguintes itens:

- **Adaptação** e mudança de comportamento de forma a evoluir (melhorar segundo algum critério). Um sistema, biológico ou artificial, que não seja capaz de evoluir ou de mudar seu comportamento diante de novas situações que lhe são propostas é um sistema sem inteligência;
- **Correção** dos erros cometidos no passado, de modo a não repeti-los no futuro. Este item está diretamente relacionado ao anterior, o sistema deve modificar seu comportamento caso o comportamento atual não satisfaça a algum tipo de exigência (onde a sobrevivência deve ser um dos quesitos mais importantes a serem considerados nos seres vivos);
- **Otimização**: melhoria da performance do sistema como um todo. O aprendizado pode implicar em uma mudança do comportamento que busque: a economia de energia gasta para realizar uma tarefa, a redução do tempo gasto numa tarefa, etc. Quando falamos em otimização, devemos lembrar que podemos ter quesitos contraditórios e opostos, onde teremos que maximizar ou minimizar custos de acordo com algum tipo de critério;
- **Interação** com o meio, pois é através deste contato com o mundo que nos cerca que podemos trocar experiências e/ou realizar experiência, de forma a adquirir novos conhecimentos;
- **Representação** do conhecimento adquirido. O sistema deve ser capaz de armazenar uma massa muito grande de conhecimentos, e isto requer uma forma de representar estes conhecimentos que permita ao sistema explorá-los de maneira conveniente. Como nossos recursos são limitados, devemos ter uma maneira de guardar conhecimentos e regras gerais, pois guardar tudo seria muito difícil (exige muita memória, dificulta o uso pela lentidão da consulta aos conhecimentos).

O aprendizado é um tema bastante interessante e amplo, onde insisto que o autor deste texto irá centrar a noção de inteligência no que diz respeito a capacidade de um sistema aprender. Uma característica básica das redes neurais é justamente esta possibilidade de se adaptar e de aprender.

2.3. Inteligência Artificial – Aprendizado de Máquinas

As ferramentas de I.A., em sua evolução, começaram adquirindo conhecimentos que eram explicitados pelos especialistas de uma certa área (o que equivale a programar um sistema para resolver um problema). Posteriormente estas ferramentas foram dotadas de mecanismos de aquisição automática de conhecimentos, onde podemos citar o exemplo da linguagem Prolog e dos sistemas especialistas (2ª. Geração) que foram dotados de mecanismos de aprendizado de máquinas – **machine learning**. Os principais métodos (ou pelos menos os mais conhecidos) de aprendizado de máquinas são:

- Aprendizado por analogia/por instâncias. Exemplo: Sistemas baseados em casos - CBR (*Case Based Reasoning*) [Mitchell 97, Kolodner 93];
- Aprendizado por Indução. Exemplos: Árvores de Decisão - ID3, C4.5, CN2 (IDT - *Induction of Decision Trees*) [Quinlan 92], e ILP - *Inductive Logic Programming (Prolog)* [Nilsson 98].

- Aprendizado por evolução/seleção. Exemplo: Algoritmos Genéticos - GA e GP (*Genetic Algorithms / Genetic Programming*) [Goldberg 89, Mitchell 97];
- Outros tipos de aprendizado: por reforço (*reinforcement learning*), não supervisionado, bayesiano e por explicações (*explanation based*) [Mitchell 97, Nilsson 98].

Estas ferramentas de aprendizado possuem fortes limitações: (1) usualmente assumem que os conhecimentos a serem adquiridos/manipulados e as informações disponíveis são corretos e devem estar completos em relação ao problema (o que dificilmente ocorre); (2) são orientadas para manipular informações simbólicas (informações qualitativas), onde os valores numéricos e contínuos (informações quantitativas) representam um problema difícil de ser tratado. Em relação a este último item, existem tentativas de integrar ao processamento simbólico com dados probabilísticos (regras bayesianas) e com uso de fatores de incerteza (regras fuzzy), como forma de expandir as potencialidades deste tipo de sistemas.

2.4. Limitações do aprendizado baseado em processamento simbólico e alternativas

O processamento e aprendizado simbólico, devido as suas características básicas, possuem algumas limitações no que diz respeito a manipulação: de incertezas, de valores aproximados, de informações contraditórias, e de uma maneira geral, de informações quantitativas [Minsky 90]. O mundo que nos cerca é extremamente “impreciso” do ponto de vista computacional, e sendo assim, as ferramentas computacionais de I.A. devem portanto ser bastante robustas para poderem trabalhar neste ambiente de informações imprecisas (informações incorretas) e cheio de problemas imprevisíveis (informações incompletas).

Muitos pesquisadores, na sua busca da implementação de ferramentas e sistemas inteligentes, se orientaram para os sistemas baseados “realmente” no funcionamento do cérebro humano. Este tipo de enfoque levou muitos pesquisadores (entre eles Marvin Minsky, Seymour Papert e John Von Neumann) a estudarem novas formas de processamento de informações baseadas nos estudos neuro-fisiológicos do cérebro. Esta corrente de pesquisas tentava reproduzir os neurônios como elementos básicos do processamento de novas arquiteturas de máquinas inteligentes, ao invés de usar portas lógicas, bits e bytes controlados por uma Unidade Central.

Esperava-se que a partir de elementos de processamento simples capazes de operar em paralelo, (baseados numa rede de neurônios ligados por um grande número de conexões entre eles), se pudesse obter como resultado a “emergência de um comportamento inteligentes”. Este ramo da I.A. foi denominado de *Inteligência Artificial Conexionista*, ou estudo das *Redes Neurais Artificiais*. A I.A. conexionista de certa forma se opõe a I.A. clássica e baseada em processamento simbólico.

Conhecimento \leq Aquisição de Conhecimentos : Explicitação Manual

Aprendizado Automático

Aprendizado Simbólico: IDT, C4.5, ILP, CBR, Algoritmos Genéticos (em geral)

Aprendizado Conexionista: Redes Neurais Artificiais

3. Redes Neurais Artificiais

3.1. Introdução

As Redes Neurais Artificiais (RNA), também conhecidas como métodos conexionistas, são inspiradas nos estudos da maneira como se organiza e como funciona o cérebro humano. Este tipo de método possui características peculiares de representação e de aquisição de conhecimentos, sendo considerado um método de nível sub-simbólico (em oposição aos métodos simbólicos, citados na seção anterior: árvores de decisão, CBR, KBS, etc).

Inicialmente vamos discutir sobre a representação de conhecimentos utilizada pelas Redes Neurais, para depois analisarmos a parte referente ao aprendizado destas. É importante salientar que existem diferentes tipos de RNAs e que cada um destes modelos diferentes possui suas características próprias quanto a forma de representar e de adquirir conhecimentos. Em função disto vamos primeiramente apresentar uma visão geral, classificando os diferentes modelos de RNAs para em seguida nos concentrarmos em um modelo mais específico: as redes neurais artificiais do tipo multi-nível baseada em Perceptrons (conhecidas como Multi-Layer Perceptron Nets ou Back-Propagation Nets). A figura 3.1 apresenta um exemplo deste modelo de rede conexionista.

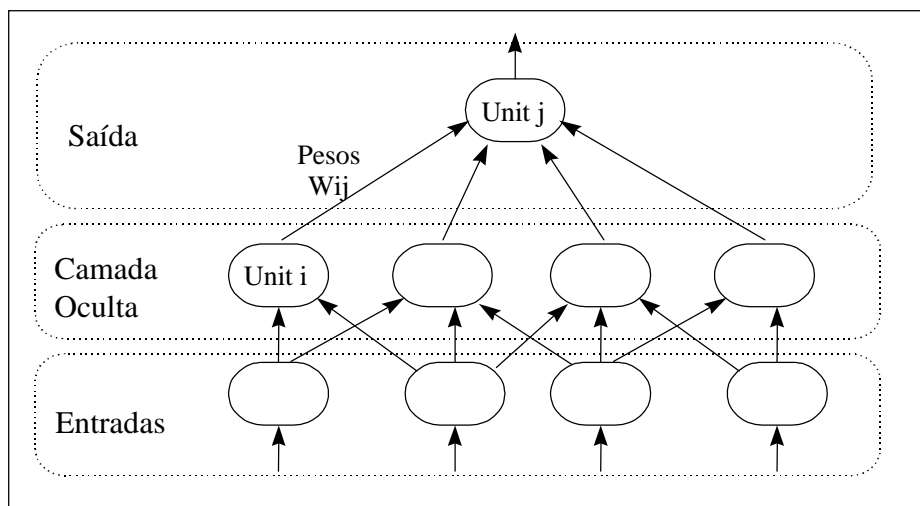


Figura 3.1. Exemplo de Rede Neural Artificial do tipo Multi-Nível (MLP)

3.2. Conceitos Básicos sobre as Redes Neurais

Vamos discutir inicialmente como se estruturam as redes neurais e seus neurônios, passando depois para uma apresentação sobre o seu funcionamento.

3.2.1. Representação de Conhecimentos

A representação de conhecimentos nas redes conexionistas, como diz o próprio nome, é fortemente ligada a noção de conexão entre neurônios (elementos processadores de informação) que interagem uns com os outros através destas ligações. O modelo conexionista possui sua origem nos estudos feitos sobre as estruturas de nosso cérebro – *sofrendo uma grande simplificação do modelo original* – onde encontramos no modelo artificial, que é simulado, elementos como os neurônios e as suas conexões, chamadas de sinapses. A proximidade e fidelidade dos modelos artificiais em relação ao modelo real é um *tema polêmico* que não vamos nos aventurar a discutir aqui, deixando apenas o registro da origem dos conceitos básicos que norteiam este campo de pesquisas.

O conhecimento de uma RNA está codificado na estrutura da rede, onde se destacam as conexões (sinapses) entre as unidades (neurônios) que a compõe. Nestas redes artificiais obtidas por simulação em computadores, associamos a cada conexão um peso sináptico (valor numérico) que caracteriza a força da conexão entre dois neurônios. O aprendizado em uma RNA é realizado por um processo de adaptação dos seus pesos sinápticos. As figuras 3.2 e 3.3 mostram a relação entre os neurônios naturais e o modelo de neurônio artificial.

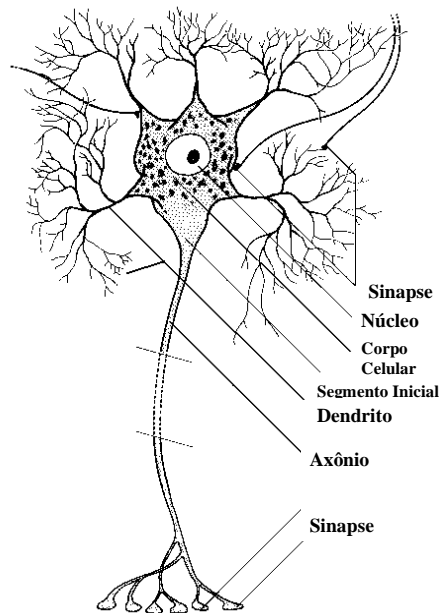


Figura 3.2. Exemplo de Neurônio Natural

Uma vez que os conhecimentos da rede estão codificados na estrutura de interconexões entre os neurônios e nos pesos associados a estas conexões, fica muito difícil para um ser humano realizar uma análise e interpretação dos conhecimentos adquiridos por uma RNA. Os conhecimentos das redes se resumem a um conjunto de valores numéricos descrevendo as conexões, e por consequência, estes valores descrevem também o comportamento da rede. Entretanto, para um ser humano estes dados não fazem muito sentido. Na seção 3.3 vamos apresentar mais em detalhes os diferentes tipos de redes, onde a escolha do tipo de neurônio artificial utilizado é fundamental para se definir como será a representação interna dos conhecimentos da rede.

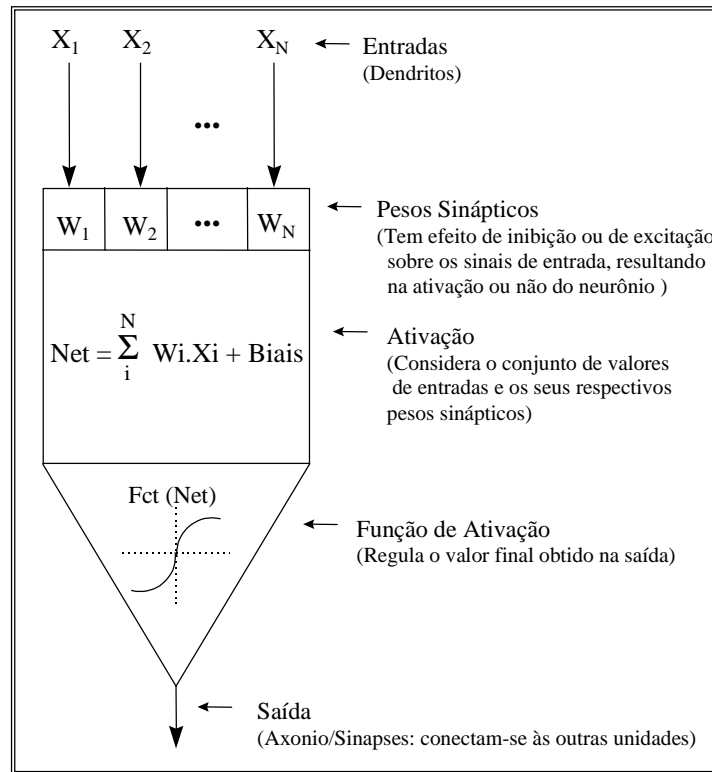


Figura 3.3. Exemplo de Neurônio Artificial

Apesar dos estudos sobre as redes neurais artificiais serem considerados como pertencentes à uma área “jovem”¹ de pesquisa, encontramos atualmente uma série de referências e exemplos de importantes aplicações práticas deste tipo de método de aprendizado, onde podemos citar alguns exemplos de obras relevantes na área, como por exemplo [Kovacs 97, Fiesler 97, Ripley 96, Bishop 95, Arbib 95, Krose 93, Freeman 92, Simpson 90, Faq 99a].

3.2.2. Origem das redes neurais

Os primeiros estudos sobre as Redes Neurais Artificiais e propostas de modelos destas redes surgiram nos anos 40. Os primeiros modelos evoluíram bastante, onde alguns deles se destacaram e tornaram-se famosos, mas mesmo assim até hoje continuam sendo propostos novos modelos de redes neurais. O estudo dos primeiros modelos e de sua evolução nos ajuda a entender melhor as redes neurais, e seu estágio atual de evolução.

O começo do estudo das rede neurais artificiais pode ser atribuído à criação do *Psychon* em 1943 por McCulloch e Pitts [McCulloch 43], sendo que alguns anos mais tarde, em 1949 D. O. Hebb publicava uma importante obra, o livro “*The Organization of Behaviour*” [Hebb 49], que influenciou vários modelos de RNAs de destaque na atualidade.

¹ Os primeiros estudos sobre as redes neurais artificiais remontam aos anos 40 (McCulloch-Pitts), mas foi somente na década de 80 que houve um grande desenvolvimento nesta área (Back-Propagation, Hopfield, Kohonen SOFM, ...)

Em 1959, Frank Rosenblatt criou o *Perceptron* [Rosenblatt 59], que deu posteriormente origem aos modelos denominados de MLP. Alguns outros modelos similares ao Perceptron foram também desenvolvidos nesta época, como é o caso do *Adaline* (Adaptive Linear Element), criado por Bernard Widrow em 1962 [Widrow 62, 88, 90]. Os modelos do tipo Perceptron, incluindo o Adaline, são baseados no aprendizado supervisionado por correção de erros, uma classe muito importante de redes neurais artificiais, que possui uma larga aplicação na atualidade.

Em 1969 os modelos baseados no Perceptron receberam uma dura crítica feita por Minsky e Papert através de sua obra “*Perceptrons: An Introduction to Computational Geometry*” [Minsky 69]. Através deste livro, Minsky e Papert provaram matematicamente que os modelos de redes neurais baseados no Perceptron (redes de um só nível, o que na época era o tipo de rede de Perceptrons utilizado), **não** eram capazes de aprender uma simples função lógica do tipo “ou-exclusivo” (XOR = *Exclusive Or*). A função XOR possui um padrão de valores de entrada e de saída cuja associação não podia ser aprendida pelos modelos de redes baseados em Perceptron disponíveis naquela época. O impacto da publicação deste livro abalou profundamente as pesquisas realizadas nesta área de estudos.

Somente na década de 80, surgiram novos modelos que deram um novo impulso às redes neurais. Em 1982 surgiu um modelo importante de rede criado por J. Hopfield [Hopfield 82], onde este modelo começou a dar um novo impulso às redes neurais. O modelo que Hopfield criou era baseado em um tipo de rede diferente dos modelos baseados no Perceptron, sendo uma rede com conexões recorrentes e com um comportamento baseado na competição entre os neurônios, onde o aprendizado era não supervisionado. Outros modelos similares ao modelo de Hopfield surgiram pouco depois, onde podemos citar alguns como por exemplo: a máquina de Boltzmann [Hinton 84] e o BAM (*Binary Associative Memory*) [Kosko 87, 87a].

A década de 80 ficou também marcada profundamente pelo reaparecimento das redes baseadas em Perceptrons. Isto deveu-se ao desenvolvimento dos computadores, que eram mais rápidos e permitiam realizar melhores simulações das redes neurais, bem como o desenvolvimento de modelos matemáticos que permitiram a solução do problema apontado por Minsky e Papert. Também podemos associar em parte este renascimento das redes neurais ao suposto desencanto com a I.A. clássica. O modelo que permitiu o ressurgimento das redes baseadas em Perceptrons foi o das redes multi-nível, onde o novo algoritmo de aprendizado chamado *Back-Propagation* resolveu em grande parte os problemas de aprendizado existentes até então. Este modelo foi desenvolvido por diferentes pesquisadores quase ao mesmo tempo, como D. Parker [Parker 82] e D. Rumelhart [Rumelhart 85], mas foi Rumelhart e Hinton que o tornaram este algoritmo famoso com a sua obra “*Parallel Distributed Processing - PDP*” [Rumelhart 86]. Este algoritmo, o *Back-Propagation* permitia realizar o aprendizado por correção de erros em uma rede com múltiplas camadas (níveis) e consequentemente resolveria o problema do XOR.

Além dos modelos de Hopfield e do modelo de redes multi-nível com *Back-Propagation* (chamado de *Multi-Layer Perceptron – MLP*), outro modelo importante que surgiu nesta década foi o modelo de Teuvo Kohonen [Kohonen 82, 87]. O modelo de Kohonen é muito interessante pois permite o aprendizado competitivo com uma auto-organização da rede neural, criando os chamados “mapas de atributos auto-organizáveis” (*self-organizing feature maps*).

Por fim, o último modelo de destaque neste período, foi o modelo ART (*Adaptive Resonance Theory*) criado por Gail Carpenter e Stephen Grossberg [Carpenter 93]. Este modelo possui um aprendizado do tipo não supervisionado, criando protótipos (*clusters*) dos padrões aprendidos. O modelo ART teve diversas versões posteriores, entre elas versões do tipo semi-supervisionado e com uso de conceitos da lógica nebulosa (Fuzzy-ART).

Os estudos sobre as redes neurais sofreram uma grande revolução a partir dos anos 80, conforme foi demonstrado acima. E, a partir dos anos 80, cada vez mais, esta área de estudos tem se destacado, seja pelas promissoras características apresentadas pelos modelos de redes neurais propostos, ou seja pelas condições tecnológicas atuais de implementação que permitem desenvolver arrojadas implementações de arquiteturas neurais paralelas em hardwares dedicados, obtendo assim ótimas performances destes sistemas (bastante superiores aos sistemas convencionais).

3.3. Modelos Conexionistas

3.3.1. Definição de uma Rede Neural Artificial

As *redes conexionistas* são formadas por um conjunto de unidades elementares de processamento de informações fortemente conectadas, que denominamos de neurônios artificiais. Uma RNA é constituída por um *grafo orientado e ponderado*. Os nós deste grafo são *autômatos simples*, os chamados *neurônios artificiais*, que formam através de suas conexões um autômato mais complexo, a rede neural, também conhecida como rede conexionista.

Cada unidade da rede é dotada de um estado interno, que nós vamos denominar de *estado de ativação*. As unidades podem propagar seu estado de ativação para as outras unidades do grafo, passando pelos arcos ponderados, que nós chamamos de *conexões*, *ligações sinápticas* ou simplesmente de *pesos sinápticos*. A regra que determina a ativação de um neurônio em função da influência vinda de suas entradas, ponderadas pelos seus respectivos pesos, se chama *regra de ativação* ou *função de ativação*.

As mudanças realizadas nos valores dos pesos sinápticos ou na estrutura de interconexão das unidades de uma rede, são responsáveis pelas alterações no comportamento de ativação desta rede. Estas alterações nas conexões e na estrutura da rede é o que nos permite realizar o *aprendizado* de um novo comportamento. Desta maneira vamos poder modificar o estado de ativação na saída da rede em resposta a uma certa configuração de entradas. Portanto, a rede é capaz de estabelecer associações de entrada-saída (estímulo e resposta) a fim de se adaptar a uma situação proposta. No caso de uma rede com aprendizado supervisionado (vide item sobre tipos de aprendizado), a rede deve adaptar os seus pesos de maneira à passar a responder de acordo com o exemplo dado, ou seja, gerando na sua saída um estado de ativação compatível para com o esperado. O método utilizado para modificar o comportamento de uma rede é denominado de *regra de aprendizado*.

A grande quantidade de modelos de redes conexionistas existentes torna difícil para nós a descrição exaustiva de todos eles. Se o leitor assim desejar, poderá se aprofundar em maiores detalhes sobre os diferentes modelos de RNAs em obras como o “Handbook of Neural Computation”[Fiesler 97]. Nós iremos nos concentrar aqui em diferenciar estes modelos, tomando como base as suas principais características.

3.3.2. Classificação e Propriedades

A grande quantidade de modelos existentes nos leva à uma análise de suas principais propriedades e diferenças em detrimento de uma análise caso à caso mais detalhada. Este estudo das principais propriedades das redes neurais nos permite compreender melhor as vantagens e/ou inconvenientes da escolha de um modelo em detrimento de um outro. Consideramos que não existe apenas uma maneira de classificar todos os modelos, mas de um modo geral devem ser considerados grupos de atributos, tais como: tipo de aprendizado, arquitetura de interconexões, forma interna de representação das informações, tipo de aplicação da rede, etc. Caso o leitor tenha o interesse de buscar uma proposta mais formal de classificação das redes neurais, esta pode ser encontrada em [Fiesler 97].

3.3.2.1. Aprendizado Conexionista

O aprendizado conexionista é em geral um processo gradual e iterado, onde os pesos são modificados várias vezes, pouco à pouco, seguindo-se uma regra de aprendizado que estabelece a forma como estes pesos são alterados. O aprendizado é realizado utilizando-se um conjunto de dados de aprendizado disponível (base de exemplos). Cada iteração deste processo gradativo de adaptação dos pesos de uma rede neural, sendo feita uma apresentação completa do conjunto de dados, é chamada de *época de aprendizado*. Os métodos de aprendizado neural podem ser divididos em três grandes classes, segundo o grau de controle dado ao usuário

- **Aprendizado supervisionado:** o usuário dispõe de um comportamento de referência preciso que ele deseja ensinar a rede. Sendo assim, a rede deve ser capaz de medir a diferença entre seu comportamento atual e o comportamento de referência, e então corrigir os pesos de maneira a reduzir este erro (desvio de comportamento em relação aos exemplos de referência). Exemplo de aplicação: reconhecimento de caracteres em uma aplicação do tipo OCR (*Optical Character Recognition*) [Osório 91].
- **Aprendizado semi-supervisionado:** o usuário possui apenas indicações imprecisas (por exemplo: sucesso/insucesso da rede) sobre o comportamento final desejado. As técnicas de aprendizado semi-supervisionado são chamadas também de aprendizado por reforço (*reinforcement learning*) [Sutton 98]. Exemplo: aplicações em robótica autônoma, onde supondo uma situação hipotética, sabemos que seguir em frente não é possível pois existe um obstáculo, mas em compensação não temos uma medida numérica que indique para que lado seguir e exatamente como devemos proceder para desviar deste obstáculo.
- **Aprendizado não-supervisionado:** os pesos da rede são modificados em função de critérios internos, tais como, por exemplo, a repetição de padrões de ativação em paralelo de vários neurônios. O comportamento resultante deste tipo de aprendizado é usualmente comparado com técnicas de análise de dados empregadas na estatística (e.g. *clustering*). Exemplo: diferenciar tomates de laranjas, sem no entanto ter os exemplos com a sua respectiva classe etiquetada (e.g. *self-organizing feature maps* [Kohonen 87]).

O aprendizado conexionista em geral precisa de uma grande quantidade de dados, que nós agrupamos em uma *base de aprendizado*. Exemplos de bases de aprendizado podem ser encontrados na Internet no *UCI-ML repository* [UCI 99]. De acordo com a técnica de aprendizado utilizada, outros conjuntos de dados podem também ser necessários, principalmente para que se possa medir a validade do aprendizado realizado pela rede (e.g. *cross-validation* [Krogh 95]). Este conjunto de dados complementar é usualmente chamado de conjunto de *teste de generalização*. A figura 3.4 apresenta um gráfico típico da evolução do erro durante o aprendizado de uma rede neural, comparando a curva do erro (aprendizado supervisionado) referente à base de aprendizado com a curva do erro da base de teste de generalização.

Nós chamamos de *generalização* a capacidade de um modelo de aprendizado responder corretamente aos exemplos que lhe são apresentados, sendo que estes exemplos *NÃO* devem estar presentes na base de aprendizado. Um modelo que tem uma boa generalização é aquele modelo que responde corretamente aos exemplos contidos na base de aprendizado, mas também a outros exemplos diferentes daqueles da base de aprendizado, e que estão contidos em uma base de teste. A capacidade de generalizar é a principal capacidade buscada nas tarefas que envolvem aprendizado.

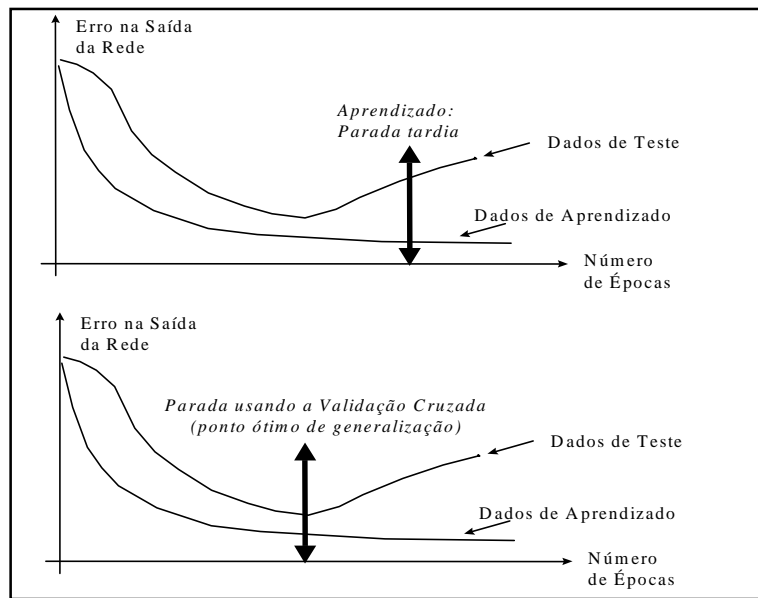


Figura 3.4. Aprendizado: Erro em relação ao conjunto de aprendizado e de teste

Uma rede pode se especializar demasiadamente em relação aos exemplos contidos na base de aprendizado. Este tipo de comportamento vai nos levar a um problema de aprendizado conhecido como super-aprendizado (*over-training* / *over-fitting*). Normalmente o *over-fitting* pode ser detectado/evitado através do uso de um teste de generalização por validação cruzada (*cross-validation*).

A adaptação/otimização dos pesos também pode ser implementada por diferentes métodos, segundo o tipo de regra de aprendizado que for empregado. As regras de aprendizado² mais usadas são [Jodoin 94, Caudill 92, Simpson 90, Faq 99]:

- Métodos de correção do erro, tais como a descida de uma superfície de erro baseada no gradiente. Exemplos de modelos deste tipo: Adaline, Perceptron, Back-Propagation, Cascade-Correlation;
- Métodos de aprendizado por reforço. Exemplos: Driver-Reinforcement Learning, AHC;
- Métodos de aprendizado por competição ou por auto-organização. Exemplos: Kohonen Self-Organizing Feature Maps, ART1;
- Métodos de aprendizado através da criação de protótipos ou *clusters*. Exemplos: RBF, ART1, ARN2;
- Métodos de aprendizado baseados em memórias associativas (auto-associativas ou hetero-associativas). Exemplos: Modelo de Hopfield, BAM.
- Métodos de aprendizado de seqüências temporais (redes recorrentes). Exemplos: SRN, BPTT, RTRL.

² Maiores detalhes sobre a implementação de algoritmos de aprendizado neural podem ser encontradas nas seguintes obras [Osório 91, 92, 98] e na Internet em <http://www.inf.unisinos.br/~osorio/neural.html>

3.3.2.2. Tipos de Unidades

As unidade de uma rede – os neurônios artificiais – podem ser de diferentes tipos, de acordo com a função interna utilizada para calcular o seu estado de ativação. As principais diferenças são relativas ao tipo de função de ativação utilizada (e.g. linear, sigmoide assimétrica (exp), sigmoide simétrica (tanh), gaussiana, etc) [Jodoin 94a, Jodoin 94b]. Outro elemento importante diz respeito a forma como os neurônios armazenam as informações: unidades baseadas em protótipos, unidades do tipo Perceptron. Vamos diferenciar aqui estes dois tipos de maneiras de representar o conhecimento nas unidades de uma rede.

- **Redes à base de protótipos:** este tipo de rede utiliza neurônios que servem para representar protótipos dos exemplos aprendidos – as unidades tem uma representação interna que agrupa as características comuns e típicas de um grupo de exemplos [Orsier 95]. As redes baseadas em protótipos tem normalmente um aprendizado não supervisionado (com um ou mais protótipos associados à cada classe) e podem facilmente adicionar novos protótipos na rede. Os protótipos são também denominados de clusters, onde apresentamos um exemplo de rede a base de protótipos na figura 3.5. Este tipo de redes vão gerar uma representação dita *localista* de conhecimentos.

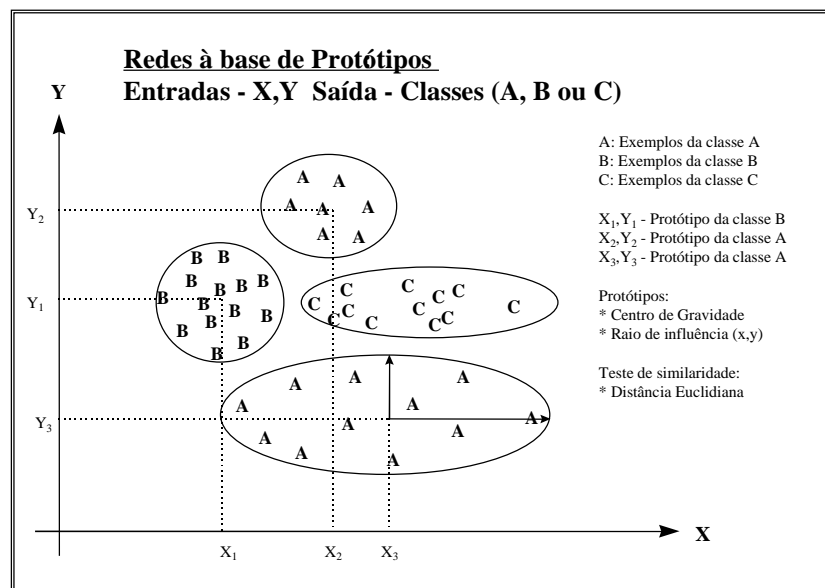


Figura 3.5. Protótipos de uma rede neural com duas entradas

- **Redes à base de Perceptrons:** as unidades do tipo “Perceptron” foram criadas por Frank Rosenblatt em 1950. Este é um dos modelos de neurônios mais utilizados na atualidade. Ele é a base de diversos tipos de RNA com aprendizado supervisionado utilizando uma adaptação por correção de erros (usualmente baseada na descida da superfície de erro usando o gradiente). O modelo do Perceptron de múltiplas camadas (MLP – *Multi-Layer Perceptron*) tornou-se muito conhecido e aplicado, sendo na maior parte das vezes associado a regra de aprendizado do Back-Propagation [Jodoin 94, Widrow 90, Rumelhart 86]. A figura 3.6 apresenta um esquema da representação de conhecimentos nas redes baseadas em Perceptrons, e como este tipo de redes é capaz de classificar padrões, gerando planos (ou hiper-planos) de divisão do espaço em que se situam os exemplos.

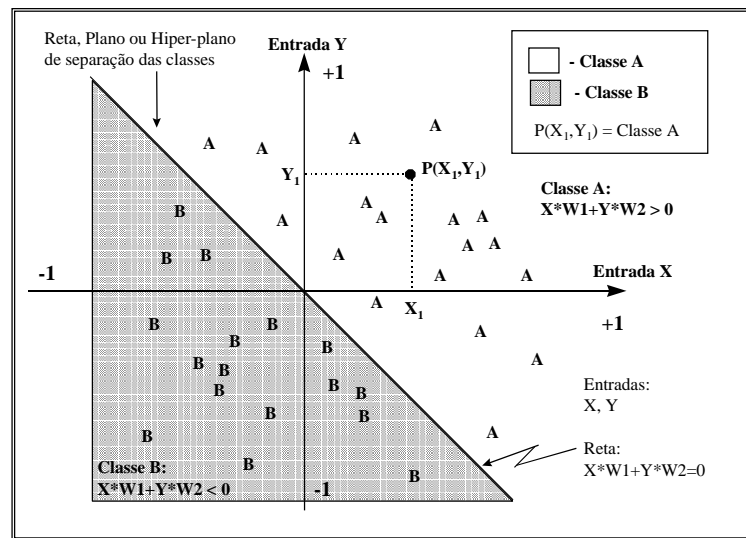


Figura 3.6. Separação de classes (classificação) através do uso de um Perceptron

3.3.2.3. Tipos de Arquiteturas de Conexão das Redes

As unidades de uma rede neural podem se conectar de diferentes modos, resultando em diferentes arquiteturas de interconexão de neurônios. A figura 3.7 apresenta alguns exemplos de possíveis maneiras de conectar os componentes de uma RNA. As arquiteturas de redes mais importantes são: **redes com uma única camada, redes com camadas uni-direcionais e redes recorrentes**.

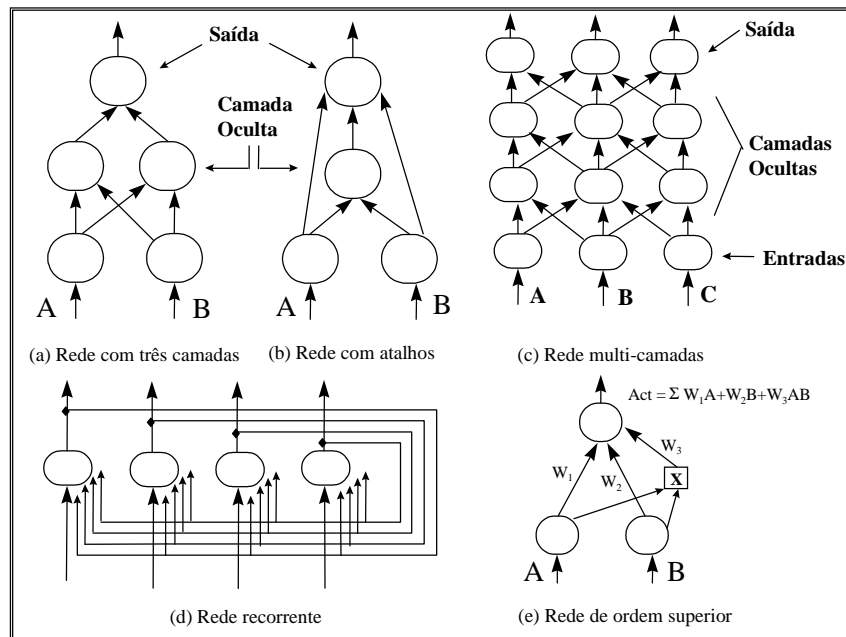


Figura 3.7. Arquiteturas de interconexão de neurônios em redes

A arquitetura de uma rede também pode ser classificada de acordo com a evolução desta no decorrer de sua utilização e desenvolvimento do aprendizado. Em função deste critério podemos ter os seguintes grupos: **redes com estrutura estática**, **redes com estrutura dinâmica**. As redes do tipo MLP com Back-Propagation são o exemplo típico de uma rede estática, e as redes do tipo Cascade-Correlation (CasCor [Fahlman 90]) são o exemplo típico das redes do tipo dinâmico e incremental. O último ponto relevante em relação a arquitetura das redes neurais está relacionado a possibilidade de se criar **redes modulares** [Ronco 96, Ronco 95, Amy 96, Rouzier 98, Jacobs 91].

3.3.2.4. Tipos de Aplicações das Redes Neurais

As RNA podem ser aplicadas à diferentes tipos de tarefas, tais como: o reconhecimento de padrões (e.g. reconhecimento de faces humanas), a classificação (e.g. reconhecimento de caracteres - OCR), a transformação de dados (e.g. compressão de informações), a predição (e.g. previsão de séries temporais, como as cotações da bolsa de valores, ou também, uso para o diagnóstico médico), o controle de processos e a aproximação de funções (e.g. aplicações na área da robótica). Um grande número de exemplos de aplicações pode ser encontrado no **UCI-ML repository** [UCI 99]. Todas estas tarefas podem ser reagrupadas em dois grupos principais, segundo o tipo de saída fornecido pela rede neural e o comportamento que é buscado. Estes dois grupos são:

- **Redes para a aproximação de funções:** este tipo de redes devem ter uma saída com valores contínuos e usualmente são empregadas para realizar aproximações de funções (interpolação) [Chentouf 97]. Neste tipo de aplicações, as funções são representada por um conjunto de pontos-exemplo desta. Este tipo de redes é capaz de aprender uma função de transformação (ou de associação) de valores de entrada em valores de saída, usualmente estimando por interpolação as respostas para os casos que não aparecem na base de exemplos. Este tipo de problemas de aprendizado neural de funções é conhecido por ser uma aplicação de um *problema de regressão* [Bishop 97]. Em geral as funções a serem aprendidas pelas redes possuem tanto as entradas como as saídas indicadas através de valores contínuos (variáveis não discretas).
- **Redes para a classificação de padrões:** este tipo de redes deve atribuir para cada exemplo que lhe é fornecido uma classe ao qual este exemplo pertence. Portanto, a saída da rede é a classe associada ao exemplo e por consequência, as classes são valores discretos e não contínuos. A classificação é um caso particular da aproximação de funções onde o valor de saída da rede é discretizado e pertence a um conjunto finito de classes. No caso do aprendizado supervisionado, o conjunto de classes é bem definido e conhecido antes de ser iniciado o processo de aprendizado. Uma rede utilizada para fins de classificação deve possuir saídas discretas, ou então, deve implementar métodos de discretização de suas saídas (e.g. aplicação de um limiar de discriminação – *activation threshold*). As entradas da rede podem ser tanto contínuas, como também podem ser discretas, o que não deve interferir no fato desta rede ser usada para uma aplicação classificação.

3.4. Considerações finais sobre as Redes Neurais

As redes conexionistas utilizam métodos de aprendizado à partir de exemplos que possibilitam o ajuste dos pesos de suas conexões, resultando em um comportamento próximo ou até mesmo exatamente igual ao esperado. Esta modificação dos pesos da rede é feita de forma que a rede generalize os conhecimentos contidos na base de exemplos de aprendizado.

Uma boa definição das redes conexionistas é dada por Giacommetti [Giacometti 92]: as redes são capazes de aprender e representar o “saber fazer algo” (*savoir-faire*), que se traduz pelo seu comportamento após o processo de aprendizado de uma tarefa; este “saber fazer”, representado pelos **conhecimentos práticos** (*empirical knowledge ~ practical examples*) adquiridos pela rede, aparece aqui em oposição ao “saber sobre algo” (*savoir-que*) que representa os **conhecimentos teóricos** sobre um determinado assunto (*theoretical knowledge ~ symbolic rules*). Esta distinção entre o saber fazer uma tarefa e o conhecimento sobre a tarefa, é um dos pontos mais importantes da discussão sobre os sistemas híbridos, pois ambos os conhecimentos se completam um ao outro [Osorio 98].

As redes conexionistas, em particular aquelas que tem sido aplicadas na construção de sistemas inteligentes, possuem as seguintes vantagens:

- **Conhecimento empírico:** o aprendizado à partir de exemplos é feito de uma maneira bastante simples e permite uma aquisição de conhecimentos de forma automática (ajuda a resolver em parte o problema do “gargalo da aquisição de conhecimentos” bem conhecido dos sistemas especialistas);
- **Degradação progressiva:** as respostas dadas por uma rede se degradam progressivamente na presença de “perturbações e distorções” dos dados de entradas. Em geral, as redes obtêm uma boa generalização dos conhecimentos presentes na base de aprendizado e sendo assim são menos sensíveis a “perturbações” do que os sistemas simbólicos;
- **Manipulação de dados quantitativos:** o fato de se trabalhar com uma representação numérica dos conhecimentos implica que as redes são melhor adaptadas para a manipulação de dados quantitativos (valores contínuos). Grande parte dos problemas de nosso mundo real, necessitam do tratamento de informações medidas de forma quantitativa, onde uma representação qualitativa muitas vezes implica na perda de informações. As redes neurais são menos vulneráveis aos dados aproximativos e a presença de dados distorcidos ou incorretos que possam estar presentes na base de aprendizado. Esta capacidade de manipular dados aproximados e até mesmo inexatos é mais difícil de ser encontrada em outros métodos de aprendizado do tipo simbólico;
- **Paralelismo em larga escala:** as redes neurais são compostas de um conjunto de unidades de processamento de informações que podem trabalhar em paralelo. Apesar da maioria das implementações de RNAs serem feitas através de simulações em máquinas sequenciais, é possível de se implementar (softwares e hardwares) que possam explorar esta possibilidade de ativação simultânea das unidades de uma rede. A maior parte das implementações de redes neurais simuladas em máquinas sequenciais pode ser facilmente adaptada em uma versão paralela deste sistema.

As redes conexionistas apresentam um certo número de inconvenientes, do mesmo modo que os outros tipos de métodos de aprendizado. No caso específico das redes, temos limitações tais como: dificuldade na definição da melhor arquitetura, parâmetros e forma de codificação dos dados; problemas de aprendizado ligados a uma inicialização ruim dos pesos; dificuldade em se compreender e visualizar os conhecimentos adquiridos, dificuldade de manipulação de conhecimentos simbólicos de modo similar aos sistemas de I.A. clássicos. Em relação a estas limitações, podemos encontrar na literatura vários trabalhos que se propõem a apresentar soluções para os problemas citados acima [Osorio 98, Orsier 95, Towell 91, Fahlman 88].

O algoritmo Back-Propagation ainda é um dos métodos mais usados junto as redes neurais, mas possui também muitas limitações. Alguns pesquisadores, conscientes dos problemas deste algoritmo, propuseram técnicas para resolver ou reduzir estes problemas [Schiffmann 93, 94]. Podemos citar aqui alguns destes métodos aperfeiçoados de aprendizado: o RPROP [Riedmiller 93], o QuickProp [Fahlman 88], o Gradiente Conjugado (Scaled Conjugated Gradient) [Moller 90] e o Cascade-Correlation [Fahlman 90], bem como as técnicas ontogênicas de aprendizado [Fiesler 94].

4. Processamento de Imagens baseado em Redes Neurais

O Processamento de Imagens é uma das áreas de estudos mais cativantes da Informática, que tem gerado um grande número de trabalhos junto a comunidade científica. As ferramentas de Processamento de Imagens tem permitido a realização de uma série de tarefas, tais como: a aquisição e transmissão de imagens digitais, o seu tratamento (filtros, compressão, processamento), e a extração e/ou identificação de seus componentes (reconhecimento de padrões). Estas tarefas vão na direção de um objetivo maior da área de Inteligência Artificial, que é o estudo e a reprodução da visão humana, ou seja, a área de visão artificial.

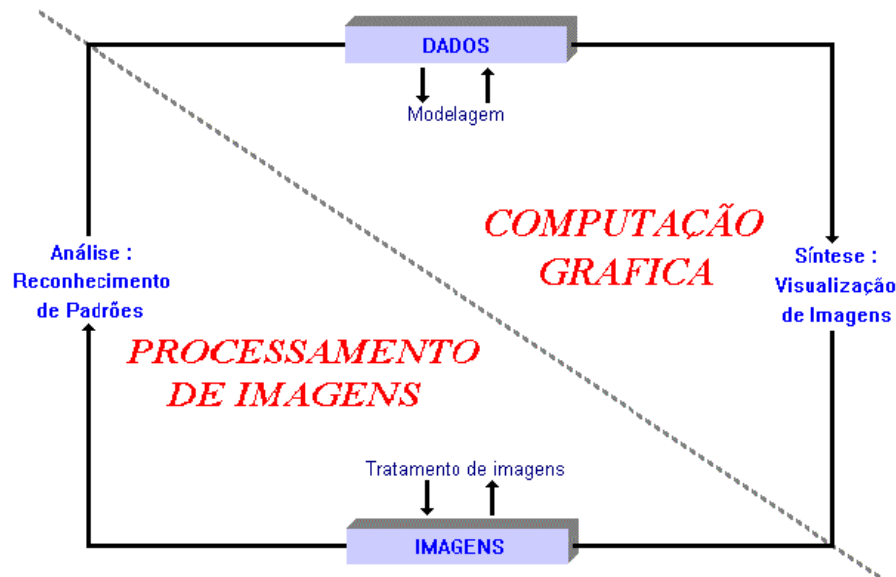


Figura 4.1. Processamento Gráfico: Computação Gráfica e Processamento de Imagens

A figura 4.1 mostra um esquema que permite nos situar melhor dentro das atividades ligadas ao Processamento Gráfico de dados e imagens digitais. Como podemos ver nesta figura, o Processamento de Imagens possui dois grandes eixos principais de atuação:

- **Tratamento de Imagens:** a partir de uma imagem original realizamos uma transformação desta imagem, gerando uma nova imagem tratada. O tratamento de imagem inclui funções como: ressaltar/restaurar elementos de uma imagem (e.g. ressaltar os contornos dos objetos), alterar seus componentes (e.g. transformar uma imagem colorida em preto e branco, aplicar um efeito especial), corrigir distorções (e.g. melhorar o foco de uma imagem pouco nítida), entre outras funções.
- **Reconhecimento de padrões:** a partir de uma imagem original realizamos um processamento que permite analisar esta imagem e identificar seus componentes. Assim podemos extrair uma descrição de alto nível dos seus componentes (e.g. modelo geométrico), agrupar os componentes que são similares, ou então classificar seus componentes em grupos predefinidos (classes de objetos).

O Processamento de Imagens convencional procura implementar, através algoritmos predefinidos, as funções matemáticas que nos permitem realizar as transformações e análises das imagens. O problema deste tipo de procedimento é que a função $F(x)$ que realiza esta transformação/classificação deve ser conhecida e bem especificada neste caso, conforme indicado na figura 4.2.

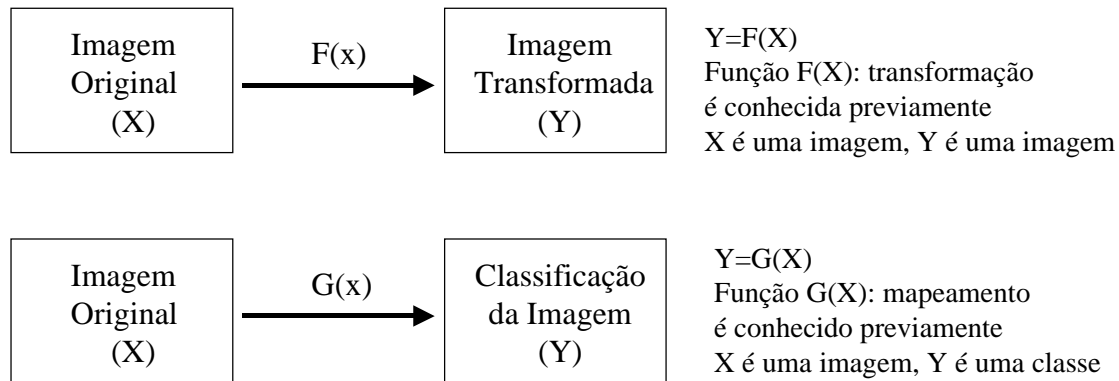


Figura 4.2. Processamento de imagens convencional

4.1. Tratamento de Imagens

O tratamento convencional de imagens é realizado por uma função $F(x)$ que irá realizar uma transformação de cada um dos pixels da imagem em novos pixels, onde estes sofrerão uma transformação segundo um algoritmo pré-especificado [Facon 93, Gonzales 93]. A principal desvantagem desta abordagem é que a realização de uma transformação sobre uma imagem deve ser feita de forma equacionada, baseando-se em algoritmos predefinidos que nem sempre são capazes de reproduzir o efeito visual desejado pelo usuário. Alguns aplicativos encontrados no mercado se utilizam de um recurso de alto nível como as macros, que caracterizam-se pela definição de uma sequência de comandos que permitem automatizar a reprodução de um determinado efeito, mas nem sempre o resultado final é completamente satisfatório. O nível de automatização também fica comprometido, pois em diversos momentos solicita-se a interação do operador para incluir e alterar novos parâmetros. Muitas vezes o processo de tratamento de uma imagem é realizado através de um processo de “tentativa e erro”, até que se encontre um efeito próximo ao desejado. Esta metodologia tradicional consiste basicamente em transformar uma imagem original em outra imagem, utilizando uma função conhecida, que consequentemente deve se apresentar disponível e implementada pela aplicação.

O que queremos obter é a capacidade de executar o processamento de imagens sem conhecer a função transformadora, onde a própria aplicação fosse capaz de gerar a imagem desejada pelo usuário (veja Figura 4.3). O usuário irá fornecer apenas exemplos de imagens originais e do resultado desejado do tratamento destas, deixando para o sistema a tarefa de “descobrir” uma maneira de realizar esta transformação (permitindo também que possamos repetir novamente esta transformação sobre outras imagens). Desta forma existe uma valorização dos atos criativos do autor, que pode criar um efeito baseado no seu gosto ou necessidade sem depender de uma equação predefinida. Podemos citar como exemplo, o caso de um cardiologista que gostaria de ressaltar, com cores específicas, detalhes existentes em uma foto ecocardiográfica. O médico pode realizar esta “operação de pintura” da imagem com naturalidade, o problema estaria em encontrar uma função que fosse capaz de colorir uma imagem, originalmente em tons de cinza (*gray scale*), da mesma forma que um médico faria.

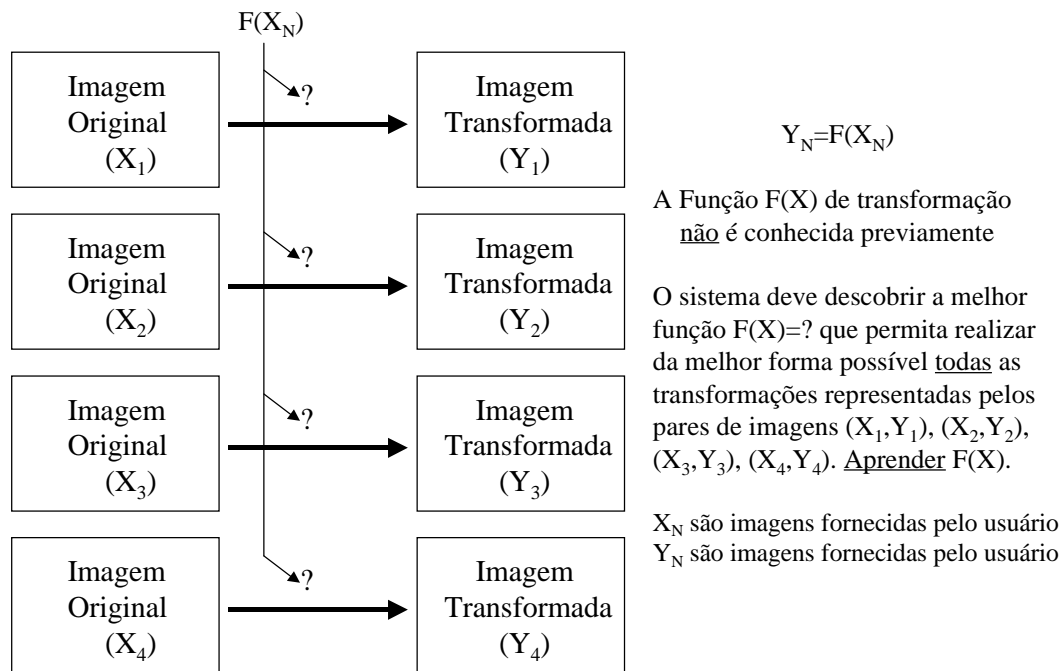


Figura 4.3. Processamento de Imagens usando o aprendizado

O uso das Redes Neurais Artificiais neste tipo de tratamento de imagens é bastante adequado, pois este tipo de ferramenta permite que se aprenda, a partir de um conjunto de exemplos, a realizar as transformações desejadas. As redes neurais também são bastante adequadas para a realização deste tipo de tarefas pois permitem que se trabalhe com: informações quantitativas (cores dos pixels – tanto como valores de entrada como de saída da rede), generalização dos conhecimentos (função obtida generaliza o filtro dados os casos particulares), robustez das redes em relação ao ruído/erros e paralelismo no processamento dos dados. Vamos apresentar a seguir exemplos de tratamento de imagens que podem ser realizados com o uso de métodos tradicionais, onde estamos atualmente desenvolvendo pesquisas com o objetivo de criar procedimentos que permitam o uso de redes neurais substituindo estes métodos convencionais.

4.1.1. Transformação de Cores:

As transformações de cores que estudamos incluem: a transformação de imagens em “TrueColor” (RGB com 24 bits/pixel = 8 bits Red, 8 bits Green, 8 bits Blue) em imagens de uma única cor do tipo “GrayScale” (escala com tons de cinza); a transformação de cores através da atribuição de cores falsas a uma imagem monocromática do tipo GrayScale; a transformação de uma imagem colorida 256 cores (uso de uma “palette”) ou GrayScale em uma imagem preto e branco (P&B = 1 bit/pixel) através de processos como *Halftoning* ou *Dithering* (realizam uma redução do número de cores e consequentemente do número de bits/pixel). A tabela abaixo resume estas transformações:

Imagem	Tipo - Bits/Pixel	Tipo - Bits/Pixel	Tipo - Bits/Pixel	Tipo - Bits/Pixel	Tipo - Bits/Pixel
Origem	RGB 24	RGB 24	Gray 8	Gray/Palette 8	Gray/Palette 8
Destino	Gray 8	Palette 8	Palette 8	P&B 1	RGB 24
Transformação	Tirar Cor	Reduzir Cor	Falsas Cores	Reduzir Cor	Falsas Cores*

* Este tipo específico de transformação tem apresentado problemas no aprendizado/generalização

Abaixo seguem algumas figuras ilustrando as transformações descritas anteriormente.

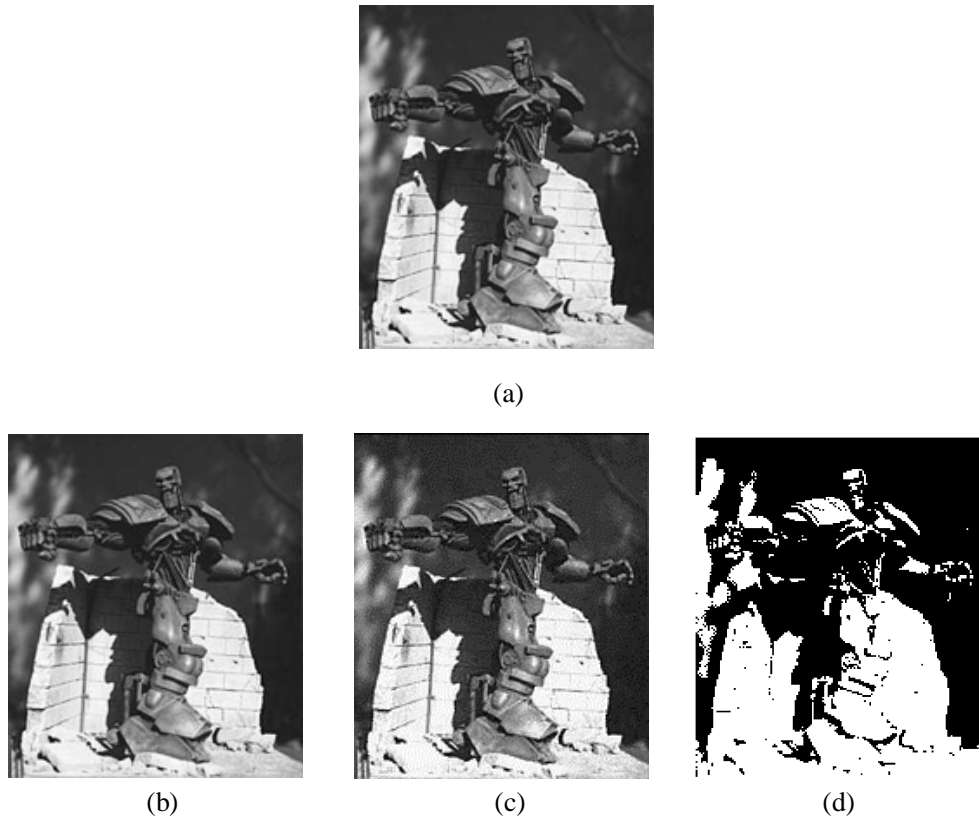


Figura 4.4. Experimento RGB 24 para GrayScale 8.

- (a) Imagem original; (b) Imagem transformada usando a equação matemática $F(\text{pixel})=(R+G+B)/3$;
(c) Imagem gerada por aprendizado sem o uso do contexto; (d) Imagem gerada por aprendizado usando contexto

	X	
X	P	X
	X	

4-vizinhança

X	X	X
X	P	X
X	X	X

8-vizinhança

Figura 4.5. Conceito de vizinhança de um pixel (contexto)

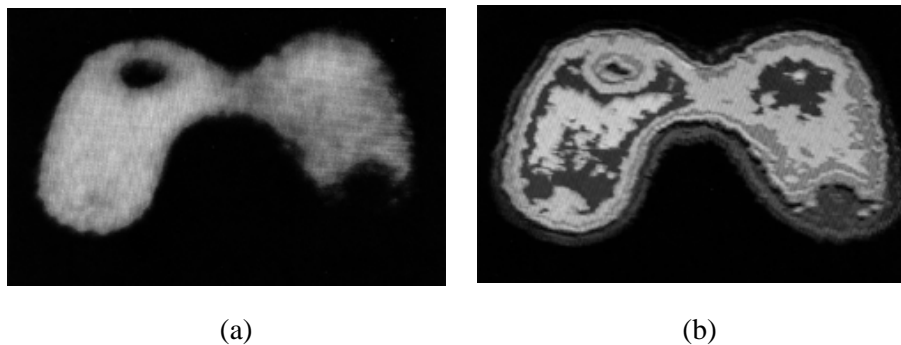


Figura 4.6. Cores falsas usando os métodos convencionais

4.1.2. Filtros – Realce, Restauração, Deformação de Imagens:

Uma transformação de imagens muito comum é aquela que pode ser obtida através da aplicação de filtros e de matrizes de convolução, que permitem: realçar elementos da imagem (detectar bordas, contornos, linhas horizontais, linhas verticais, etc), restaurar ou eliminar pontos da imagem, melhorar as imagens acentuando (*sharpening* - *focar*) ou suavizando (*smoothing* - *desfocar*) diferenças, deformar a imagem, entre outras operações. De um modo geral podemos alterar os pixels de modo individual ou considerando-se um grupo destes pixels que formam uma matriz de contexto/vizinhança similar a apresentada anteriormente. As imagens abaixo demonstram a aplicação de transformações como as citadas aqui:

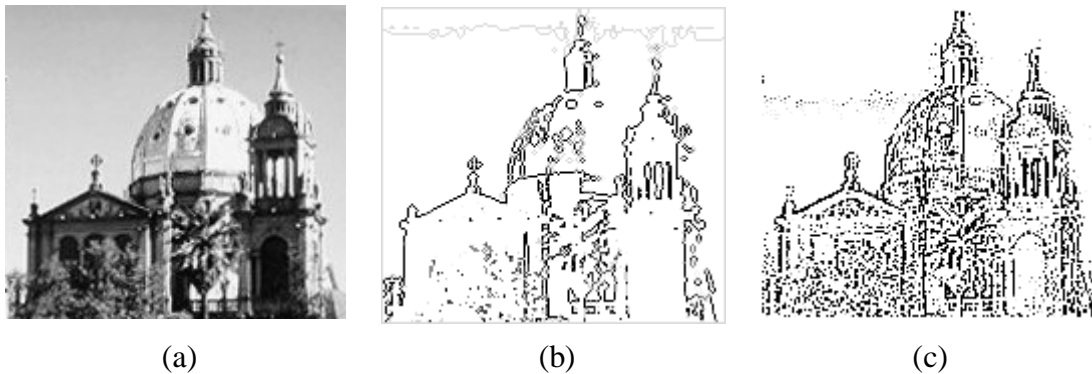


Figura 4.7. Detecção de contornos em uma imagem: RGB 24 para P&B
(a) Imagem original (b) Imagem gerada por um filtro (c) Imagem gerada pela RNA

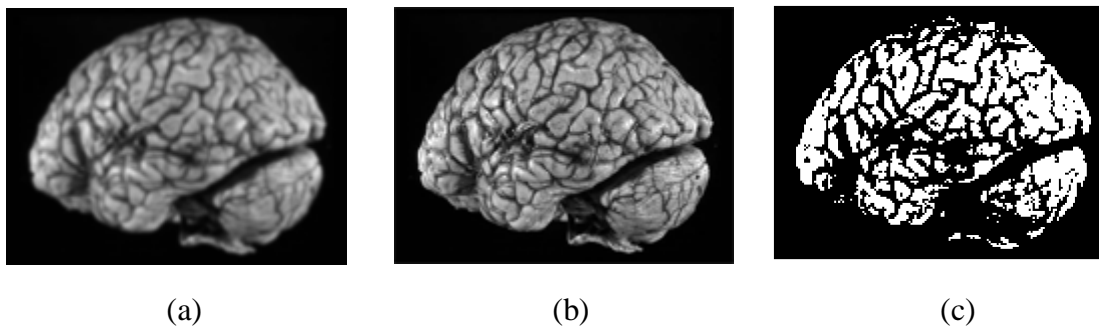


Figura 4.8. Focalização de imagens distorcidas (*sharpening*)
(a) Imagem distorcida (b) Imagem de referência usada no aprendizado (c) Imagem obtida após o aprendizado

4.1.3. Ferramenta NeuronColor:

O **NeuronColor** (vide Fig. 4.9) é uma aplicação que foi desenvolvida junto ao grupo de pesquisa em I.A. da Unisinos, tendo sido desenvolvida especialmente para viabilizar experimentos de processamento de imagens usando Redes Neurais Artificiais do tipo Back-Propagation. Seus principais objetivos são a transformação de uma imagem qualquer através do uso de uma rede neural treinada e a criação de conjuntos de exemplos com entradas e saídas formatadas para serem usados no processo de aprendizagem. O NeuronColor possui também a possibilidade de executar uma avaliação visual e/ou quantitativa da qualidade de uma imagem gerada pela rede em relação a uma outra imagem que representa o objetivo.

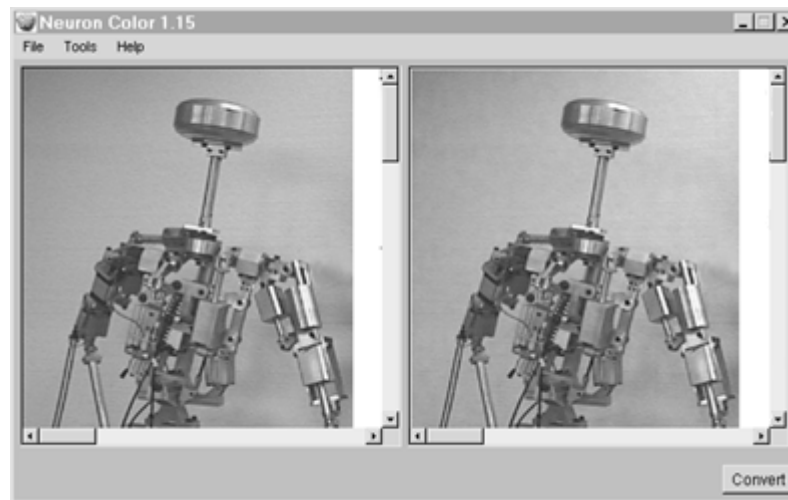


Figura 4.9. Interface da janela principal do NeuronColor

Para o desenvolvimento desta ferramenta foi utilizada a linguagem Java (versão 1.2.2) pelos seguintes motivos: (a) a linguagem já possui classes especializadas na manipulação de imagens - transformação de uma imagem em um vetor de inteiros, extração de informações de vermelho, verde e azul de cada pixel da imagem; (b) a reusabilidade das classes facilitando a manutenção, principalmente nesta fase de prototipação; (c) o fato de ser multiplataforma (portabilidade).

Como podemos perceber nos exemplos destas imagens obtidas com o uso de uma rede neural, o aprendizado de tarefas de tratamento de imagens tem dado resultados muito promissores. Atualmente estudos tem sido desenvolvidos no sentido de melhorar a performance do algoritmo de aprendizado, bem como estamos realizando novas experiências visando caracterizar melhor o desempenho da rede e as possíveis aplicações práticas desta técnica.

4.2. Reconhecimento de Padrões

As tarefas de reconhecimento de padrões podem ser facilmente implementadas através do uso de um algoritmo de aprendizado supervisionado, onde necessitamos apenas de um conjunto de exemplos de imagens previamente “etiquetadas” com as respectivas classes, a serem usadas no treinamento da rede. O exemplo clássico deste tipo de aplicação é o reconhecimento de caracteres em imagens de textos que foram digitalizados (OCR). O reconhecimento de caracteres é uma tarefa difícil, dadas as inúmeras variações em relação a posição, orientação, tamanho e mesmo forma e erros de leitura dos caracteres, que dificultam a aplicação de um algoritmo predeterminado para o reconhecimento destes. Um algoritmo de aprendizado permite ao sistema de OCR que ele seja “treinado” para reconhecer novos fontes de textos. Os sistemas OCR são bastante complexos, consistindo de múltiplos módulos, onde podemos usar uma rede neural como mecanismo de reconhecimento de padrões [Osorio 91].

No Apêndice B deste trabalho encontramos exemplos de imagens de caracteres extraídos de textos digitalizados. Estes caracteres podem ser identificados com o uso de um sistema OCR, onde a partir da matriz de pontos devemos ser capazes de retornar a classe ao qual esta matriz pertence (e.g. código ASCII do caracter). Uma rede neural pode receber como entrada os dados contidos na matriz, e como saída retornar o código correspondente ao caracter fornecido.

5. Conclusão e Perspectivas

Neste trabalho apresentamos uma visão geral sobre os sistemas de I.A. e a necessidade do aprendizado para que um sistema inteligente possa ser considerado como tal. Enfocamos o aprendizado neural como sendo uma forma robusta de aquisição de conhecimentos, que dadas as suas peculiaridades, possui um interesse particular na área de Inteligência Artificial e um grande potencial de aplicação no Processamento de Imagens.

Procuramos apresentar neste trabalho uma visão bastante ampla do assunto, levantando questionamentos e pontos em aberto para estudos futuros, de forma que o leitor possa ter ao mesmo tempo uma visão global da área, e também uma noção dos temas de pesquisa na atualidade neste domínio. Este trabalho visa ser uma fonte de questionamentos e idéias para novos trabalhos, onde a extensa relação de bibliografias remetem o leitor as demais obras da área que podem complementar os temas aqui abordados.

Concluindo, acreditamos que as pesquisas futuras nos levam em direção aos sistemas com formas robustas de aquisição e representação de conhecimentos. As pesquisas sobre os sistemas híbridos, sistemas multi-agentes e sistemas com múltiplas inteligências parecem ser uma tendência atual nesta área [Osório 99]. Devemos buscar a integração dos métodos simbólicos com os métodos conexionistas de forma a expandir as potencialidades dos “sistemas inteligentes”, para quem sabe assim podermos realmente ter sistemas de processamento de imagens capazes de realizar tarefas mais complexas, como por exemplo, as tarefas ligadas a implementação de sistemas de Visão Artificial.

APÊNDICES

A.1. Exemplo de Dados para o Aprendizado (Exemplos)

A	B	XOR
0	0	0
0	1	1
1	0	1
1	1	0

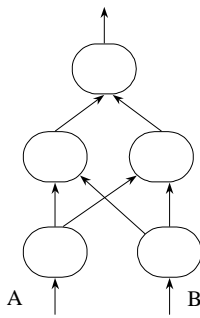
⇒ Conhecimentos Empíricos

Entradas : A e B

Saída : XOR (saída usada no aprendizado supervisionado)

Número de exemplos : 4

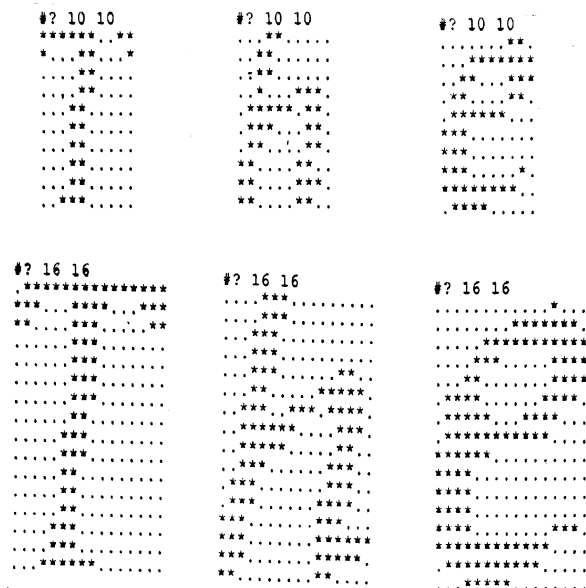
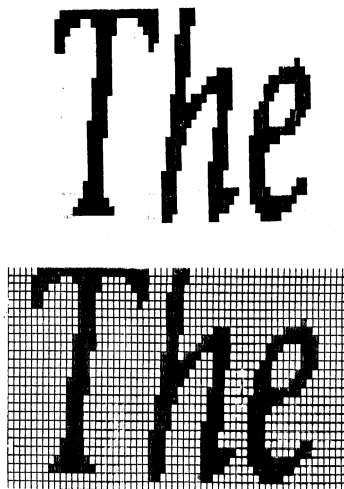
A.3. Rede Neural – Solução para o problema do XOR



Topologia da rede neural usada no aprendizado do XOR

A.4. Texto digitalizado e Caracteres – Aplicação de um OCR

*The Quick Brown Fox Jumps Over The Lazy Dog.
Its Crackers to Slip a Rozzer the Dropsy in Shine*



BIBLIOGRAFIA

[Amy 96] AMY, Bernard. *Recherches et Perspectives dans le Domaine des Réseaux Connexionnistes*. Rapport de Recherche - DRET, Octobre 1996. Web: <http://www-leibniz.imag.fr/RESEAUX/public.html>
Ftp: <ftp://ftp.imag.fr/pub/LEIBNIZ/RESEAUX-D-AUTOMATES/>

[Anderson 93] ANDERSON, J. R. *Rules of the Mind*. Erlbaum Ed., Hillsdale, NJ. 1993.

[Arbib 95] ARBIB, Michael A. (Editor) *The Handbook of Brain Theory and Neural Networks*. MIT Press, 1995.

[Bishop 97] BISHOP, Christopher. *Classification and Regression*. In: Handbook of Neural Computation (section B6.2). E. Fiesler and R. Beale (Eds.) Institute of Physics and Oxford University Press. New York, NY - U.S.A., 1997. Web: <http://www.idiap.ch/publications/fiesler-96.1.bib.abs.html> or http://www.oup-usa.org/acadref/nc_accs.html

[Bishop 95] BISHOP, C.M. *Neural Networks for Pattern Recognition*, Oxford: Oxford University Press. ISBN 0-19-853849-9. 482 pages.1995.

[Carpenter 83] CARPENTER, G. & GROSSBERG, S. *A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine*. Computer Vision, Graphics and Image Processing, v.37, p.54-115, 1983.

[Caudill 92] CAUDILL, Maureen & BUTLER, Charles. *Understanding Neural Networks - Vol.1: Basic Networks, Vol.2: Advanced Networks*. Bradford Books, MIT Press, 1992.

[Chentouf 97] CHENTOUF, Rachida. *Construction de Réseaux de Neurones Multicouches pour l'Approximation*. Thèse de Doctorat en Sciences Cognitives, Laboratoire TIRF - INPG, Grenoble - France, Mars 1997.

[Elman 93] ELMAN, Jeffrey L. *Learning and Development in Neural Networks: The Importance of Starting Small*. Cognition, 48(1993), pp.71-99. 1993. Web: <http://crl.ucsd.edu/~elman/>
Ftp: <ftp://crl.ucsd.edu/pub/neuralnets/cognition.ps.Z>

[Facon 93] FACON, Jacques. *Processamento e Análise de Imagens*. Córdoba. Universidade Nacional de Córdoba, 1993. 198p.

[Fahlman 88] FAHLMAN, Scott E. *An Empirical Study of Learning Speed in Back-Propagation Networks*. Carnegie Mellon University - CMU. Computer Science Technical Report CMU-CS-88-162. September 1988. Web: <http://www.cs.cmu.edu/Reports/index.html>

[Fahlman 90] FAHLMAN, S. E.; LEBIERE, C. *The Cascade-Correlation Learning Architecture*. Carnegie Mellon University - CMU, Computer Science Technical Report - CMU-CS-90-100. February 1990. Web: <http://www.cs.cmu.edu/Reports/index.html>
Ftp: <ftp://archive.cis.ohio-state.edu/pub/neuroprose/fahlman.cascor-tr.ps.Z>

[Faq 99] FAQ. *Faq ANN - Comp.ai.neural-nets* 1999.
Web: <http://www.cis.ohio-state.edu/hypertext/faq/usenet/ai-faq/neural-nets/>

[Faq 99a] FAQ. *Faq ANN - Comp.ai.neural-nets FAQ*, Part 4 of 7: Books, data, etc. 1999.
Web: <http://www.cis.ohio-state.edu/hypertext/faq/usenet/ai-faq/neural-nets/part4/faq-doc-1.html>

[Fiesler 94a] FIESLER, E. *Neural Networks Formalization and Classification*. Computer Standard & Interfaces, Special Issue on Neural Networks Standards, John Fulcher (Ed.). V.16, N.3. Elsevier Sciences Publishers, Amsterdam, June, 1994. Web: <http://www.idiap.ch/idiap-networks.html> .
Ftp: <ftp://ftp.idiap.ch/pub/papers/neural/fiesler.formalization.ps.Z>

[Fiesler 94b] FIESLER, Emile. *Comparative Bibliography of Ontogenic Neural Networks*. Proceedings of the International Conference on Artificial Neural Nets - ICANN'94. Sorrento, Italy, May 1994. Web: <http://www.idiap.ch/idiap-networks.html> .
Ftp: <ftp://ftp.idiap.ch/pub/papers/neural/fiesler.ontogenic-summary.ps.Z>

[Fiesler 97] FIESLER, E. & BEALE, R. *Handbook of Neural Computation*. Institute of Physics and Oxford University Press. New York, NY - U.S.A., 1997. Web: <http://www.idiap.ch/publications/fiesler-96.1.bib.abs.html> or http://www.oup-usa.org/acadref/nc_accs.html

[Freeman 92] FREEMAN, James A., SKAPURA, David M.. *Neural Networks: Algorithms, Applications, and Programming Techniques*. 1. ed., Reading: Addison-wesley, 1992. 401 p. il.

[Giacometti 95] GIACOMETTI, Arnaud. *ARN2 - A Prototype-Based Incremental Neural Network*. WNDA - Workshop on Neural Network Design and Analysis. University of Geneva. January 1995.

[Goldberg 89] GOLDBERG, D. E. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Publishing. 1989.

[Gonzales 93] GONZALES, Rafael C.; WOODS, Richard E. *Digital Image Processing*. Addison-Wesley Publishing, Reading. 1993. 716p.

[Hebb 49] HEBB, D. O. *The Organization of Behaviour*. John Wiley & Sons. New York, 1949.

[Hinton 84] HINTON, G.; SEJNOWSKI, T; ACKLEY D. *Boltzmann Machines: Constraint Satisfaction Networks that Learn*. Carnegie-Mellon University – Tech. Report CMU-CS-84-119. 1984.

[Hopfield 82] HOPFIELD, J. J. *Neural Networks and Physical Systems with Emergent Computational Abilities*. In: Proceedings of the National Academy of Sciences, v.79, Washington, USA. p.2554-2558, April 1982.

[Jacobs 91] JACOBS, R. A., JORDAN, M. I., NOWLAN, S., and HINTON, G. E. (1991). *Adaptive mixtures of local experts*. In *Neural Computation*, (3), 1-12.

Web: <http://www.ai.mit.edu/projects/cbcl/people/jordan/jordan-hp.html>

[Jodouin 94a] JODOUIN, Jean-François. *Les Réseaux de neurones : Principes et définitions*. Editions Hermès, Paris, 1994.

[Jodouin 94b] JODOUIN, Jean-François. *Les Réseaux Neuromimétiques : Modèles et applications*. Editions Hermès, Paris, 1994.

[Kohonen 82] KOHONEN, Teuvo. *Self-Organized formation of Topologically Correct Feature Maps*. *Biological Cybernetics*, v.43, p.32-48. Jan. 1982.

[Kohonen 87] KOHONEN, Teuvo. *Self-Organization and Associative Memory*. Springer-Verlag Series in Information Science. 1987.

[Kolodner 93] KOLODNER, J. *Case-Based Reasoning*. Morgan Kaufmann Series in Representation and Reasoning, San Mateo, CA. 1993.

[Kosko 87] KOSKO, Bart. *Adaptive Bidirectional Associative Memories*. *Applied Optics*, v.26, p.4947-4960. Dec. 1987.

[Kosko 87a] Kosko, Bart. *Constructing an Associative Memory*. *Byte*, Highstown, v. 12, n.10, p.137-144. Sept. 1987.

[Kovacs 97] KOVACS, Zsolt Laslo. *Redes Neurais Artificiais: Fundamentos e Aplicações*. 2 ed. São Paulo: Collegium Cognitio, 1997. 174p. il.

[Krogh 95] KROGH, Anders & VEDELSBY, Jesper. *Neural Network Ensembles, Cross Validation and Active Learning* in *NIPS - Advances in Neural Information Processing Systems*, Vol. 7, pp. 231-238, The MIT Press, 1995), also available in the neuroprose archive as [krogh.ensemble.ps.Z](http://neuroprose.mit.edu/krogh.ensemble.ps.Z).

[Krose 93] KRÖSE, Ben J. & VAN DER SMAGT, Patrick. *An Introduction to Neural Networks*. University of Amsterdam, 1993. Web: <http://www.fwi.uva.nl/research/ias/> ou <ftp://ftp.wins.uva.nl/pub/computer-systems/aut-sys/reports/neuro-intro/>

[Larousse 99] LAROUSSE. Grande Enciclopédia Larousse Cultural. Editora Nova Cultural, 1999.

[McCulloch 43] McCULLOCH, W. S. & PITTS, W. *A Logical Calculus of the Ideas Imminent in Nervous Activity*. Bulletin of Mathematical Biophysics, 5, p.115-133, 1943.

[Minsky 85] MINSKY, Marvin. *The Society of Mind*. Simon & Schuster, New York, 1985. (possui uma edição traduzida para o português).

[Minsky 69] MINSKY, M. & PAPERT, S. *Perceptrons: An Introduction to Computational Geometry*. MIT Press, Cambridge. 1969.

[Minsky 90] MINSKY, Marvin. *Logical vs. Analogical or Symbolic vs. Connectionist or Neat vs. Scruffy*. In: Artificial Intelligence at MIT _ Expanding Frontiers, P. Winston (ED.), vol 1. MIT Press (reprinted in AI Magazine). 1990. Web: <http://www.ai.mit.edu/people/minsky/minsky.html>

[Mitchell 97] MITCHELL, Tom M. *Machine Learning*. McGraw-Hill, 1997.
Web: <http://www.cs.cmu.edu/afs/cs.cmu.edu/user/mitchell/ftp/tomhome.html>

[Moller 90] MOLLER, Martin F. *A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning*. Technical Report PB-339 - Computer Science Dept., University of Aarhus, Denmark. November 1990.
Ftp: <ftp://archive.cis.ohio-state.edu/pub/neuroprose/moller.conjugate-gradient.ps.Z>

[Nilsson 98] NILSSON, Nils J. *Artificial Intelligence: A New Synthesis*. Morgan Kaufmann Publishers. 1998. Web: <http://robotics.stanford.edu/people/nilsson/mlbook.html>

[Nikolopoulos 97] NIKOLOPOULOS, Chris. *Expert Systems – Introduction to First and Second Generation and Hybrid Knowledge Based Systems*. Marcel Dekker Inc. Press. 1997.

[Orsier 94] ORSIER, Bruno; AMY, Bernard; RIALLE, Vincent & GIACOMETTI, Arnaud. *A study of the hybrid system SYNHESYS*. Workshop ECAI-94 (European Conference on Artificial Intelligence) - Combining Connectionist and Symbolic Processing. Amsterdam, Agosto 1994.

[Orsier 95] ORSIER, Bruno. *Etude et Application de Systèmes Hybrides NeuroSymboliques*. Thèse en Informatique, Laboratoire LIFIA-IMAG, UJF - Grenoble, 1995.
Web: <http://www-leibniz.imag.fr/RESEAUX/public.html> .
Ftp: <ftp://ftp.imag.fr/pub/LEIBNIZ/RESEAUX-D-AUTOMATES/orsier.these.ps.gz>

[Osório 91] OSORIO, Fernando Santos. *Um Estudo sobre Reconhecimento Visual de Caracteres através de Redes Neurais*. Dissertação de Mestrado, CPGCC, UFRGS, Porto Alegre - Brasil. Outubro 1991.

[Osório 92] OSORIO, F. S. *Simulação de Redes Neurais Artificiais de Neurônios com Aprendizado Supervisionado*. Revista Scientia, Unisinos. v.3, n.1, p.45-66. 1992.

[Osório 98] OSORIO, Fernando Santos. *INSS: Un Système Hybride Neuro-Symbolique pour l'Apprentissage Automatique Constructif*. Thèse de Doctorat (Ph.D.) en Informatique. Laboratoire Leibniz – IMAG / INPG. Grenoble, France. 1998.
Web: <http://www-leibniz.imag.fr/RESEAUX/osorio/These/>

[Osório 99] OSORIO, F. S. & VIEIRA, R. Sistemas Híbridos Inteligentes. In: ENIA'99 – Encontro Nacional de Inteligência Artificial (Tutorial). Congresso da SBC'99. Rio de Janeiro, 1999.
Web: <http://www.inf.unisinos.tcche.br/~osorio/enia99/>

[Osório 99a] OSORIO, Fernando S. Redes Neurais Artificiais: Do Aprendizado Natural ao Aprendizado Artificial. I Forum de Inteligência Artificial, ULBRA – Canoas, agosto 1999.
Web: <http://www.inf.unisinos.br/~osorio/IForumIA/fia99.pdf>

[Parker 82] PARKER, D. *Learning-Logic*. Invention Report S81-64, File 1, Office of Technology Licensing, Stanford University, Stanford. Oct. 1982.

[Pinker 99] PINKER, Steve. *Como a Mente Funciona*. Ed. Companhia das Letras, São Paulo, 1999.

[Quinlan 93] QUINLAN, J. R. C4.5- Programs for Machine Learning. Morgan Kauffman Publishers. San Mateo, CA. 1993. Web: <http://www.cs.su.oz.au/People/quinlan.html>

[Riedmiller 93] RIEDMILLER, Martin & BRAUN, Heinrich. *A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm*. Proceedings of the IEEE International Conference on Neural Networks. San Francisco - CA - USA. 1993. Web: <http://i11www.ira.uka.de/~riedml/> (or ~neuro)

[Ripley 96] RIPLEY, B.D. *Pattern Recognition and Neural Networks*, Cambridge: Cambridge University Press, ISBN 0-521-46086-7. 403 pages. 1996.

[Ronco 95] RONCO, Eric & GAWTHROP, Peter J., 1995. *Modular Neural Networks: a state of the art*. Tech. rep CSC-95026. University of Glasgow.

Web: <http://www.ee.usyd.edu.au/~ericr/pub/techrep.html>

[Ronco 96] RONCO, Eric; GOELLE, Henrik & GAWTHROP, Peter J., 1996. *Modular Neural Network and Self-Decomposition*. Tech. Rep CSC-96012. University of Glasgow.

Web: <http://www.ee.usyd.edu.au/~ericr/pub/techrep.html>

[Rouzier 98] ROUZIER, Sophie. *Réseaux Modulaires*. Thèse de Doctorat en Sciences Cognitives, Eq. Réseaux d'Automates - Lab. LEIBNIZ - IMAG, Grenoble - France, 1998.

[Rosenblatt 59] ROSENBLATT, R. Principles of Neurodynamics. Spartan Books. New York. 1959.

[Rumelhart 85] RUMELHART, D. ; HINTON, G.; WILLIAMS, R. *Learning Internal Representations by Error Propagation*. ICS Report 8506, Institute for Cognitive Science, University of California at San Diego, La Jolla. Sept. 1985.

[Rumelhart 86] RUMELHART, D.; HINTON, G. & WILLIAMS, R. *Learning Internal Representations by Error Propagation*. In : Rumelhart & McClelland: Parallel Distributed Processing - Explorations in the Microstructure of Cognition - Vol.1: Foundations. Cambridge: MIT Press, 1986.

[Schiffmann 93] SCHIFFMANN, W.; JOOST, M. & WERNER, R. *Comparison of Optimized Backpropagation Algorithms*. Proceedings of the European Symposium on Artificial Neural Networks, ESANN'93, Brussels, p.97-104, 1993. Web: <http://www.uni-koblenz.de/~schiff/publications.html>

[Schiffmann 94] SCHIFFMANN, W.; JOOST, M. & WERNER, R. *Optimization of the Backpropagation Algorithm for Training Multilayer Perceptrons*. Technical Report, University of Koblenz, Deutschland. September 1995. Web: <http://www.uni-koblenz.de/~schiff/publications.html>

[Simpson 90] SIMPSON, Patrick K. *Artificial Neural Systems: Foundations, Paradigms, Applications and Implementations*. Pergamon Press, 1990.

[Sutton 98] SUTTON, Richard S. & BARTO, Andrew G. *Reinforcement Learning: An Introduction*. MIT Press (A Bradford Book), Cambridge, MA, 1998.

[Towell 91] TOWELL, Geoffrey. *Symbolic Knowledge and Neural Networks: Insertion, Refinement and Extraction*. Ph.D. Thesis, Computer Science Dept., University of Wisconsin-Madison, U.S.A. 1991.

Web: <http://www.cs.wisc.edu/~shavlik/uwml.html>

Ftp: <ftp://ftp.cs.wisc.edu/machine-learning/shavlik-group/> (towell.thesis.*.ps)

[UCI 99] UCI-ML. *Machine Learning Repository*. UCI – University of California Irvine. Dept. of Computer Science. Web: <http://www.ics.uci.edu/~mllearn/MLRepository.html>

[Widrow 62] WIDROW, Bernard. *Generalization and Information Storage in Networks of ADALINE Neurons*. In: Self-Organization Systems. Spartan Books. Washington: p.435-461, 1962.

[Widrow 88] WIDROW, Bernard & WINTER, R. *Neural Nets for Adaptive Filtering and Adaptive Pattern Recognition*. IEEE Computer, New York, v.21, n.3, p.25-39, march 1988.

[Widrow 90] WIDROW, Bernard & LEHR, M. *30 Years of Adaptive Neural Networks : Perceptron, Madaline, and Back-Propagation*. Proceedings of the IEEE, New York, Vol.78, N.9, pp.1415-1441. September 1990.