Redes Neuronais na Previsão de Séries Temporais

Carlos Mauricio Raposo

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Aprovada por:

Prof. Luis Alfredo Vidal de Carvalho, D. Sc.
(presidente)

Profa. Doris Ferraz de Aragon, D. Sc.

Prof. Basilio de Bragança Pereira, Ph.D.

RIO DE JANEIRO, RJ – BRASIL
ABRIL DE 1992

Carlos Mauricio Raposo

Redes Neuronais na Previsão de Séries Temporais [Rio de Janeiro] 1992
X, 74 p., 29.7 cm, (COPPE/UFRJ, M. Sc., ENGENHARIA DE SISTEMAS E COMPUTAÇÃO, 1992)
TESE – Universidade Federal do Rio de Janeiro, COPPE
1 – Redes Neuronais 2 – "Backpropagation" 3 – Séries Temporais 4 – Previsão
I. COPPE/UFRJ II. Título(Série).

Resumo da Tese apresentada à COPPE como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M. Sc.)

Redes Neuronais na Previsão de Séries Temporais

Carlos Mauricio Raposo

Abril de 1992

Orientadores: Luis Alfredo Vidal de Carvalho
Basilio de Bragança Pereira
Programa: Engenharia de Sistemas e Computação

Apresentaremos nesta tese o modelo de redes neuronais "Backpropagation" como ferramenta para previsões em séries temporais. Propomos também um procedimento para fornecer a topologia mais adequada da rede neuronal na modelagem da série em questão.

Como parâmetro de avalição foi utilizado uma das metodologias mais tradicionais de modelagem e análise de séries temporais, denominado por *Metodolgia Box-Jenkins*. Nossos testes demonstraram, que mesmo com uma rede neuronal simples, o modelo "Backpropagation" foi superior a essa metodologia.

Abstract of Thesis presented to COPPE as partial fulfillment of the requirements for the degree of Master of Science (M. Sc.)

Neural Networks in time series forecasting

Carlos Mauricio Raposo

April, 1992

Thesis Supervisors: Luis Alfredo Vidal de Carvalho
Basilio de Bragança Pereira
Department: Programa de Engenharia de Sistemas e Computação

It's present in this work, a neural networks model denoted "Backpropagation", used as a forecasting tool in times series. It is also proposed a procedure to determine an adequate topology for the neural network which models tese series.

In order to evaluate the performance of "Backpropagation" model it was used the Box-Jenkins methodology, a classical in time series analysis and modelling. The tests which were developed demonstrated that "Backpropagation" model, even the simple one, had a better performance than Box-Jenkins.

Índice

Lista de Figuras

Lista de Tabelas

Capítulo I

Introdução

Em algumas abordagens, a Inteligência Artificial é considerada como um ramo da ciência da computação que trata da construção de sistemas inteligentes, supostamente capazes de executar tarefas específicas, que o ser humano executa e os sistemas tradicionais não o conseguem de forma eficiente.

Nesse ramo da ciência da computação podemos destacar duas concepções diferentes quanto ao tratamento das funções cognitivas: *simbolista* e *conexionista.*

Os sistemas simbolistas são compostos de símbolos, que se combinam formando as expressões e estruturas que operam sobre expressões. Essas estruturas são capazes de modificar, destruir, reproduzir ou criar novas expressões. A concepção simbolista vê a solução de problemas como um processo essencialmente algoritmico, e tem sido largamente utilizado, e com sucesso, em certas áreas, tais como: robótica, sistemas especialistas, provadores de teoremas, etc. Porém,encontrando grandes limitações, pelo seu caráter sequencial e centralizador, em tarefas como percepção, controle motor, associação e outras atividades, por serem difíceis de algoritmizar de forma eficiente.

Os conexionistas se inspiram no sistema neuronial biológico. Para eles as informações inteligentes devem ser processadas tal como no cérebro: de forma paralela por um conjunto de elementos computacionais simples (ex.: redes neuronais). Só assim seria possível executar as tarefas acima mencionadas, que tem

sido muito difíceis de serem tratadas através dos algoritmos tradicionais.

Redes Neuronais tem sido vastamente estudadas e aplicadas, através de varios modelos, em diferentes áreas. Um dos modelos mais utilizados é o modelo "Backpropagation", que tem se mostrado eficiente na solução de varios problemas, tais como: criptografia, compactação de dados, etc. Pela sua grande capacidade de modelar funções complexas [17] e abilidade de generalização, nos últimos anos tem surgido varias aplicações deste modelo na área de previsões em séries temporais [10,9,7,6], obtendo-se, em alguns casos, bons resultados.

Uma boa modelagem de uma série temporal, através de modelos matemáticos, possibilita a obtenção de previsões de valores futuros dessa série. Esse modelo matemático é sugerido pela análise da série temporal, que tem como objetivo caracterizar e sumarizar as suas propriedades, como por exemplo, a relação entre *observações* em diferentes instantes tempo.

Um dos grandes problemas na aplicação do modelo "Backpropagation" à previsão de séries temporais, é a falta de um *procedimento efetivo* para a obtenção de um método de treinamento, e principalmente de uma arquitetura para a rede, de tal forma que se consiga uma boa modelagem da série, e portanto, uma previsão com maior acurácia possível. Para tal, desenvolvemos um procedimento, baseado na análise da série temporal realizada pela metodologia Box-Jenkins, com o intuito de se obter o número de unidades de entrada da rede neuronal que melhor se adapte ao problema de previsão.

I.1 Objetivo

O objetivo dessa tese é avaliar as redes neuronais, para ser mais preciso, o modelo "Backpropagation", como ferramenta para previsão de séries temporais.

Como a arquitetura das redes neuronais e o procedimento de aprendizagem tem uma grande impacto sobre a modelagem e acurácia das previsões de séries temporais, propomos um procedimento para a obtenção da arquitura da rede, baseada na análise da série temporal em questão. Em relação ao procedimento de

aprendizagem, faremos testes com algumas modificações no aprendizado original do modelo "Backpropagation".

Além de avaliar o nosso procedimento para a obtenção do número de unidades de entrada da rede neuronal, usaremos como parâmetro de avaliação a metodologia Box-Jenkins, por ser uma das mais tradicionais na análise e modelagem de séries temporais.

I.2 Descrição dos capítulos

No capítulo II são apresentados os conceitos e noções básicas de redes neuronais. Com o objetivo de exemplificar o funcionamento e o comportamento de uma rede neuronal simples, são expostos dois tipos de aprendizados: a regra *Hebb*, que é conhecido também como um método de síntese, e a regra *delta*, sempre procurando apresentar um mínimo de fundamentação matemática.

O modelo "Backpropagation", que será utilizado na tese como uma ferramenta para previsão de séries temporais, é apresentado no capítulo III, juntamente com algumas modifições do seu algoritmo. Essas modificações tem como objetivo melhorar a acurácia da modelagem da série temporal pela rede neuronal.

O capítulo IV introduz a Metodologia Box-Jenkins. Por ser uma das mais tradicionais metodologias utilizadas para a análise e modelagem de séries temporais, foi aqui escolhida como um parâmetro de avaliação do modelo "Backpropagation" como ferramenta para previsão de séries temporais. Essa avaliação será realizada através da comparação entre os resultados obtidos pelo modelo desenvolvido pela metodologia Box-Jenkins e o modelo "Backpropagation".

No capítulo V, propomos um procedimento para a utilização do modelo "Backpropagation" na modelagem de séries temporais, com o objetivo de se fazer previsões da série. Uma parte desse procedimento é baseada na análise das séries realizadas na metodologia Box-Jenkins, apresentada no capítulo IV. Além desse procedimento, são aplicados testes com o modelo "Backpropagation" e comparados com os resultados dos modelos obtidos através da metodologia Box-Jenkins.

Capítulo II

Redes Neuronais

II.1 O Sistema Nervoso

O Sistema Nervoso Central do ser humano é constituído pelo cérebro e pela medula espinhal, que tem como função básica possibilitar a comunicação, através de sinais elétricos, entre o cérebro e os outros orgãos do ser humano.

Os neurônios são as células responsáveis por transmitir esses sinais e sua população no cérebro humano é da ordem de vinte e cinco bilhões. Apesar de não existir dois neurônios iguais, eles possuem uma estrutura básica, formada por um corpo celular (*soma*) e prolongamentos chamados dentritos e axônio (Fig.II.1).

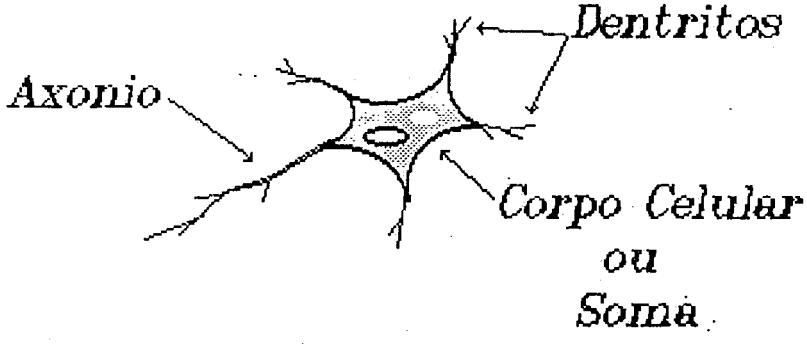

Figura II.1: Esquema simplificado de um neurônio

As conexões entre os neurônios são formadas pelo encontro das ra-

mificações dos axônios com o corpo celular ou dentritos de outros neurônios. Estas junções entre os neurônios são chamadas de *sinapses* (Fig.II.2), e é através delas que os sinais elétricos são propagados na rede neuronal. Além de permitir a união entre vários neurônios, as sinapses funcionam como "válvulas", regulando o fluxo de informações transmitidas na rede neuronal. Acredita-se que as sinapses são responsáveis pelo *conhecimento* e *aprendizado* dos seres vivos. De acordo com essa suposição, as alterações da capacidade de transmissão de informações das sinapses (*eficiência sináptca*) estariam relacionadas com o processo de aprendizado.

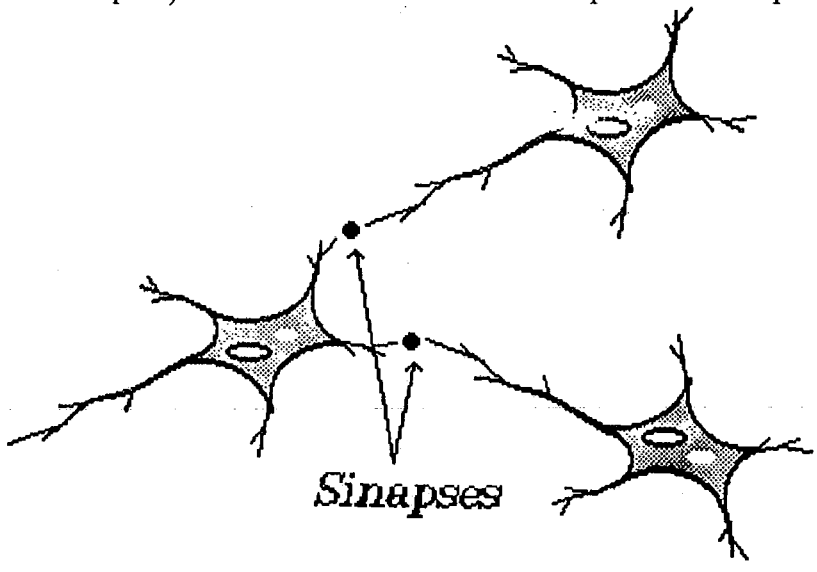

Figura II.2: Esquema simplificado de uma rede neurônal natural

Os dentritos tem como função principal receber as informações, sinais elétricos, de outros neurônios e transmití-las ao corpo celular. Alías informações são processadas e dependendo da intensidade dos sinais recebidos, este pode ou não enviar impulsos elétricos para outros neurônios através do seu axônio (Fig.II.3).

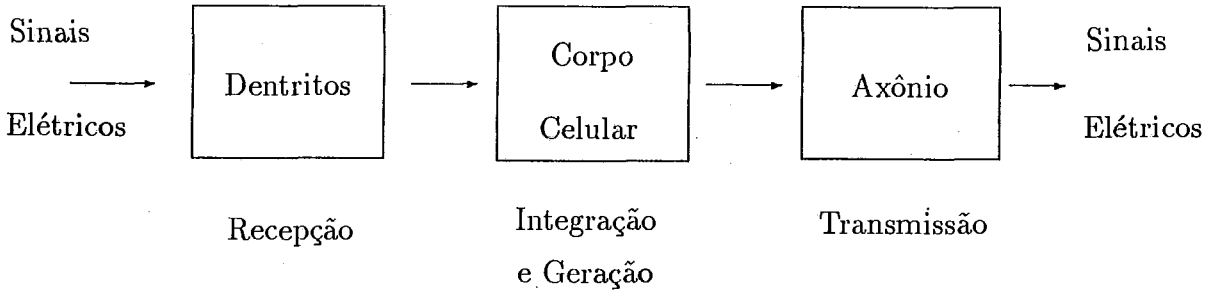

Figura II.3: Diagrama funcional de um neurônio

Os impulsos que chegam dos diversos axônios podem ser integrados,

gerando o *potencial pós-sináptico.* Esta integração dos impulsos pode ser realizada de acordo com dois tipos de sinapses: as excitatórias e as inibitórias. As sinapses excitatórias permitem a passagem de informações entre neurônios, dando um sentido de cooperação entre as células, enquanto que as sinapses inibitórias sugerem uma idéia de competição, impedindo ou dificultando a passagem de informações.

O funcionamento e o tratamento de informações pelos neurônios podem ser encontrados numa forma simples e mais detalhada em [24] e [19].

II.2 Redes Neuronais Artificiais

Há uma grande variedade de redes neuronais artificiais, que podem ser caracterizadas e diferenciadas por uma estrutura, ou conjunto de propriedades básicas. Iremos apresentar, de forma breve, uma nomenclatura que representa as características funcionais de uma rede neuronal, ilustradas na Fig.II.4.

As redes neuronais são formados por um conjunto de unidades de processamento simples, neurônios, que podem representar, isoladamente (representação local) ou através de grupos (representação distribuida), conceitos, símbolos ou características.

Esses neurônios estão interconectados, formando o padrão de conectividade (*eficiência sináptica*) e normalmente atuam em paralelo. Esse padrão é caracterizado pela topologia dessas interconexões, juntamente com seus respectivos pesos, constituindo assim o "conhecimento" do sistema. A representação do padrão de conectividade é realizada através de uma matriz de pesos, denotada por W, onde cada w_{ij} representa o valor do peso da conexão da unidade j para a unidade i, podendo assumir valores positivos, negativos ou nulos, correspondendo, respectivamente, a sinapses excitatórias, sinapses inibitórias e a ausência de sinapse.

Em cada instante de tempo, cada unidade i tem um estado de ativação $a_i(t)$, que expressa o grau de excitação ou inibição do neurônio, determinando o padrão de ativação da rede, fornecendo informações instantâneas do sistema.

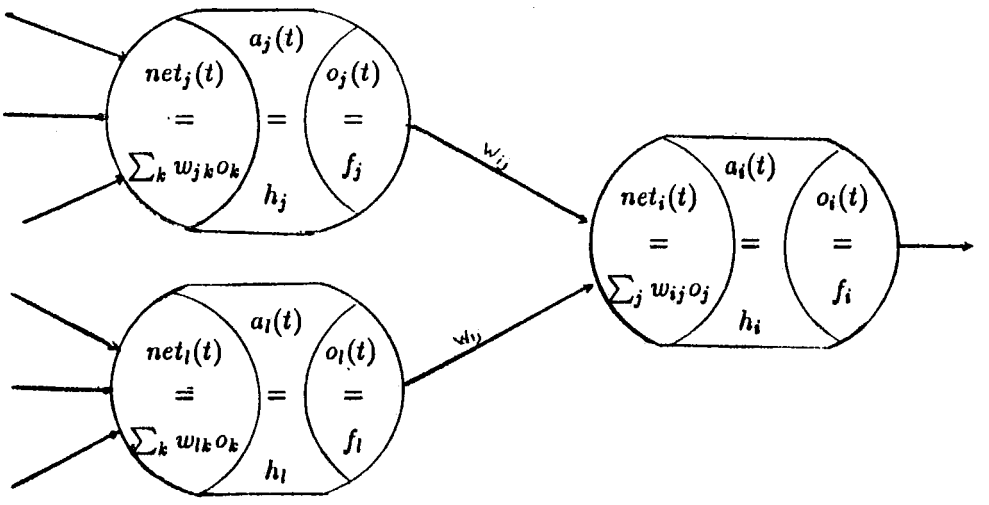

Figura II.4: Esquema de uma rede neuronal artificial

O estado de ativação $a_i(t)$ é fornecido pela regra de ativação h_i através da combinação do impulso total de entrada, net_i, e do estado de ativação anterior, $a_i(t-1)$.

$$a_i(t) = h_i(net_i, a_i(t-1)).$$

A essa ativação, $a_i(t)$, é aplicada a função ou *regra de saída* f_i produzindo o impulso de saída $o_i(t)$, ou seja,

$$f_i(a_i(t)) = o_i(t).$$

A saída (impulso do neurônio) é transmitida para as outras unidades através da *regra de propagação*, p_i, que combina as saídas das unidades conectadas à unidade i com os pesos das respectivas conexões, produzindo assim o impulso total de entrada do neurônio i, $net_i(t)$, correspondente ao *potencial pós-sináptico* das células nervosas.

Alguns modelos de redes neuronais possuem ainda uma *regra de aprendizado*, permitindo que as eficiências sinápticas, neste caso, pesos das conexões, sejam alteradas, possibilitando que a rede neuronial adquira novos "conhecimentos".

Através dessa estrutura básica podemos definir uma infinidade de modelos neuronais. Nesta tese abordaremos apenas os modelos *lineares*, com objetivo introdutório, e os *semi-lineares*.

O modelo do *neurônio linear* ou *modelo linear*, é o mais elementar.

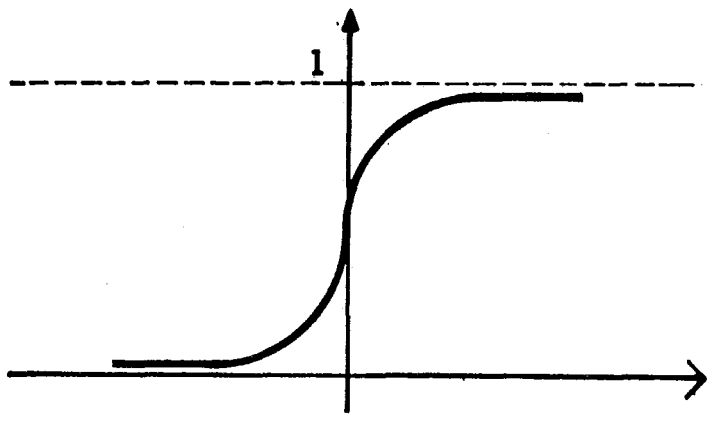

Figura II.5: Função logística

As funções de **ativação**, **propagação** e **saída** são funções lineares, e a *entrada*, a *saída* e o *estado de ativação* são números reais sem qualquer restrição.

Em uma rede neuronal com n neurônios lineares, a **regra de propagação** é dada por

$$net_i(t) = \sum_{j=1}^{n} w_{ij} o_j(t),$$

onde $net_i(t)$ representa uma combinação das saídas $o_j(t)$ recebidas pelo neurônio i através de uma soma algébrica ponderada pelos pesos w_{ij}.

A **regra de ativação** é linear com o impulso total de entrada $net_i(t)$ e independe do estado de ativação anterior do neurônio, isto é,

$$a_i(t) = \alpha net_i(t), \alpha \in R_+.$$

E a **regra de saída** é a identidade:

$$o_i(t) = a_i(t).$$

Os modelos *semi-lineares*, possibilitam uma maior capacidade de aprendizado para as redes do que os *lineares*. Suas regras de propagação e saída são idênticas as dos modelos lineares. A única alteração se dá na *regra de ativação*, que é uma função *sigmoidal*, como por exemplo, a *logística* : $a_i(t) = [1 + e^{net_i(t)}]^{-1}$ (Fig.II.5).

Apresentada a estrutura básica de uma rede neuronal, podemos reconhecer duas fases na sua execução:

- **Fase de Aprendizado.** Nesta fase as conexões apresentam uma característica dinâmica, ou seja, o padrão de conectividade é mudado de acordo com a "experiência", "conhecimento" que a rede adquire, com objetivo de prepará-la para realizar uma certa função corretamente. O modo pelo qual a rede adquire este "conhecimento" é dado pela *regra de aprendizado.*

- **Fase de execução.** Nesta fase o padrão de conectividade é estático, estando a rede, portanto, pronta para executar a sua função (ex. associar padrões).

Sendo o *aprendizado* uma das características mais importantes das redes neuronais, será apresentado, de forma um pouco mais detalhada, nas próximas seções.

II.3 Classificações do aprendizado

Os mecanismos de apredizado podem ser classificados de diversas formas. Deteremonos em apenas duas, as quais acreditamos serem as as mais utilizadas na literatura.

II.3.1 Paradigmas de aprendizado

As regras de aprendizado podem ser divididas em dois grandes grupos:

1. **Aprendizado associativo.** De forma geral, esse tipo de aprendizado ensina a rede a produzir um particular padrão de saída dado um padrão de entrada, permitindo também que se faça uma associação arbitrária entre padrões através do que a rede aprendeu.

2. **Detector de regularidades.** O sistema aprende com o objetivo de descobrir características interessantes no conjunto de entrada, separando-os em categorias.

O aprendizado associativo pode ainda ser subdividido em:

- **associadores de padrões.** A rede aprende asssociações entre padrões de entrada (e) e padrões de saída (o), ajustando o padrão de conectividade , de forma que toda vez que for fornecido à rede o padrão de entrada e, ela retorne o padrão de saída o.

- **auto-associador.** Neste caso a rede aprende a associar o padrão de entrada a ele mesmo. Ao apresentarmos um padrão reduzido ou com distorções, a rede é capaz de recuperar e devolvê-lo em sua forma original.

II.3.2 Supervisionamento

Um outro tipo de classificação possível é quanto ao supervisionamento nas regras de aprendizado.

- **Aprendizado supervisionado.** O sistema é treinado através de exemplos, ou seja, o padrão de entrada é dado juntamente com o padrão de saída desejado. Através de simulações e das respectivas saídas obtidas, os pesos são ajustados com objetivo de se obter, na saída, o padrão desejado.

- **Aprendizado não supervisionado.** Nenhum tipo de informação, além dos padrões de entrada, é fornecido ao sistema. O próprio sistema aprende, sem ter conhecimento prévio da saída desejada.

- **Aprendizado por reforço.** Pode-se dizer que este é um nível intermediário entre o supervisionado e o não supervisionado. São fornecidos estímulos (incrementa os pesos das conexões) ou punições (decrementa os pesos das conexões) de acordo com o comportamento do sistema.

II.4 Regras de aprendizado

O primeiro processo de aprendizado que surgiu em redes neuronais foi o de *tentativas*. O aprendizado era realizado através de sucessivas simulações da rede neuronal, utilizando os resultados (saídas) para a modificação do padrão de conectividade, através da sensibilidade do projetista, até que o desempenho da rede atingisse um

nível satisfatório em relação às associações desejadas de padrões. Por ser um processo altamente empírico e informal, torna-se ineficiente para utilizações práticas.

A primeira regra de aprendizado que se tem conhecimento foi proposta por Donald Hebb [14] em 1949, através de seu postulado :

"Quando uma célula A está próxima o suficiente de uma célula B de forma a excitá-la, algum processo de crescimento sináptico ocorre de maneira a aumentar a capacidade de A excitar B."

Apesar deste postulado não ter um embasamento matemático e nem abordar as conexões inibitórias, ele teve uma grande importância no desenvolvimento do aprendizado nas redes neronais. Pode-se dizer que diversas regras de aprendizado são variantes desse postulado.

Serã0 apresentados a seguir dois processos distintos de aprendizado para redes neuronais, como motivação e introdução ao modelo *"Backpropagation"*, que será apresentado no capítulo III, e utilizado nesta tese. O primeiro processo, que é um método de síntese [24], também chamado de regra de *Hebb* [15], consiste em ajustar a rede neuronal em uma única vez, de forma a executar as associações de padrões desejadas pelo projetista do sistema. O método de síntese é um procedimento matemático que determina o padrão de conectividade, representado pela matriz W, de forma direta e precisa. Por outro lado, o segundo processo, estabelecido por Widrow [26] e chamado de *regra delta*, modifica gradualmente o padrão de conectividade, através de sucessivas aplicações da regra de aprendizado, minimizando a diferença entre a saída obtida pela rede e a saída desejada pelo projetista à cada apresentação dos padrões.

A topologia das redes neuronais que serão abordadas neste capítulo, através desses dois processos, é formada por duas camadas de *neurônios lineares*, *entrada* e *saída*, altamente interligadas (Fig.II.6).

Para facilitar a exposição utilizaremos matrizes para modelar essas redes neuronais e o seu funcionamento. Considere para isso a fig.II.7.

Suponha que existam m unidades de saída e que cada uma esteja

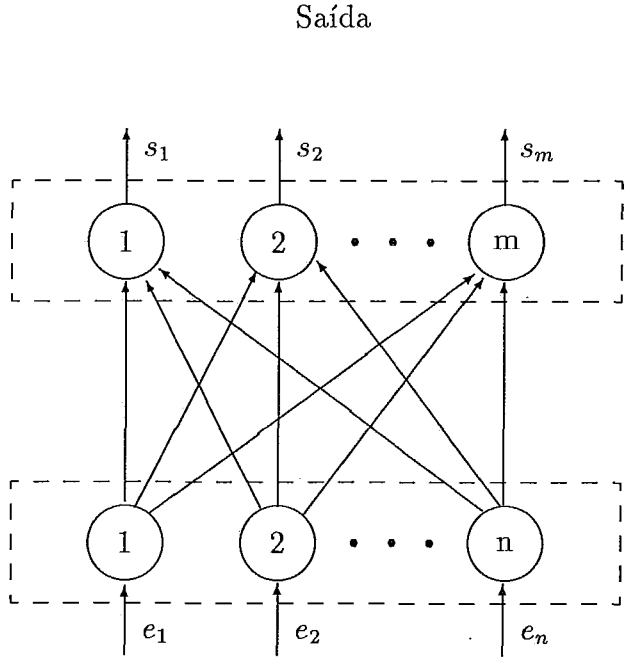

Figura II.6: Rede neuronal de duas camadas

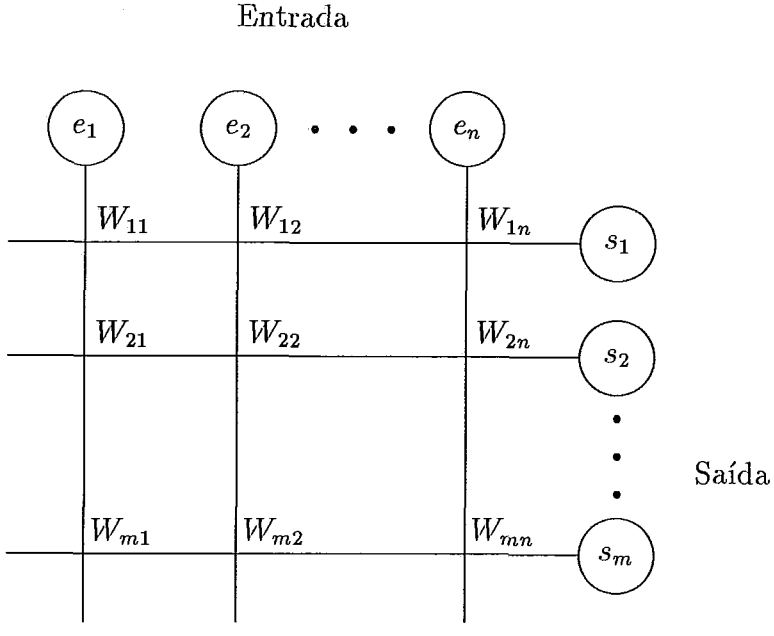

Figura II.7: Rede neuronal em forma de matriz

conectada com todas as n unidades de entrada. Sejam as ativações das unidades de saída e de entrada denotadas respectivamente por $s_1, s_2, ..., s_m$ e $e_1, e_2, ..., e_n$. Como os neurônios são lineares, temos

$$\begin{cases} net_i(t) = \sum_{j=1}^{n} w_{ij} o_j(t); \\ a_i(t) = net_i(t); \\ o_i(t) = a_i(t). \end{cases} \tag{II.1}$$

Dado que a ativação do neurônio de saída i, $a_i(t)$, é igual ao impulso de entrada $net_i(t)$, e que a função de saída é a função identidade; temos que a saída do neurônio i é dada por:

$$s_i = \sum_{j=1}^{n} w_{ij} o_j(t). \tag{II.2}$$

De (II.1) e (II.2)

$$s_i = \sum_{j=1}^{n} w_{ij} e_j(t),$$

pois a entrada e_j constitui a saída do neurônio j. Portanto, podemos modelar matematicamente as redes neuronais lineares de duas camadas pela equação:

$$s = W.e,$$

onde

$$s = \begin{vmatrix} s_1 \\ s_2 \\ \vdots \\ s_m \end{vmatrix},$$

$$e = \begin{vmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{vmatrix},$$

$$w = \begin{vmatrix} w_{11} & w_{12} & \cdots & w_{1n} \\ w_{21} & w_{22} & \cdots & w_{2n} \\ \vdots & \vdots & & \vdots \\ w_{m1} & w_{m2} & \cdots & w_{mn} \end{vmatrix}.$$

É importante ressaltar qua para as redes de três ou mais camadas existe uma rede equivalente com apenas duas camadas [15].

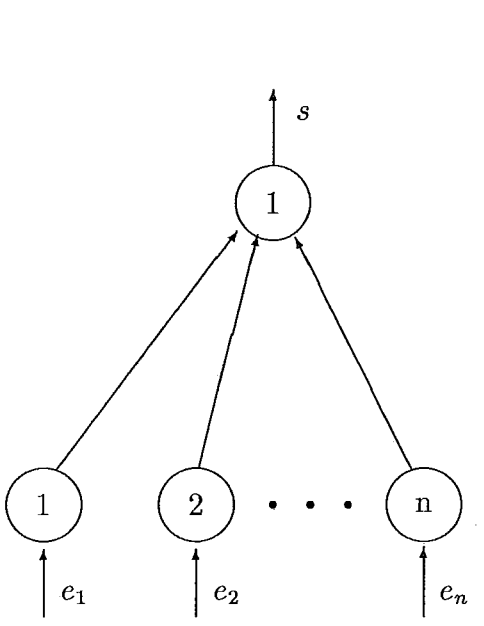

Figura II.8: Rede neuronal com uma unidade de saída

Podemos representar a rede neuronal, com neurônios lineares, através de um sistema de equações, onde w_{ij} são os coeficientes a serem ajustados pela regra de aprendizado (eqação II.3).

$$\begin{cases} w_{11}e_1 + w_{12}e_2 + \cdots w_{1n}e_n = s_1 \\ w_{21}e_1 + w_{22}e_2 + \cdots w_{2n}e_n = s_2 \\ \vdots \quad\quad \vdots \quad\quad\quad \vdots \quad\quad \vdots \\ w_{m1}e_1 + w_{m2}e_2 + \cdots w_{mn}e_n = s_m \end{cases} \tag{II.3}$$

II.4.1 Regra de Hebb

Os métodos de síntese conhecidos até o momento são aplicados apenas aos modelos conexionistas mais simples. Abordaremos apenas um desses métodos, chamado por regra de Hebb [15], aplicável às redes neuronais lineares.

Primeiramente será analisado o caso em que a rede neuronal possue apenas uma unidade de saída, n neurônios de entrada (Fig.II.8) e um padrão para o aprendizado, sendo $e = e_1, e_2, \cdots, e_n$ o padrão de entrada e s o padrão de saída

desejado.

O objetivo é treinar a rede neuronal de tal forma que se ajuste o padrão de conectividade, obtendo a matriz W, onde

$$s = W.e, \tag{II.4}$$

ou seja, dado o padrão de entrada e, a rede teria como saída o padrão s.

Supondo que $\|e\| = e^T.e = 1$, onde $\|e\|$ é a norma de e e e^T é a transposta de e, podemos encontrar o padrão de conectividade W através da equação:

$$W = s.e^T, \tag{II.5}$$

pois, da equação (II.4) e aplicando e^T nos dois lados da igualdade temos

$$s.e^T = W.e.e^T,$$

mas com $e^T.e = 1$,

$$s.e^T = W.$$

Logo, dado os padrões de entrada e saída desejada, s e e, geometricamente o problema se resume em achar um vetor W, tal que a projeção de W sobre e seja igual s. Como mostrado na fig.II.9, todo ponto da reta tracejada satifaz ao problema, pois $s = \|W\|.\frac{W.e}{\|W\|.\|e\|} = W.e$.

Para modelos com m unidades de saída e n unidades de entrada, podemos visualizar a rede neuronal como uma generalização do caso anterior, ou seja, m redes neuronais com apenas uma unidade de saída i (Fig.II.10).

Aplicando a equação (II.5) para cada $s_i, 1 \leq i \leq m$, temos

$$W_i = s_i.e^T,$$

onde W_i é a i-ésima linha da matriz W. Portanto

$$W = s.e^T. \tag{II.6}$$

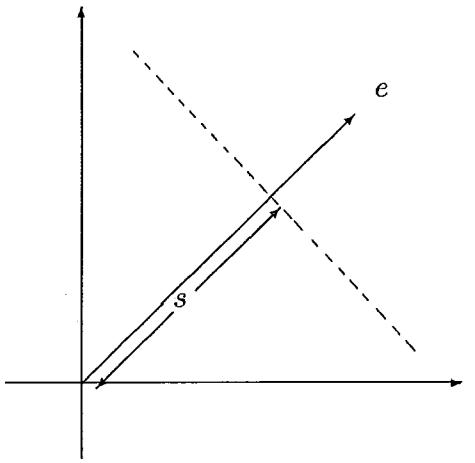

Figura II.9: Solução geométrica

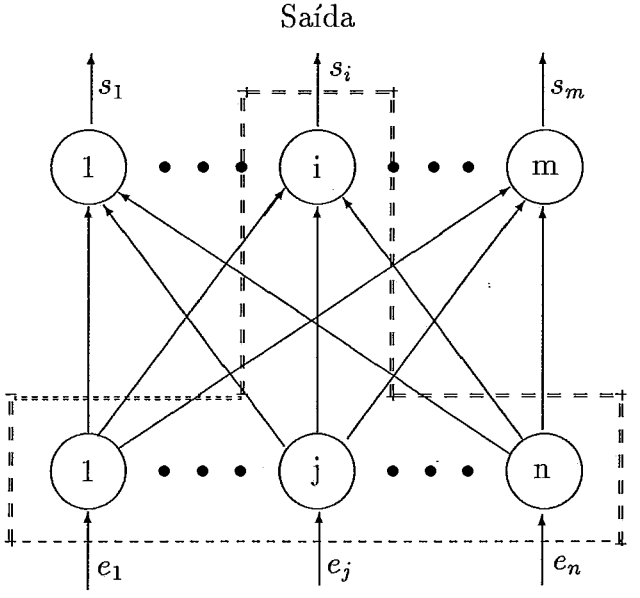

Figura II.10: Rede neuronal com m unidades de saída

Analisamos o aprendizado apenas para uma associação de padrões. Veremos agora o comportamento e as restrições para o treinamento de um conjunto de associações de padrões.

Sejam os z vetores *m-dimensionais* de saída, $s_1, s_2, \cdots, s_z$ no qual desejamos associar com os z vetores *n-dimensionais* de entrada, $e_1, e_2, \cdots, e_z$. Usando a equação (II.6), temos:

$$W_i = s_i e_i^T,$$

para cada padrão i, $1 \leq i \leq z$, formando a matriz W, tal que

$$W = W_1 + W_2 + \cdots + W_i + \cdots + W_z$$

Analisando o aprendizado, para cada $i, 1 \leq i \leq z$,

$$\begin{aligned} We_i &= (W_1 + W_2 + \cdots + W_i + \cdots + W_z)e_i \\ &= (s_1 e_1^T + s_2 e_2^T + \cdots + s_i e_i^T + \cdots + s_z e_z^T)e_i \\ &= (s_1 e_1^T)e_i + (s_2 e_2^T)e_i + \cdots + (s_i e_i^T)e_i + \cdots + (s_z e_z^T)e_i \\ &= s_1(e_1^T e_i) + s_2(e_2^T e_i) + \cdots + s_i(e_i^T e_i) + \cdots + s_z(e_z^T e_i). \end{aligned} \tag{II.7}$$

Como

$$e_i^T . e_i = 1 \quad pois, \quad \|e_i\| = 1,$$

e da equação (II.7)

$$We_i = s_1(e_1^T e_i) + s_2(e_2^T e_i) + \cdots + s_i + \cdots + s_z(e_z^T e_i).$$

Para garantir um aprendizado correto, ou seja, $We_i = s_i$, para todo i,$1 \leq i \leq z$, o conjunto dos padrões de entrada tem que ser ortogonal

$$e_i^T . e_j = 0, \forall i \forall j, 1 \leq i, j \leq z,$$

garantido assim um aprendizado sem "ruídos". Simulando a rede com um padrão arbitrário de entrada e_t, $1 \leq t \leq z$,

$$\begin{aligned} We_t &= (W_1 + W_2 + \cdots + W_i + \cdots + W_z)e_t \\ &= (s_1 e_1^T + s_2 e_2^T + \cdots + s_i e_i^T + \cdots + s_z e_z^T)e_t \\ &= (s_1 e_1^T)e_t + (s_2 e_2^T)e_t + \cdots + (s_i e_i^T)e_t + \cdots + (s_z e_z^T)e_t. \end{aligned} \tag{II.8}$$

Da equação (II.8) podemos obter as seguintes propriedades:

- Se o padrão de entrada do teste e_t for ortogonal ao conjunto de padrões de entrada do aprendizado, então a saída fornecida pela rede para o teste será zero.

- Se o padrão de entrada do teste (e_t) for "similar" apenas a um padrão de entrada do aprendizado, isto é, for ortogonal a todos os outros padrões de aprendizado, menos a este, então a saída fornecida pela rede será proporcional ao produto do padrão de saída s_i, correspondente ao seu "similar", pela norma do produto escalar dos padrões de entrada e_t e e_i.

- Em qualquer outro caso, o padrão de saída será proporcional a uma combinação dos padrões de saída do aprendizado.

Essas redes nos fornecem uma grande gama de aplicações, mas a execução dessa regra Hebb nos leva a uma importante restrição. Se o conjunto formado pelos padrões de entrada do treinamento não for ortogonal, podemos criar uma interferência no aprendizado das associações, ou seja, o aprendizado de um novo padrão pode influir no comportamento da rede em relação aos padrões previamente aprendidos.

II.4.2 Regra delta

Uma forma de relaxar essa restrição sobre a ortogonalidade do conjunto dos padrões de entrada no treinamento, imposta pela *regra de Hebb*, é dada pela *regra delta*. Nela, o aprendizado é proporcional a diferença entre a saída atual e a saída desejada, tendo como objetivo o ajuste gradual dos pesos das conexões para minimizar essa diferença.

Dado um conjunto de p pares de padrões *entrada/saída desejada* (e_l/t_l), $1 \leq l \leq p$, a *regra delta* é dada por

$$\Delta W_{ij} = \eta(t_{li} - o_{li})e_{lj},$$

onde o sub-índice li representa o *i-ésimo* componente do padrão de treinamento de índice l, $1 \leq l \leq p$.

Derivação da regra delta

Nosso principal objetivo é mostrar que a *regra delta* realmente minimiza o erro do sistema, que é dado pela equação abaixo

$$E = \sum_{l=1}^{p} E_l = \sum_{l=1}^{p} \frac{1}{2} \sum_{i=1}^{m} (t_{li} - o_{li})^2, \tag{II.9}$$

onde p é o número de padrões para o aprendizado, m o número de unidades de saída, t_{li} e o_{li} são os i-ésimos elementos da saída desejada e da saída obtida pela rede neuronal.

Para isso deveremos mostrar que as mudanças nos pesos são proporcionais ao simétrico do gradiente de E, ou seja,

$$\Delta W_{ij} = -\frac{\partial E_l}{\partial W_{ij}}.$$

Usaremos unidades lineares, como na regra de Hebb, garantindo assim que a minimização alcance um mínimo global, pois com unidades lineares a função E não possui mínimos locais [15]. Através da regra da cadeia podemos escrever a seguinte igualdade

$$\frac{\partial E_l}{\partial W_{ij}} = \frac{\partial E_l}{\partial o_{li}} \cdot \frac{\partial o_{li}}{\partial W_{ij}}. \tag{II.10}$$

Da equação (II.9) temos que

$$\frac{\partial E_l}{\partial o_{li}} = -(t_{li} - o_{li}). \tag{II.11}$$

Como estamos usando unidades lineares

$$\begin{cases} net_i(t) = \sum_{j=1}^{n} w_{ij} o_j(t); \\ a_i(t) = net_i(t); \\ o_i(t) = a_i(t), \end{cases}$$

portanto

$$o_{li} = \sum_i W_{ij} o_{lj},$$

logo

$$\frac{\partial o_{li}}{\partial W_{ij}} = o_{lj} \tag{II.12}$$

substituindo (II.11) e (II.12) na equação (II.10) obtemos o desejado, que é

$$\frac{\partial E_l}{\partial W_{ij}} = -(t_{li} - o_{li}).e_{lj} = -\Delta W_{ij}$$

mas como

$$\frac{\partial E}{\partial W_{ij}} = \sum_{l=1}^{p} \frac{\partial E_l}{\partial W_{ij}}$$

concluímos, então, que realmente a regra delta implementa o simétrico do gradiente do erro do sistema E.

É importante ressaltar o papel da constante η . Essa constante mede a velocidade do aprendizado, sendo um número positivo (caso contrário teríamos W_{ij} proporcional ao gradiente, maximizando assim o erro). De um modo geral, valores entre 0,8 e 0,9 garantem uma rapidez no aprendizado, mas podem levar a uma oscilação quando se aproximam do ponto de mínimo. Por outro lado, valores entre 0,1 e 0,2 evitam esta oscilação, mas podem levar a uma aprendizagem lenta.

Análise da regra delta

Um dos grandes problemas encontrados na *regra delta* é que em muitos casos é necessário um grande número de aplicações da regra de aprendizado, já que os ajustes dos pesos das conexões são realizados de modo gradual, tornando a execução muito lenta, em contraste com o método de síntese apresentado anteriormente. Além disso, a aplicação dessa regra pode ocasionar oscilações e a medida que há um incremento no número de padrões a serem associados pela rede, é possivel uma interferência nas associações anteriormente aprendidas, pois um novo aprendizado altera o padrão de conectividade estabelecido para o aprendizado anterior.

Outra diferença, é que além de trabalhar com conjuntos ortogonais, a regra delta é capaz de mapear (fornecer um cojunto de pesos) a rede corretamente para associações de padrões, desde que esses pesos existam. Minsky e Papert [12] mostraram que, se existe um padrão de conexão que resolve a associação, então a regra delta é capaz de fornecê-las. Para quais condições então existe esse conjunto de pesos ?

Restrição das redes de duas camadas com neurônios lineares

Esse conjunto só existe se para cada par de padrão *entrada/saída desejada* (e_l/t_l), o padrão de saída desejada puder ser obtido por uma combinação linear das ativações das unidades de entrada, ou seja, o conjunto dos pesos precisa satisfazer

$$t_{li} = \sum_{j=1}^{n} W_{ij} e_{lj},$$

para toda unidade de saída i, e para todo par entrada/saída l. Esta restrição nos garante o aprendizado para cada associação isoladamente. Vejamos agora, o caso em que serão apresentados à rede neuronal z associações de padrões. Sejam os z vetores *m-dimensionais* de saída, $s^1, s^2, \cdots, s^z$ no qual desejamos associar com os z vetores *n-dimensionais* de entrada, $e^1, e^2, \cdots, e^z$. O problema consiste em achar a matriz W que satisfaça a seguinte equação:

$$W.E = S \tag{II.13}$$

onde

$$W = \begin{vmatrix} w_{11} & w_{12} & \cdots & w_{1m} \\ w_{21} & w_{22} & \cdots & w_{2m} \\ \vdots & & & \vdots \\ w_{n1} & w_{n2} & \cdots & w_{nm} \end{vmatrix},$$

$$E = \begin{vmatrix} e_1^1 & e_1^2 & \cdots & e_1^z \\ e_2^1 & e_2^2 & \cdots & e_2^z \\ \vdots & \vdots & & \vdots \\ e_n^1 & e_n^2 & \cdots & e_n^z \end{vmatrix},$$

$$S = \begin{vmatrix} s_1^1 & s_1^2 & \cdots & s_1^z \\ s_2^1 & s_2^2 & \cdots & s_2^z \\ \vdots & \vdots & & \vdots \\ s_m^1 & s_m^2 & \cdots & s_m^z \end{vmatrix}$$

tal que s_i^j é o i-ésimo elemento do padrão de saída s^j e e_i^j é o i-ésimo elemento do padrão de entrada e^j.

A solução da equação (II.13) é

$$W = S.E^+,$$

tal que E^+ é a matriz *pseudo-inversa* de E, sendo necessário que as colunas da matriz E sejam *linearmente independentes* [15]. Portanto, para assegurar que qualquer associação de um conjunto de padrões possa ser aprendido corretamente, esse conjunto precisa ser linearmente independente, ou seja, nenhum elemento pode ser escrito como combinação dos outros. Nesses casos, a representação externa torna-se fundamental.

Um exemplo clássico é o problema do *ou exclusivo (XOR)*, ilustrado na tabela II.1. Como o padrão 4 pode ser obtido por uma combinação linear dos padrões 2 e 3, o conjunto dos padrões não é linearmente independente, impossibilitando um aprendizado correto. Esse problema pode ser contornado, modificando-se a representação dos padrões de entrada, obtendo-se um conjunto linearmente independente, como mostra a tabela 2.

	Padrão de entrada		*Padrão de saída*
1	(0,0)	→	0
2	(0,1)	→	1
3	(1,0)	→	1
4	(1,1)	→	0

Tabela II.1: XOR - 2 unidades de entrada

	Padrão de entrada		*Padrão de saída*
1	(0,0,0)	→	0
2	(0,1,0)	→	1
3	(1,0,0)	→	1
4	(1,1,1)	→	0

Tabela II.2: XOR - 3 unidades de entrada

Capítulo III

Modelo "Backpropagation"

III.1 Histórico

O modelo "Backpropagation" foi desenvolvido independentemente por varios pesquisadores. Alguns , como Bryson e Ho em 1969, Werbos em 1974 e Parker em 1982, já apresentavam estudos e algoritmos similares ao "Backpropagation", mas foi em 1985 que Rumelhart, Hinton e Willians do grupo PDP exploraram o potencial e divulgaram o modelo, tornando-o um dos mais utilizados modelos de redes neuronais.

III.2 Motivação

Os modelos neuronais que possuem redes com duas camadas, *entrada e saída*, possibilitam uma grande variedade de aplicações. Mas pelo fato de não haver uma representação interna, estas redes só conseguem mapear padrões de entrada com padrões de saída quando estes são similares [15], tornando imprescindivel uma representação adequada dos padrões em relação a topologia da rede, como no exemplo do **XOR** apresentada no capítulo anterior. No caso de redes em que os neurônios são linares, essa similaridade se restringe ao fato que o padrão de saída deve ser uma combinação linear dos elementos do padrão de entrada. Além disso o conjunto de padrões de entrada tem que ser *independente.*

Minsky and Papert [12] mostraram que como o problema do **ou ex-**

clusivo, muitos outros não poderiam ser mapeados pelas redes de duas camadas. Por outro lado, a criação de uma camada "intermediária", possibilitando assim uma representação interna, acabaria com essa limitação na representação dos padrões. Minsky e Papert provaram então que: *se há uma camada "intermediária", conectada corretamente com as camadas de "entrada" e "saída" e com um tamanho suficiente, então há sempre uma representação interna do padrões de entrada, através da camada "intermediária", no qual a sua similaridade possibilita um mapeamento das unidades de entrada com os de saída.*

Para o aprendizado destas redes foi criado uma variante da regra delta, pois como atualizariamos os pesos da camada "intermediária", já que não se dispõe das informações à respeito do erro em cada unidade intermediária, tendo que não está caracterizado qual é a saída desejada para estas unidades.

Esta variante, chamada *regra delta generalizada*, é dada pela equação:

$$\Delta W_{ij} = \eta \delta_{li} o_{li} \tag{III.1}$$

onde

$$\delta_{li} = \begin{cases} (t_{li} - o_{li}).f_i'(net_{li}), & \text{se } i \text{ é unidade de saída} \\ f_i'(net_{li}) \sum_{k=1}^{m} \delta_{lk} W_{ik}, & \text{se } i \text{ é unidade "intermediária"} \end{cases}$$

e f é a função de ativação, f' sua derivada e η é a *a constante de aprendizado.* Para as unidades de saída, a regra delta generalizada (equação (III.1)) utiliza a diferença entre a saída obtida e a desejada, como na regra delta, enquanto que para a camada "intermediária" essa diferença é retropropagada proporcionalmente ao valor dos pesos das conexões entre as unidades. O erro então é calculado com base na sua "contribuição" para essa diferença.

O *"Backpropagation"* é um modelo de múltiplas camadas, onde cada neurônio de uma camada está totalmente conectado aos neurônios da camada imediatamente posterior, como ilustra a fig.II.1. Sendo utilizado a *regra delta generalizada* como a regra de aprendizado, que retropropaga os erros da camada de saída para as a camamdas anteriores.

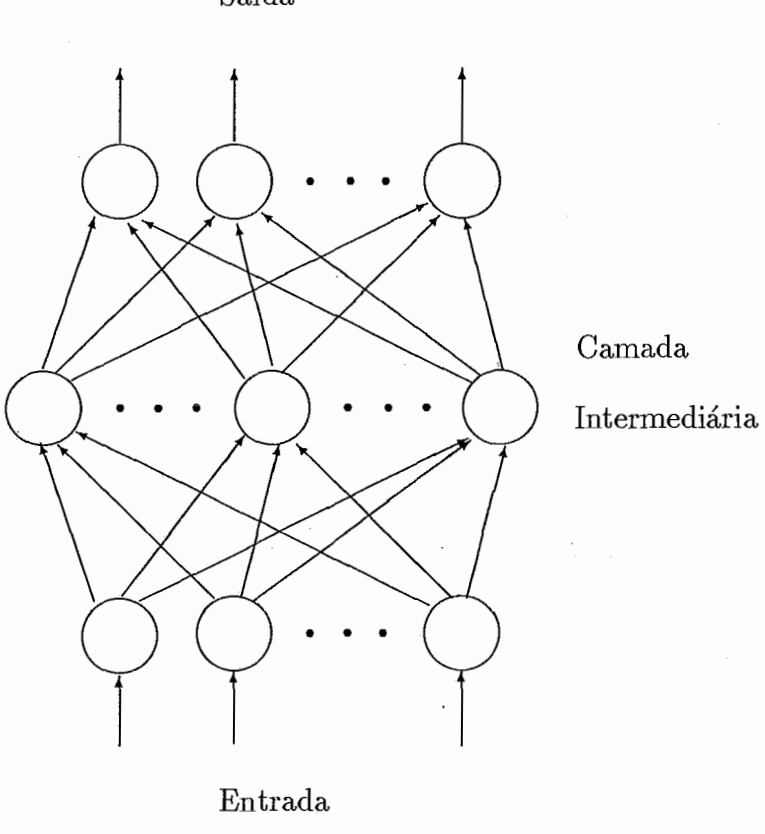

Figura III.1: Rede de multiplas camadas

III.3 Função de ativação

As unidades lineares não são utilizadas neste caso, já que com elas, as redes de três camadas não teriam nenhuma vantagem sobre as de duas. Pois, para qualquer rede com unidades lineares é possível construir uma rede equivalente com apenas duas camadas [15].

Usaremos os *neurônios semi-lineares*:

$$\begin{cases} net_{li} = \sum_j W_{ij} o_{li} + bias_i; & \text{se } i \text{ é unidade de entrada, } o_{li} = e_{li} \\ o_{li} = a_{li}; \\ a_{li} = f_i(net_{li}), \end{cases} \tag{III.2}$$

onde f é diferenciavel e não decrescente e $bias_i$ é um tipo de unidade que possui ativação constante e igual a *um* . A função mais comumente usada é a *logistica* (Figura III.2) :

$$f_i(net_{li}) = [1 + e^{-net_{li}}]^{-1}.$$

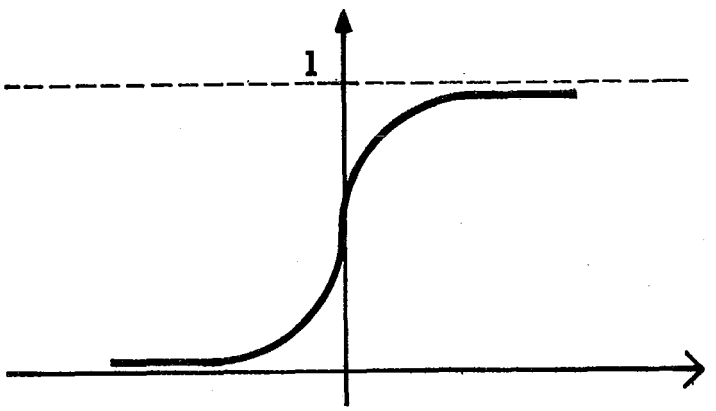

Figura III.2: Função logistica

III.4 Derivação da regra delta generalizada

Analogamente ao capítulo anterior, mostraremos que a regra delta generalizada também minimiza o erro do sistema E (dado pela equação (II.9)). Utilizando a regra da cadeia, temos

$$\frac{\partial E_l}{\partial W_{ij}} = \frac{\partial E_l}{\partial net_{li}} \cdot \frac{\partial net_{li}}{\partial W_{ij}}. \tag{III.3}$$

Da equação (III.2)

$$\frac{\partial net_{li}}{\partial W_{ij}} = o_{li}. \tag{III.4}$$

Definindo

$$\delta_{li} = \frac{\partial E_l}{\partial net_{li}},$$

e de (III.3) e (III.4) obtemos

$$\frac{\partial E_l}{\partial W_{ij}} = \delta_{li} . o_{li}. \tag{III.5}$$

Logo para implementar o simétrico do gradiente de E, o ajuste de pesos será dado por

$$\Delta W_{ij} = \eta \delta_{li} . o_{li}. \tag{III.6}$$

Examinaremos agora δ_{li}, para isto usaremos mais uma vez a regra da cadeia

$$\delta_{li} = -\frac{\partial E_l}{\partial net_{li}} = -\frac{\partial E_l}{\partial o_{li}} . \frac{\partial o_{li}}{\partial net_{li}}. \tag{III.7}$$

Da equação (III.2) temos

$$\frac{\partial o_{li}}{\partial net_{li}} = f_i^{'}(net_{li}). \tag{III.8}$$

Agora para examinar o primeiro fator da equação (III.7) consideraremos dois casos:

1. i pertence a camada de "saída":

   Do erro do sistema, equação (II.24)

   $$\frac{\partial E_l}{\partial o_{li}} E_l = -(t_{li} - o_{li})$$

   obtemos, portanto

   $$\delta_{li} = (t_{li} - o_{li}) . f_i^{'}(net_{li}).$$

2. i pertence a camada "intermediária":

   Usando a regra da cadeia e a equação (III.2) temos que

$$\begin{aligned} \frac{\partial E_l}{\partial o_{li}} &= \sum_k \frac{\partial E_l}{\partial net_{lk}} . \frac{\partial net_{lk}}{\partial o_{li}} \\ &= \sum_k \frac{\partial E_l}{\partial net_{lk}} . \frac{\partial}{\partial o_{li}} \sum_j W_{kj} o_{lj} \\ &= \sum_k \frac{\partial E_l}{\partial net_{lk}} . W_{ki} \\ &= -\sum_k \delta_{lk} W_{ki}, \end{aligned} \tag{III.9}$$

logo, substituindo a equação acima e a (III.8) em (III.9) temos

$$\delta_{li} = f_i^{'}(net_{li}) \sum_k \delta_{lk} W_{ki}.$$

Dada a regra de aprendizado, podemos resumir o procedimento para a utilização e o funcionamento do modelo *"Backpropagation"* da seguinte forma:

1. Seleciona-se os padrões (dados) de entrada e saída desejada para serem fornecidos à rede;

2. escolhe-se a arquitetura da rede:
   - quantidades de camadas,
   - número de unidades para cada camada;
3. inicializa-se os pesos das conexões aleatoriamente;
4. apresenta-se os padrões de entrada e, propagando-os através da rede, e computando os valores t_{li} obtidos para cada unidade da camada de saída. Através dessa saída t_{li} é calculada o erro da unidade, obtendo δ_{li} pela equação

   $$\delta_{li} = (t_{li} - o_{li}).f'.(net_{li});$$

5. propaga-se o sinal de erro δ_{li} para as camadas intermediárias, obtendo-se o erro δ_{lj} para cada unidade j dessa camada por

   $$\delta lj = f'_j(net_{lj}) \sum_i \delta_{li}.w_{ij};$$

6. calculado os erros de cada unidade, obtem-se o ajuste dos pesos das conexões pela equação

   $$\Delta W_{ij} = \eta.\delta_{li}.o_{li}$$

7. repete-se os passos 4,5 e 6 até que o erro de cada unidade tenha atingido um valor aceitável.

III.5 Desempenho da rede

Um dos problemas em se utilizar o modelo *"Backpropagation"* é a complexidade do algoritmo de aprendizado. Geralmente varias apresentações (iterações) do conjunto de padrões são necessários para a realização de um aprendizado preciso. Por se tratar de um problema real, que pode algumas vezes invibializar o uso do modelo, alguns refinamentos e alterações surgiram com o intuito de acelerar a convergência [2,8].

Outro problema encontrado, é que a hipersuperfície do erro pode conter mínimos locais, consequentemente o algoritmo de aprendizado, que é baseado no método do gradiente, pode obter um mínimo local ao invés do global.

Segundo o grupo PDP [15], o conjunto inicial de pesos, atribuídos as conexões aleatoriamente, possui um papel importante em relação a obtenção do mínimo local ou global. Pois pesos muito grandes podem saturar os neurônios, levando a um mínimo local perto do ponto inicial, estabilizando rapidamente o erro do aprendizado.

Uma estratégia proposta pelo grupo PDP é a de inicializar os pesos w_{ij} aleatoriamente, dentro do intervalo $(\frac{-1}{\sqrt{k_i}}, \frac{1}{\sqrt{k_i}})$, onde k_i é o número de unidades j que estão conectadas com a unidade i pelas conexões do tipo $j \rightarrow i$.

Serão apresentados a seguir alguns procedimentos diferentes no aprendizado, que foram utilizados nesta tese, não só com o objetivo de se acelerar a convergência do aprendizado ou de se evitar mínimos locais, mais principalmente com o intuíto de se ajustar a rede o melhor possível ao problema de previsão em séries temporais.

III.5.1 Termo de Momento

A *constante de aprendizado* η é responsável pela velocidade do aprendizado. Quanto maior o valor dessa constante maior é a variação dos pesos das conexões, tornando teoricamente mais rápido o aprendizado. Mas, constantes grandes podem levar a uma oscilação quando a função do erro se aproxima do ponto de mínimo. Por outro lado constantes pequenas podem levar a uma aprendizagem demasiadamente lenta. Com o objetivo de se acelerar o aprendizado sem levar a uma oscilação, o grupo PDP [15] propôs uma pequena modificação na *regra delta generalizada*, acrescentando um *termo de momento*:

$$\Delta W_{ij}(n+1) = \eta \delta_{li} o_{li} + \alpha \Delta W_{ij}(n),$$

onde α é a *constante de momento* que determina o efeito do ajuste anterior $(W_{ij}(n))$ dos pesos em relação ao ajuste corrente $(W_{ij}(n+1))$. Se a último ajuste foi numa particular direção, o *termo de momento* infuenciará para que a próximo ajuste seja na mesma direção. Deste momento em diante, ao nos referirmos ao algortimo de aprendizado do modelo "Backpropagation",*regra delta generalizada*, estará incluso o termo de momento.

III.5.2 Normal

Neste caso, as conexões são ajustadas a cada apresentação dos padrões (entrada / saída desejada) à rede. Sendo que esses padrões são apresentados em uma ordem fixa e predeterminada.

III.5.3 Atualização acumulativa de pesos

Uma das técnicas utilizadas para acelerar a convergêncaia da rede é o de apenas ajustar os pesos das conexões após todo o conjunto de padrões ter sido apresentado à rede, ao invés de se ajustar à cada apresentação.

Quando o conjunto de padrões não é muito grande essa técnica pode ser eficaz, pois sempre caminha na direção da redução do erro global, enquanto que atualizações individuais podem reduzir o erro para um padrão particular e aumentar para os outros. Por outro lado, se houver muita redundância e o conjunto de padrões for grande, os ajustes podem assumir valores absolutos grandes, provocando oscilações, se a taxa de aprendizado não for pequena o suficiente. Portanto, quando o conjunto de padrões for grande, atualizações a cada subconjunto pode ser mais eficiente.

III.5.4 Apresentação aletória

Em relação ao mínimo local, uma técnica simples e bastante utilizada é o de fornecer os padrões à rede numa ordem aletória à cada apresentação do conjunto de padrões. Essa técnica evita um média sobre os padrões e os cancelamentos de ajustes, que podem ocorrer quando o aprendizado é realizado ciclicamente com uma sequência fixa na apresetação dos padrões.

III.5.5 Erro mínimo

Durante o aprendizado os erros são retropropagados somente para os padrões no qual o erro de saída exceder a um dado valor tolerância, desconsiderando-se assim,

os erros menores do que um dado limite, assumindo-os iqual a zero, e portanto, não realizando o ajuste dos pesos para esses casos.

Esse procedimento reduz o número de cálculos e teóricamente fornece um mapeamento dos padrões mais suave.

III.6 Implentação

O modelo "Backpropagation", juntamente com as cincos modificações apresentadas acima, foram implentados na linguagem de programação "C". Foram utilizadas as estações de trabalho SUN, devido a sua alta velocidade de processamento, minimizando uns dos grandes problemas do aprendizado, que é o grande número de cálculos a serem realizados nas inúmeras iterações, tornado necessário um tempo de execução muito grande para os micros PC AT.

Capítulo IV

Box-Jenkins

IV.1 Séries Temporais

Uma série temporal é formada por um conjunto de observações realizadas sequencialmente no tempo. As *observações* são dependentes e o estudo de uma série consiste em analisar e modelar esta dependência.

As séries temporais são representadas por um sequência de observações $z(t)/t \in T$ de uma variável z, onde T é um conjunto de índices, no nosso caso *tempo* discreto.

As séries podem ser discretas, contínuas, multivariadas ou multidimensionais [22]. Neste trabalho nos preocuparemos apenas com as séries discretas e unidimensionais, onde T será o conjunto dos tempos das observações realizadas em intervalos eqüidistantes, sendo as séries denotadas por $z_1, z_2, \cdots, z_n$, onde n é o número de observações.

IV.2 Metodologia Box-Jenkins

Um dos fatores mais importantes num programa de planejamento é o "conhecimento" das condições e possibilidades gerais pertinentes ao fenômeno, tanto no presente quanto no *futuro*. Portanto é de suma importância realizar uma previsão que venha estar o mais próximo o possível da realidade, possibilitando um planejamento inteligente e de sucesso.

Através da análise de uma série temporal é possível sumarizar suas propriedades e caracterizar seu comportamento identificando ou sugerindo um modelo matemático que a descreva, tornando possível previsões de valores futuros.

Um dos métodos mais tradicionais e utilizados para a análise e modelagem de séries de temporais é o sugerido por Box e Jenkins [18]. Essa técnica se diferencia das outras por:

- tentar minimizar a necessidade de um especialista no desenvolvimento do modelo, tornando-o o mais sistemático possível.
- eliminar ou ajustar , através de certos procedimentos, modelos inapropriados até chegar ao desejado.

Serão usados aqui algumas terminologias de estatística sem definições previas, que podem ser encontradas, caso hajam dúvidas, na maioria dos livros do assunto. As técnicas de previsão utilizando séries temporais são baseadas na repetição do passado no futuro. Essas séries podem ser divididas em duas classes:

1. **Estacionária.** Se caracterizam por manterem um equilíbrio através de uma *média* e *variância* constantes (fig.IV.1).

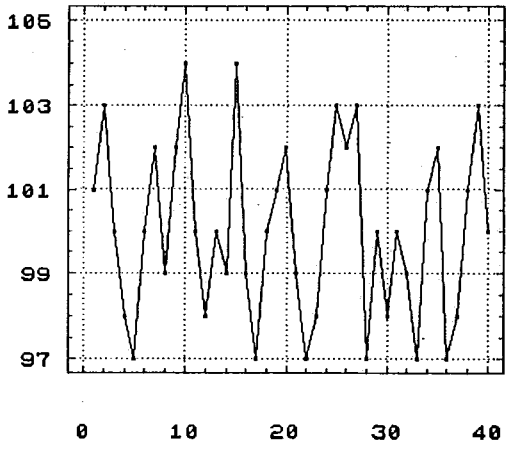

Figura IV.1: Série estacionária

2. **Não estacionária.** Têm uma fundamental importância por serem frequentemente encontradas na prática. Se caracterizam pela alteração da média ou variância ao longo do tempo (fig.IV.2).

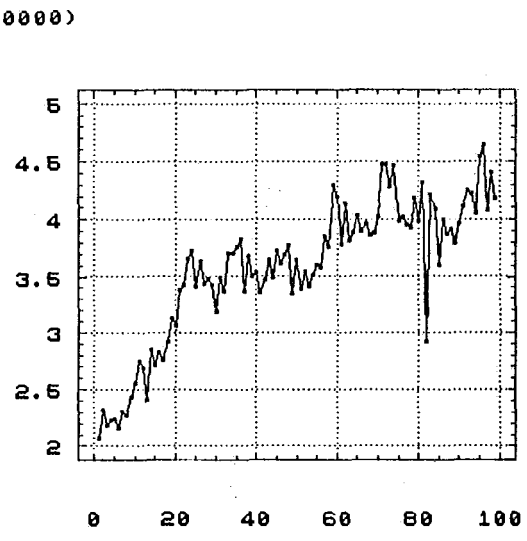

Figura IV.2: Série não estacionária

O desenvolvimento de modelos através da metodologia *Box-Jenkins* envolve quatro fases (Fig.IV.3):

1. identificação do modelo;
2. estimativa dos parâmetros;
3. Diagnóstico e ajuste (se necessário) do modelo proposto e
4. previsão baseada no modelo escolhido.

IV.3 Séries estacionárias e suas propriedades

A característica central do desenvolvimento de um modelo é o *equilíbrio estatístico.* Usualmente uma série *estacionária* pode ser descrita pela sua média e variância , pois dado uma série estacionária $z_1, z_2, \cdots, z_n$, a sua *média* μ é constante e os erros de previsão, $a_t = z_t - \hat{z}_t$, são assumidos serem distribuídos aleatoriamente com média zero e variância constante, σ^2. A seguir apresentaremos, sucintamente, os modelos utilizados pela metodologia Box-Jenkins para descrever as séries estacionárias.

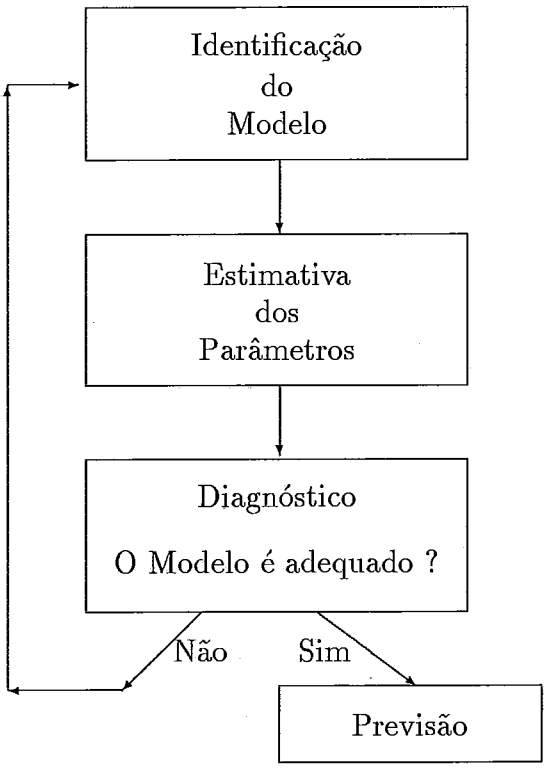

Figura IV.3: Fases da metodologia Box-Jenkins

IV.3.1 Modelo Auto-regressivo

O modelo *auto-regressivo* é muito importante na representação de certas séries. Neste modelo, o valor corrente de uma série é expresso como uma combinação linear finita de valores anteriores acrescida de um erro a_t. Denotando as observações de uma série por $z_1, z_2, \cdots, z_n$ e os desvios em relação à μ por $Y_1, Y_2, \cdots, Y_n$, ou seja, $Y_i = z_i - \mu$ para $1 \leq i \leq n$, então

$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + a_t \tag{IV.1}$$

é chamado de *modelo auto-regressivo* de ordem p, *(MA(p))* e $\phi_i, 1 \leq i \leq p$, é um parâmetro que pode ser estimado através de uma regressão linear [18] ou através de rotinas iterativas que utilizam o procedimento não linear ou mínimos quadrados [22]. Se difinirmos o *operador auto-regressivo* de ordem p por

$$\phi_p(B) = 1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p, \tag{IV.2}$$

onde o operador B^i é tal que $B^i z_t = z_{t-i}$, podemos reescrever a equação (IV.1) como

$$\phi_p(B) Y_t = a_t. \tag{IV.3}$$

IV.3.2 Modelos Médias Móveis

Neste modelo, os erros de previsão são considerados como importantes informações para futuras predições. O valor corrente de uma série pode ser expresso através de uma combinação linear finita dos erros de previsões anteriores. O *modelo média móvel* de *ordem* q é dado por

$$Y_t = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \cdots - \theta_q a_{t-q} \tag{IV.4}$$

onde $\theta_i, 1 \leq i \leq q$, é um parâmetro que pode ser estimado através do *método dos mínimos quadrados* ou da *estimação não linear*, entre outros métodos [22]. Se difinirmos o *operador média móvel* de ordem q por

$$\theta_q(B) = 1 - \theta_1(B) - \theta_2 B^2 - \cdots - \theta_q B^q, \tag{IV.5}$$

podemos então reescrever a equação (IV.4) como

$$Y_t = \theta_q(B) a_t. \tag{IV.6}$$

IV.3.3 Modelo ARMA

Para possibilitar uma maior flexibilidade e ajuste às séries temporais combinaram-se os modelos *auto-regressivos (AR(p)) e média móvel (MA(q))* obtendo-se o modelo $ARMA(p,q)$, dado por

$$\phi_p(B) Y_t = \theta_q(B) a_t \tag{IV.7}$$

onde $\phi_p(B)Y_t$ representa o operador *auto-regressivo de ordem* p e $\theta_q(B)a_t$ o operador *média móvel de ordem* q. Na prática, é frequente obter representações adequadas de séries estacionárias através dos modelos auto-regressivo *(AR(p))*, média móvel *(MA(q))* ou *ARMA(p,q)*, no qual p e q não são maiores que 2.

IV.3.4 Ordem dos modelos

Na construção de um modelo de uma série temporal é necessário conhecer o relacionamento entre a observação corrente e as anteriores. Uma das ferramentas utilizadas

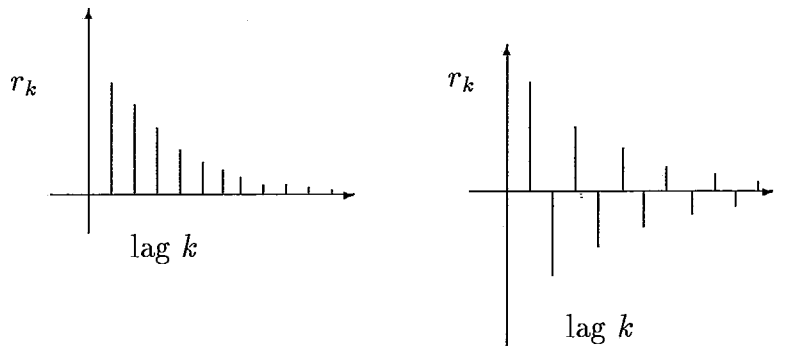

Figura IV.4: Função de autocorrelação do modelo *AR(1)*

para tal é a função de autocorrelação:

$$r_k = \frac{\sum_{t=1}^{n-k}(Y_t - \bar{Y})(Y_{t+k} - \bar{Y})}{\sum_{t=1}^{n}(Y_t - \bar{Y})^2}, k = 0, 1, \cdots, v. \tag{IV.8}$$

onde Y_t é o *desvio*, ou melhor, a diferença entre a observação t e a média μ, $Y_t = z_t - \mu$, $\bar{Y}$ é a média dos desvios e v é o número de "lags", não precisando ser maior que $\frac{n}{4}$ [18], tal que n é o número de *observações* da série.

Nas séries estacionárias a função de autocorrelação, através do seu comportamento, nos sugere o uso dos modelos *auto-regressivo* ou *média móvel.*

Em geral, o gráfico da função de autocorrelação de um processo *auto-regressivo* consistirá de uma combinação de decaimento exponencial com um decaimento senoidal.

Quando uma série é melhor descrita por processo *auto-regressivo* de primeira ordem, $z_t = \phi_1 z_{t-1} + a_t$, a função de autocorrelação decai exponecialmente para zero (ϕ_1 é positivo) ou decai oscilando, quando ϕ_1 é negativo (fig.IV.4).

Quanto ao processo *auto-regressivo* de segundo ordem, a função de autocorrelação é descrita por um decaimento senoidal, como na fig.IV.5.

A autocorrelação nem sempre indica a ordem exata do modelo *auto-regressivo.* Para suprir essa deficiência é usado frequentemente a autocorrelação

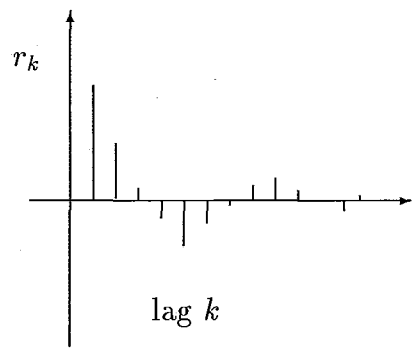

Figura IV.5: Função de autocorrelação do modelo *AR(2)*

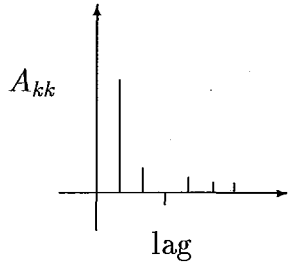

Figura IV.6: Função de autocorrelação parcial do modelo *AR(1)*

parcial, dada por

$$A_{kk} = \begin{cases} r_1 & se \quad \mathrm{k} = 1 \\ r_k - \frac{\sum_{j=1}^{k-1} A_{k-1,j} r_{k-j}}{\sum_{j=1}^{k-1} A_{k-1,j} r_j} & se \quad \mathrm{k} = 0, 1, \cdots, v, \end{cases} \tag{IV.9}$$

onde r_i é a função de autocorrelação aplicada ao "lag" i e v é o número de "lags", não precisando ser maior que $\frac{n}{6}$ [18], tal que n é o número de *observações* da série.

Do mesmo modo que a função de autocorrelação, a função de autocorrelação parcial mede a correlação entre as *observações* das séries temporais, sendo que neste caso a relação entre as *observações* z_t e z_{t-p} é dada pelo valor da função de autocorrelação parcial no *p-ésimo "lag"*. Por exemplo, se a *observação* z_t está diretamente relacionado apenas com a *observação* z_{t-1}, teremos um valor grande para o "lag" 1, e pequeno para os outros "lags", indicando portanto o modelo *AR(1)* (fig.IV.6).

Em geral, podemos utilizar as funções de autcorrelação e de autocorrelação parcial para a identificação da ordem do *modelo auto-regressivo* da seguinte

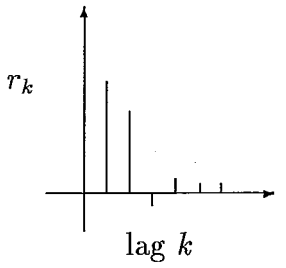

Figura IV.7: Função de autocorrelação do modelo *MA(2)*

forma:

- uma série temporal é melhor descrita pelo modelo AR(p) quando a função de autocorrelação exibe um padrão semelhante aos apresentados nas fig.IV.4 ou IV.5, e a função de autocorrelação parcial tem um valor absoluto grande no *p-ésimo* "lag".

É importante ressaltar que no caso da identifiação da ordem do Média Móvel é suficiente analisar a função de autocorrelação, onde um "lag" com valor absoluto grande ("pico"), que destoe dos outros, no *p-ésimo "lag"* indica MA(p).

Por exemplo, o gráfico da função de autocorrelação apresentado na fig.IV.7 possue "picos" nos "lags" 1 e 2, indicando o modelo *média móvel de ordem 2* - $\theta_2(B) = 1 - \theta_1(B) - \theta_2 B^2$.

IV.4 Séries não estacionárias

Na prática muitas séries temporais não apresentam um comportamento estacionário, ou seja, possuem média e variância variáveis através do tempo. Porém muitas dessas séries apresentam uma certa homogeneidade, podendo em alguns casos ser divididas em partes que possuem comportamentos semelhantes. Essas séries podem ser transformadas em séries estacionárias através da aplicação da operação de *diferença* entre as diversas observações da série original, possibilitando assim utilizar todas as ferramentas anteriormente apresentadas para análise das séries estacionárias.

Definiremos a *"diferença regular" de primeira ordem* por

$$Y_t = z_t - z_{t-1} = (1 - B)z_t \tag{IV.10}$$

Em muitos casos, a aplicação da *"diferença regular"* de primeira ordem pode não ser suficiente para se obter uma nova série estacionária, requerendo aaplicação da *"diferença regular"* de ordem superior d, dada por

$$Y_t = (1 - B)^d z_t \tag{IV.11}$$

Quando a série apresenta um forte componente *sazonal*, ou melhor, exibem uma variação periódica, ocorrendo similaridades em intervalos de tempo de comprimento c, como por exemplo, medidas de temperatura, que possue uma variação anual, pode ser necessário a aplicação da *"diferença sazonal"* de ordem dp, definada por

$$Y_t = (1 - B^c)^{dp} z_t \tag{IV.12}$$

Generalizando,

$$Y_t = \begin{cases} (1 - B)^d (1 - B^c)^{dp} z_t & \text{se } d > 0 \text{ ou } dp > 0 \\ z_t - \mu & \text{se } d = 0 \text{ e } dp = 0. \end{cases} \tag{IV.13}$$

O apropriado nível (ordem) da "diferença" necessário para se obter uma série estacionária a partir da original, é obtido, normalmente, através de sucessivas aplicações de combinações da "diferença regular" com a "diferença sazonal" até que a função de autocorrelação apresente um decaimento rápido e valores absolutos pequenos com o incremento dos "lags", caracterizando uma série estacionária. Na prática, a *ordem* não ultrapassa a dois.

Um modelo capaz de representar uma grande classe dessas séries temporais é o $ARIMA(p,ps,d,dp,qs,q)$, definido por

$$\phi_p(B)\phi_{ps}(B^c)(1 - B)^d(1 - B^c)^{dp} z_t = \theta_q(B)\theta_{qs}(B^c)a_t, \tag{IV.14}$$

onde os novos parâmetros ϕ_{ps} e θ_{qs} são os operadores *sazonais* auto-regressivo e da média móvel, respectivamente.

IV.5 Estimativa

Depois de ter sido identificado o modelo, é necessário que se estime os parâmetros. A obtenção desses parâmetros pode ser realizada por meio de diferentes métodos [22], entre estes estão a estimativa não linear e os mínimos quadrados, que são caracterizados pela minimização da função

$$S(\hat{\phi}, \hat{\theta}) = \sum(z_t - \hat{z_t^2}) \tag{IV.15}$$

onde $\hat{\phi}$ e $\hat{\theta}$ representam os parâmetros dos modelos *auto-regressivo* e *média móvel* respectivamente, e $\hat{z}_t$ é a previsão obtida a partir do modelo com os respectivos parâmetros. Esta estimativa pode ser também obtida através de tabelas e gráficos [18], para os casos em que a ordem dos operadores *média móvel* e *auto-regressivo* não ultrapassem a dois. Maiores detalhes sobre a estimativa de parâmetros pode ser vistos em [18], [22].

IV.6 Diagnóstico e ajuste

Após identificar e estimar os parâmetros do modelo, verifica-se se este realmente ajusta-se a série. Em [13] encontra-se uma revisão sobre os testes mais utilizados para a verificação da adequabilidade do modelo e parâmetros estimados.

Os testes sobre os resíduos têm sido bastante empregados com sucesso [22]. Um desses testes é o teste individual da autocorrelação aplicada à série residual, ou seja, quando o modelo é adequado, o erro $\hat{a}_t = z_t - \hat{z}_t$ será independente e distribuído aleatoriamente [18]. A análise desse fato é realizada através do estudo da autocorrelação residual, isto é, da função de autocorrelação aplicada a série formada pelos erros $\hat{a}_i, 1 \leq i \leq n$, onde n é o numero de elementos da série original.

Quando a série residual não é independente nem aleatoriamente distribuída, a função de autocorrelação residual apresenta "picos", indicando caminhos que possibilitem uma melhora no modelo. Por exemplo, apresentação de um "pico" no *p-ésimo* "lag", indica que se deveria incluir um novo parâmetro do média móvel,

da forma

$$(1 - \phi_p B^p) \tag{IV.16}$$

Incluído este novo termo, verifica-se novamente a adequabilidade do modelo, e assim por diante, até que se tenha sucesso.

Capítulo V

"Backpropagation" aplicado à previsão

Propomos neste capítulo um procedimento para a realização de previsões de séries temporais, utilizando o modelo neuronal *"Backpropagation"*.

Este procedimento fornecerá sugestões para alteração dos dados de entrada (*observações* da série), e principalmente para a topologia e o treinamento da rede neuronal.

Serão apresentados também alguns testes utilizando este procedimento para o modelo *"Backpropagation"* e comparações com a metodologia *Box-Jenkins*.

V.1 Dados

Como as séries temporais normalmente apresentam valores grandes em suas observações, elas foram modificadas com os objetivos de se evitar a saturação dos neurônios e possibilitar um aprendizado mais eficiente. A cada elemento da série original, $z_1, \cdots, z_n$, aplicou-se uma função f:R $\rightarrow [0.1, 0.9]$, obtendo uma nova série $s_1, s_2, \cdots, s_n$ onde

$$s_i = f(z_i) = (z_i * escala) + offset, \tag{V.1}$$

com

$$escala = 0.8/(max - min),$$

$$offset = 0.1 - escala * min$$

e *max* e *min* representando o maior e o menor valor da série original, respectivamente.

V.2 Topologia

Um dos grandes problemas de se trabalhar com redes neuronais é achar o número de unidades que fazam com que a rede se ajuste o melhor possível ao problema. No caso específico de previsão em séries temporais, precisamos de uma rede que seja capaz de generalizar, isto é, interpolar (ou extrapolar) com uma boa precisão. Redes muito grandes possuem uma grande capacidade de aprendizado, perdendo por sua vez o poder de generalização. Pelo princípio da parcimônia [11] devemos minimizar o número de parâmetros livres, neste caso, conexões.

Os testes foram realizados com redes de uma ou duas camadas *intermediárias*, denotadas respectivamente por $e : i_1 : s$ e $e : i_1 : i_2 : s$, que possuem

- e unidades de *entrada*, cujas ativações recebemos valores $s_{t-e}, \cdots, s_{t-2}, s_{t-1}$;
- i_1 unidades da primeira camada *intermediária*, que estão completamente conectadas as unidades da camada de *entrada*;
- i_2 unidades da segunda camada *intermediária*, que estão completamente conectadas as unidades da primeira camada *intermediária*;
- s unidades de *saída*, conectadas as unidades da primeira camada *intermediária*, ou a segunda camada *intermediária*, se esta existir.

Neste trabalho preferimos considerar apenas uma unidade de *saída*, $s = 1$, mesmo nos casos em que as previsões serão realizadas para k *passos à frente*. Em cada teste será detalhado o procedimento a se tomar para a realização das previsões.

Seguindo o princípio da parcimônia, utilizaremos redes neuronais com camadas *intermediárias* pequenas, para ser mais exato, $i_1 = 2$ e $i_2 = 1$, evitando assim um número muito grande de conexões para serem ajustadas.

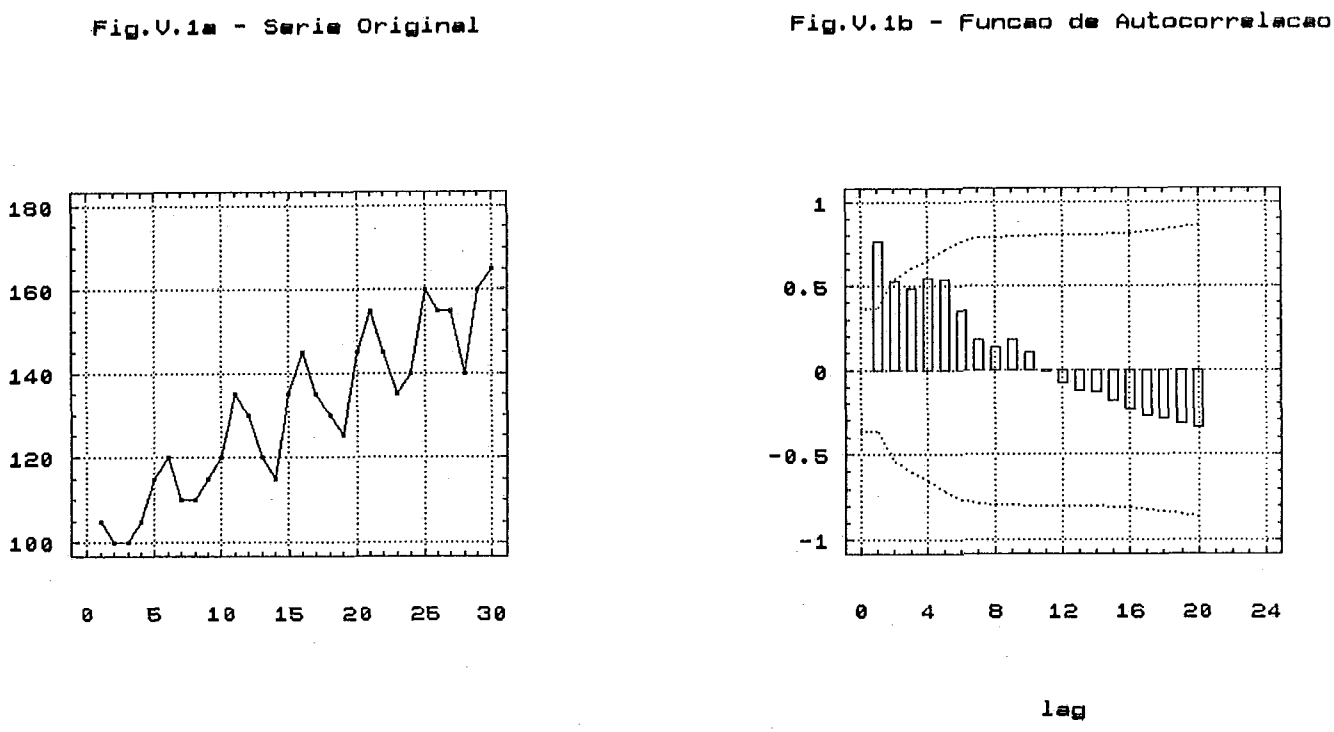

Figura V.1: Série original com a sua função de Autocorrelação

O problema das unidades de *entrada* é um pouco mais delicado, pois se diminuirmos muito o número de unidades, estaremos descartando dados, informações que podem ser relevantes para a previsão. Por isso, adotaremos uma análise gráfica , baseada na metodologia *Box-Jenkins*, tentando assim obter informações para o número de unidades de entrada.

Essa análise será realizada com base nas funções de autocorrelação e autocorrelação parcial, apresentadas no capítulo anterior.

O primeiro passo será verificar se a série original, $z_1, z_2, \cdots, z_n$, necessita da aplicação da "diferença regular" ou da "diferença sazonal de comprimento c" e em que grau, para se tornar uma série estacionária. Para isso, contrói-se os gráficos das funções de autocorrelação e autocorrelação parcial, não precisando conter mais do que $\frac{n}{4}$ e $\frac{n}{6}$ [18] pontos respectivamente, onde n é o número de observações da série original.

Normalmente, quando a função de autocorrelação e autocorrelação parcial apresentam um padrão de decaimento rápido e valores absolutos pequenos para "lags" grandes, a série não precisa sofrer nenhuma modificação. Caso contrário,

aplica-se algumas combinações da "diferença regular" com a "diferença sazonal de comprimento c"até que se consiga ou se aproxime desse padrão.

A fig.V.1 apresenta uma série com seu respectivo gráfico da função de autocorrelação. Verificamos que a fig.V.1(a) indica uma série não estacionária, que é confirmada pela função de autocorrelação (fig.V.1(b)), pois esta apresenta valores absolutos crescentes com o incremento dos "lags".

Aplicando algumas combinações das diferenças, como apresentado na fig.V.2, verificamos que as novas séries (figs. V.2.1(a), V.2.2(a) e V.2.3(a)) apresentam características de séries estacionárias. Para confirmar que realmente se transformaram em séries estacionárias, analisaremos os respectivos gráficos das funções de autocorrelação e autocorrelação parcial. Note que realmente os gráficos das figs.V.2.1(b) - 2.1(c) e V.2.2(b) - 2.2(c) apresentam um decaimento, obtendo valores pequenos com o incremento dos"lags", enquanto que os gráficos das figs.V.2.3(b) - 2.3(c) possuem ocasionais "picos"(valores grandes) com o incremento dos "lags", indicando que não foi aplicada uma "diferença" apropriada.

Tendo obtido os graus d e dp respectivamente da "diferença regular" e da "diferença sazonal de comprimento c", na qual se obteve uma nova série $r_1, r_2, \cdots, r_m$ com características estacionária, passamos a analisar as relações entre uma observação corrente, r_t, com as observações anteriores, também através das funções de autocorrelação e autocorrelação parcial. Dividiremos esta análise em dois casos:

1. Quando a função de autocorrelação apresenta um decaimento exponencial, oscilante ou senoidal como na fig.V.3, é necessário uma análise da função de autocorrelação parcial. A existência de grandes "picos" revelam a relevância das relações entre as observações. Por exemplo, a presença de "picos"nos "lags" 2 e 5 revelam uma grande correlação de r_{t-2} e r_{t-5} com r_t. Por isso destacaremos o último "lag" (de ordem p) que destoe dos outros por possuir um maior valor absoluto. Como exemplo, na fig.V.4 destacariamos o "lag" 2, tornando $p = 2$..

   Obtidos os valores *c, d, dp e p* teremos N_1 como candidato para o número de

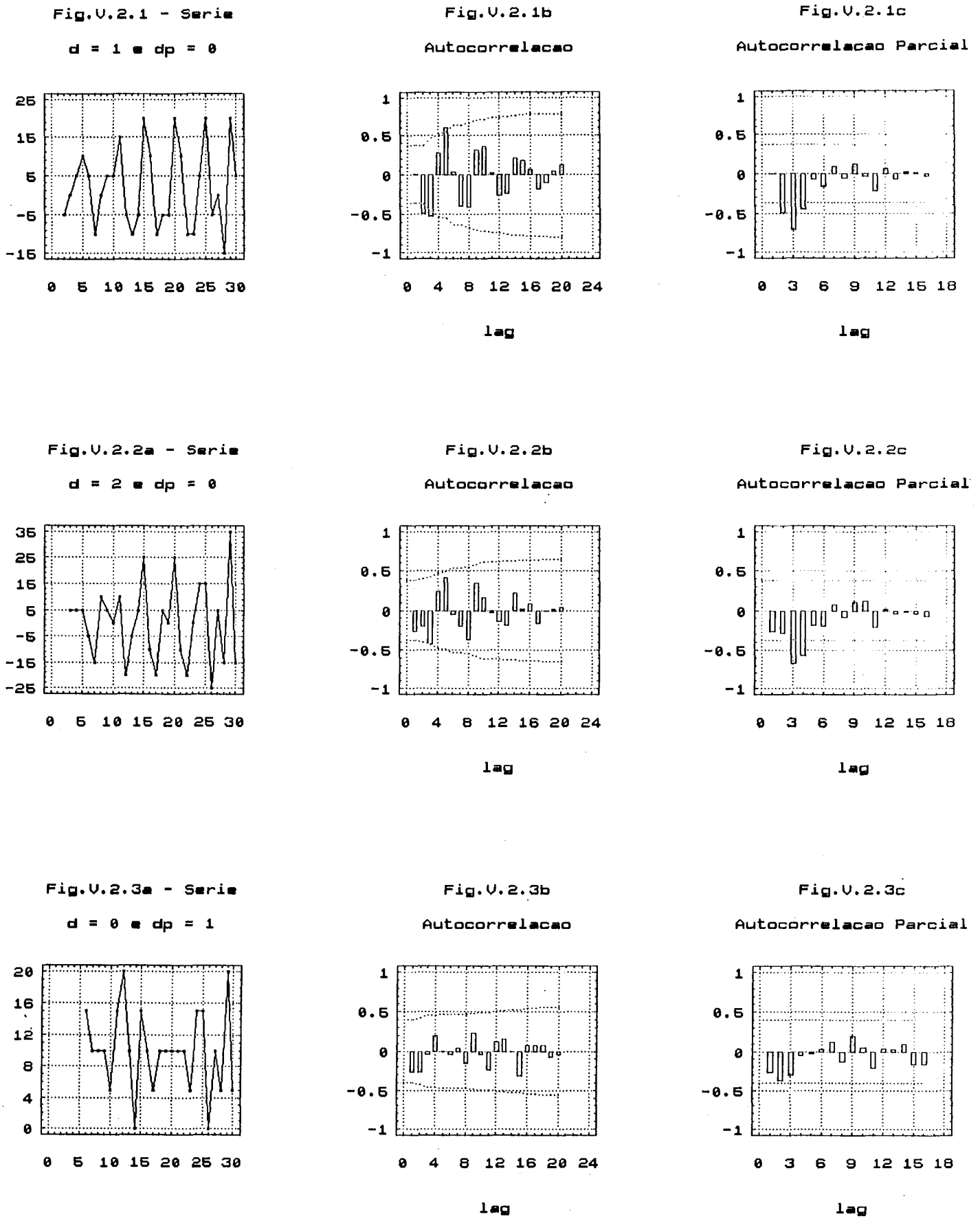

Figura V.2: Funções de Autocorrelação e Autocorrelação Parcial

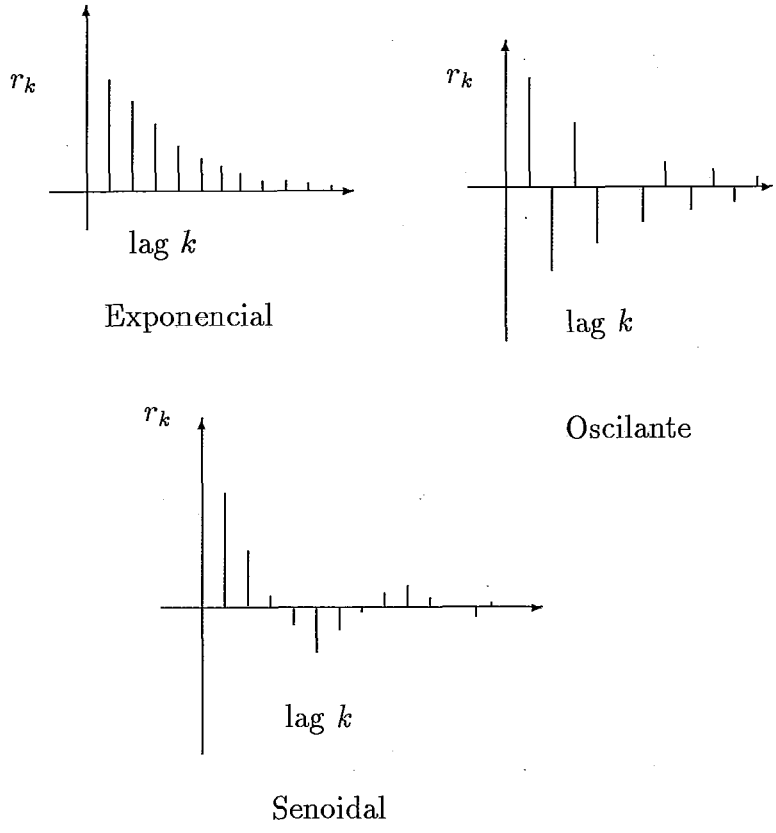

Figura V.3: Funções Autocorrelação

unidades de entrada, onde

$$N_1 = d + c * dp + p. \tag{V.2}$$

2. Quando a função de autocorrelação possue grandes "picos", ou seja, "lags" que destoem por terem valores absolutos muito maiores que os outros, destacaremos o último desses "lags", o *q-ésimo*. Outro candidato portanto, seguindo a idéia da metodologia Box-Jenkins, será N_2

$$N_2 = máximo\{d + s * dp, q\}. \tag{V.3}$$

Tendo N_1, N_2 e eventualmente o comprimento da periodicidade da série, c, o número N de unidades de *entrada* será

$$N = máximo\{N_i, c\}, i = 1, 2. \tag{V.4}$$

V.2.1 Justificativa

Como dito anteriormente, as funções de autocorrelação e autocorrelação parcial nos fornecem a força das relações entre a observação corrente e as anteriores, por isso

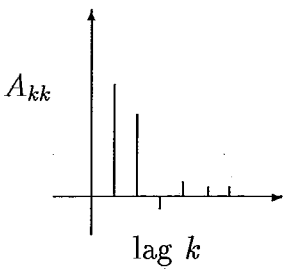

Figura V.4: Função de Autocorrelação Parcial

tentamos destacar um conjunto que contenha as observações mais relevantes.

A análise é baseada no modelo *ARIMA*, sendo que o *primeiro caso*, para a obtenção do N_1, está diretamente relacionado com a ordem do operador *auto-regressivo*, enquanto que o *segundo* (N_2) com o *"média móvel"*.

Por exemplo, tomemos a série $z_1, z_2, \cdots, z_n$. Seja $d = 1$ o grau da "difereça regular" aplicada à série, formando a nova série $r_1, r_2, \cdots, r_{n-1}$. Suponha que os gráficos das funções de autocorrelação e autocorrelação parcial sejam os da fig.V.5.

Através da análise gráfica verificamos que r_t sofre uma maior influência de r_{t-1} e r_{t-2}, tornando $N_1 = 3$,pois $p = 2$ *e* $d = 1$. Pela aplicação da "diferença regular" $r_{t-1} = z_{t-2} - z_{t-1}$ e $r_{t-2} = z_{t-3} - z_{t-2}$, logo z_{t-1}, z_{t-2} e z_{t-3} seriam as observações mais relevantes para z_t. Então, o conjunto de dados de entrada seria formado por z_{t-1}, z_{t-2} e z_{t-3}, deixando a cargo da própria rede descobrir a importância de cada observação. Quando o comprimento do período (c) é maior que N_1 ou N_2 preferimos trabalhar com c unidades de entrada, pois acreditamos que z_t e z_{t-c} possuem uma forte relação, já que estão numa mesma posição em relação ao período.

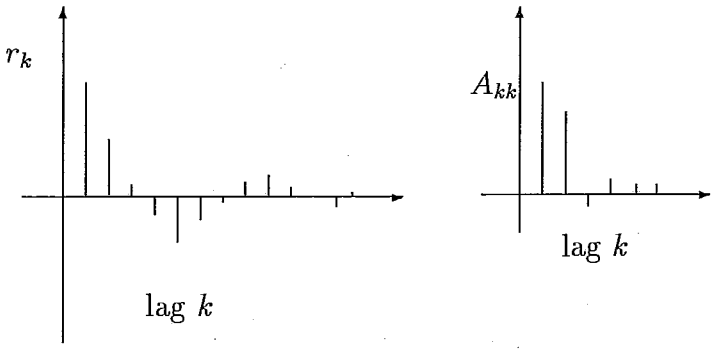

Figura V.5: Funções de Autocorrelação e Autocorrelação Parcial

V.3 Treinamento

Após fixados a topologia e os dados, inicia-se uma nova fase, o treinamento. Nesta fase os pesos das conexões são ajustados de acordo com o conhecimento adquirido através das apresentações dos dados. Os pesos iniciais dessas conexões são aleatoriamente escolhidos no intervalo [-0.2,+0.2].

O critério de parada do treinamento é um fator importante no problema de generalização. Uma rede treinada com poucas iterações pode não ser capaz de obter "conhecimento" suficiente, a partir dos dados apresentados, para a realização de previsões. Por outro lado, uma rede treinada exaustivamente está sujeita a absorver informações não desejadas (ruídos). Como não se tem conhecimento de procedimentos efetivos para se obter um critério de parada ideal, e de forma a tornar o procedimento de previsão o mais sistemático possível, foram fixados os números de apresentações do conjunto de dados (iterações) em 10.000, e as constantes de aprendizado e momento 0.1 e 0.1, com o objetivo de se fazer um ajuste bem suave. Logo as atuaizações dos pesos será de acordo com a *regra delta generalizada*:

$$\Delta W_{ij}(n+1) = 0.1.\delta_{li} o_{li} + 0.1.\Delta W_{ij}(n).$$

Em relação ao treinamento propriamente dito, foram realizados quatro

procedimentos diferentes:

1. **Normal.** Os dados (padrões) são apresentados à rede em ordem cronológica.
2. **Grupo** n. Os pesos das conexões são ajustados a cada conjunto de n padrões apresentado à rede.
3. **Aleatório.** Os padrões são apresentados à rede em ordem aleatória.
4. **Erro** $e\%$. Erros menores do que $e\%$ para as unidades de saída são desprezados, ou seja, quando o erro da unidade de saída for menor que $e\%$ não haverá ajuste nos pesos, assumindo o erro igual a zero.

V.4 Testes

Esses testes tem como objetivo verificar a aplicabilidade do modelo *"Backpropagation"* à previsão. Como medida de comparação utilizaremos a metodogia desenvolvida por Box e Jenkins. Como o treinamento no modelo *"Backpropagation"* não é deterministico, pois os pesos são inicializados aleatoriamente, realizamos cinco vezes cada teste, podendo assim realizar comparações mais reais.

Foram realizados quatro comparações entre "Backpropagation" e Box-Jenkins. Como a obtenção de um modelo na metodologia Box-Jenkins é dependente de uma análise técnica e as estimativas dos parâmetros variam de acordo com o software utilizado, três desses resultados foram obtidos da monografia *"An introduction to short Term Forecasting Using Box-Jenkins Methodology"* [20]. A outra comparação foi baseada no modelo identificado por Box e Jenkins em *Time Series Analysis - forecasting and control* [18], com a estimativa dos parâmetros realizada através do software Statgraphics [25]. Optamos por utilizar testes da metodolgia Box-Jenkins já prontos, ao invés de realizá-los, por acreditar na importância da experiência na análise e modelagem de uma série por essa metodologia.

O teste retirado do livro *Time Series Analysis - forecasting and control* [18], foi escolhido por ter sido o modelo identificado pelos autores George E. P.

Box e Gwilym M. Jenkins, que foram os responsáveis pelo desenvolvimento dessa metodologia.

Os outros testes foram retirados de uma monografia, que tem como objetivo divulgar e promover a metodologia Box-Jenkins como ferramenta para a realização de previsões em séries temporais, por isso, acreditamos que a análise da série temporal tenha sido rigorosa e os procedimentos utilizados para a estimativa dos parâmetros sejam eficientes, obtendo-se os melhores resultados possíveis.

V.4.1 Consumo de energia de Ohio

Os dois primeiros testes foram realizados com o *consumo de energia de Ohio* de 1955 a 1970 (apêndice, *série B*). Essa série foi dividida em duaspartes. A fig.V.6 mostra a primeira parte, 1955 a 1968, utilizada para o desenvolvimento do modelo. Note que a série apresenta uma tendência crescente, além de um padrão sazonal de doze meses de comprimento. A segunda parte, 1969 e 1970, será usada para comparar com as previsões obtida pelo modelo desenvolvido.

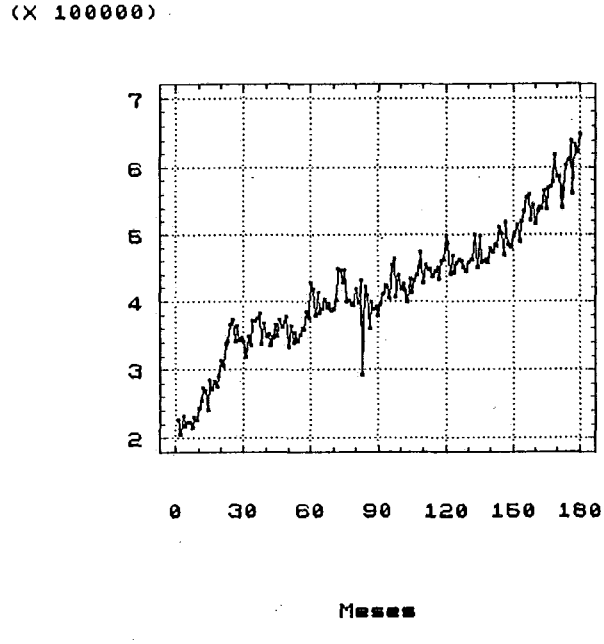

Figura V.6: Consumo de Energia de Ohio

O primeiro passo se resume em identificar o número mais apropriado

Fig.V.7a - d = 0 e dp = 0

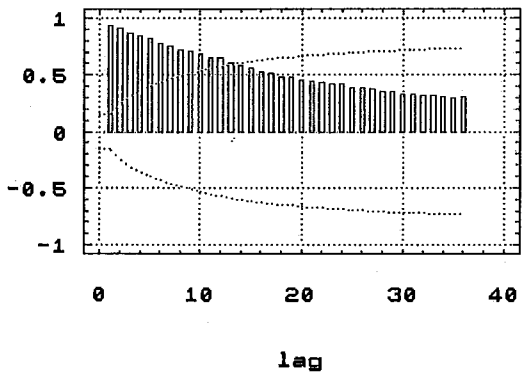

Fig.V.7b - d = 1 e dp = 0

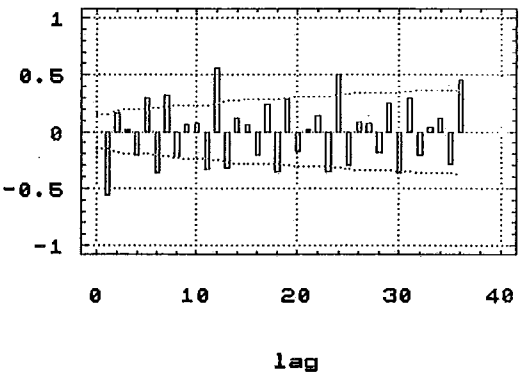

Fig.V.7c - d = 0 e dp = 1

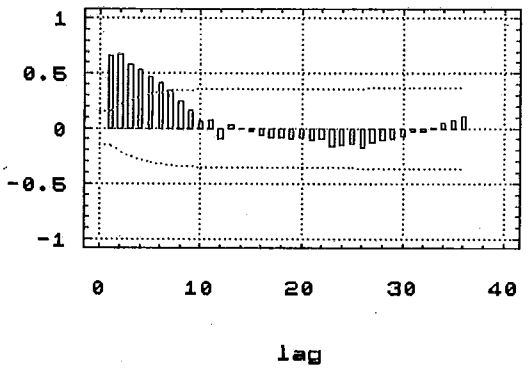

Fig.V.7d - d = 1 e dp = 1

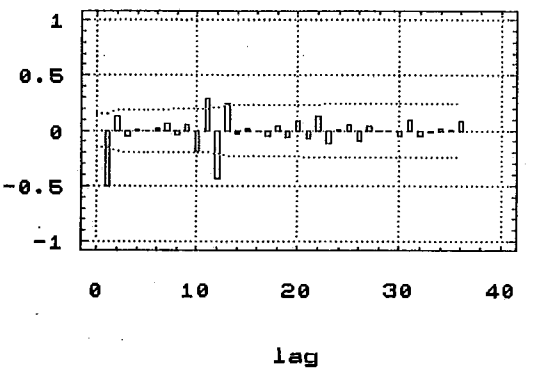

Figura V.7: Funções de Autocorrelação da Série de Energia de Ohio

de *unidades de entrada.* Através do exame das funções de autocorrelação e autocorrelação parcial aplicadas a algumas combinações da "diferença regular" com "diferença sazonal" da série original (figs.V.7 e V.8), notamos que não há um padrão de **decaimento forte** e mantêm-se valores altos com o incremento dos "lags" nos gráficos das funções de autocorrelação apresentados nas figs.V.7(a) e V.7(b).

O gráfico da fig.V.7(c) ilustra a função de autocorrelação para a "diferença sazonal" de grau 1, apresentando um decaimento senoidal, sendo portanto, necessário a análise da função de autocorrelação parcial. Analisando o gráfico fig.V.8(c) verificamos que a função de autocorrelação parcial apresenta valores grandes para os "lags" 1 e 2, logo, $p = 2$. Tornando

$$N_1 = d + dp * c + p = 0 + 1 * 12 + 2 = 14 \tag{V.5}$$

um candidato ao número de unidades de entrada.

Como $N = máximo\{N_1, c\} = 14$, a rede terá 14 unidades de entrada.

Voltando a análise gráfica, a função de autocorrelação para a "diferença regular" e a "diferença sazonal" de graus 1 (fig.V.7(d)) nos indica o *segundo caso*, N_2. Como o gráfico apresenta picos nos "lags" 1 e 12, teremos $q = 12$. Logo o número de unidades indicado será

$$N = máximo\{N_2, c\} = máximo\{d + dp * c, c\} = 13 \tag{V.6}$$

Portanto, através da análise gráfica chegamos a dois candidatos para o número de unidades de entrada, 14 e 13.

Com o objetivo de verificar a eficácia desta análise, treinamos três redes, 14:2:1, 13:2:1 e 12:2:1 com os dados até dezembro de 1968. Esse treinamento foi executado de modo aleatório, ou seja, a ordem dos dados de entrada são fornecidos aleatoriamente.

Depois de fixadas as topologias das redes, as simulações foram realizadas usando os **dados reais**, a cada simulação da rede foi realizado a previsão para um passo-à-frente, utilizando sempre como dados de entrada, as observações da série.

Fig.V.8a - d = 0 e dp = 0

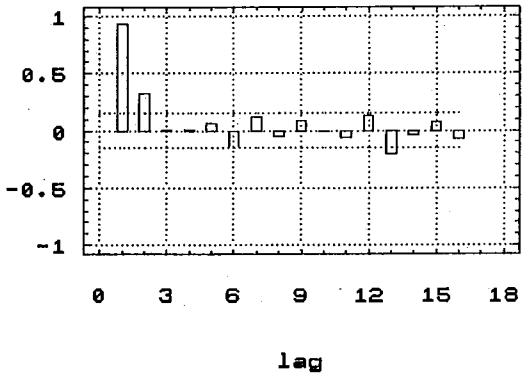

Fig.V.8b - d = 1 e dp = 0

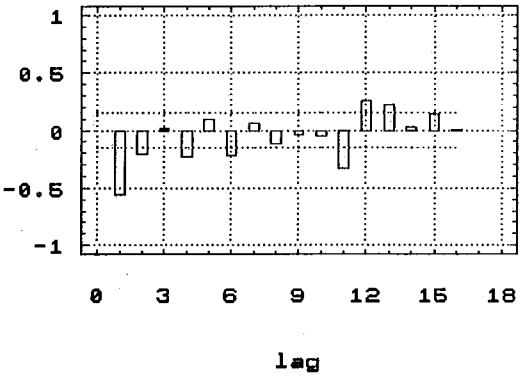

Fig.V.8c - d = 0 e dp =1

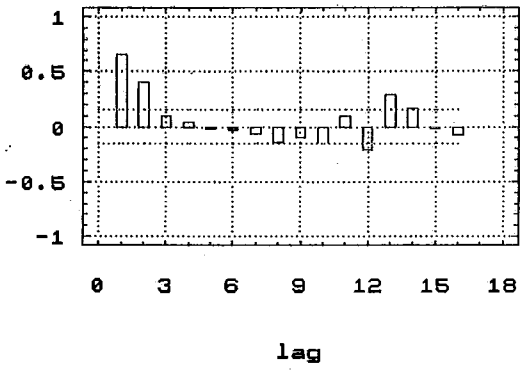

Fig.V.8d - d = 1 e dp = 1

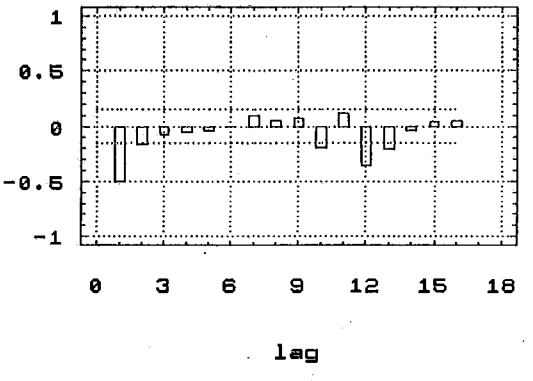

Figura V.8: Funções de Autocorrelação Parcial da Série de Consumo de Energia de Ohio

Na tabela V.1 podemos constatar que as redes 14:2:1 e 13:2:1 tiveram desempenhos equivalentes, superando e bem a rede 12:2:1 e mostrando portanto, que a análise gráfica foi relevante. Note também que as duas primeiras redes tiveram resultados melhores do que o obtido pelo modelo

$$z_t = z_{t-1} + z_{t-12} - z_{t-13} - 0.297a_{t-1} - 0.0704a_{t-12} + 0.209a_{t-13} \qquad \text{(V.7)}$$

proposto por Vicent A. [20], através da metodolgia Box-Jenkins.

Tabela V.1:

Modelo	*Erro de Previsão*			*Treinamento*	
	Média	Menor	Maior	Erro	Método
Box-Jenkins	3.23 %	-	-	-	-
14:2:1	3.16 %	3.05 %	3.25 %	1.95 E-2	Aleatório
13:2:1	3.15 %	3.07 %	3.31 %	2.04 E-2	Aleatório
12:2:1	3.48 %	3.41 %	3.60 %	2.20 E-2	Aleatório

Antes de voltarmos a comparar os resultados obtidos através do modelo "Backpropagation" com a metologia Box-Jenkins faremos algumas comparações entre diferentes procedimentos e topologias utilizadas nas redes.

As outras modificações sobre o algoritmo do *"Backpropagation"*, propostas na seção V.4, são aqui testadas e apresentadas na tabela V.2.

Tabela V.2:

Modelo	*Erro de Previsão*			*Treinamento*	
	Média	Menor	Maior	Erro	Método
Box-Jenkins	3.23 %	-	-	-	-
13:2:1	3.15 %	3.07 %	3.31 %	2.04 E-2	Aleatório
13:2:1	3.24 %	3.23 %	3.28 %	2.01 E-2	Normal
13:2:1	3.19 %	3.14 %	3.28 %	2.02 E-2	Grupo 10
13:2:1	3.25 %	3.19 %	3.29 %		Erro 2 %
13:2:1	3.21 %	3.09 %	3.30 %		Erro 2% e Aleat.
13:2:1	3.26 %	3.20 %	3.34 %	2.00 E-2	Grupo 10 e Aleat.

Notamos que o treinamento randômico se sobresaiu aos outros.Por esse motivo, utilizou-se esse procedimento em todos os testes realizados a seguir.

Foram também realizados testes com relação ao número de camadas *intermediárias*. A tabela V.3 fornece os resutados das redes 13:2:1:1 e 13:2:1.

Tabela V.3:

Modelo	*Erro de Previsão*			*Treinamento*	
	Média	Menor	Maior	Erro	Método
Box-Jenkins	3.23 %	-	-	-	-
13:2:1	3.15 %	3.07 %	3.31 %	2.0 E-2	Aleatório
13:2:1:1	3.19 %	3.11 %	3.31 %	2.0 E-2	Aleatório

As redes de uma camada *intermediária*, utilizando o treinamento aleatório, são as que mais se adaptaram a esse problema de previsão, confirmando nosso intuito de utilizar este tipo de modelo daqui para frente.

Um outro teste realizado a partir da mesma série, é o da simulação utilizando a própria **previsão como dado de entrada**, ao invés dos **dados reais**. A tabela V.4 novamente nos confirma uma melhor adaptação da rede à série em-comparação ao modelo desenvolvido pela metodologia Box-Jenkins.

Tabela V.4:

Ano	*13 : 2 : 1*			*Box-Jenkins*
	Menor	Maior	Média	
1969	3.18 %	3.38 %	3.28 %	3.2 %
1970	3.72 %	4.20 %	4.00 %	4.18 %
Média	3.42 %	3.75 %	3.60 %	3.65 %

Note que a performance da rede em relação ao modelo Box-Jenkins melhora com o aumento do horizonte da previsão. Ou seja, a rede se adaptou a previsões a longo prazo, absorvendo melhor os erros de previsões anteriores do que o modelo proposto pela metodologia *Box-Jenkins*.

V.4.2 Vendas de equipamentos

A série apresentada na fig.V.9 representa as vendas realizadas de 1961 a 1972 de um equipamento elétrico. Ela foi dividida em *períodos de comprimento 13*, correspon-

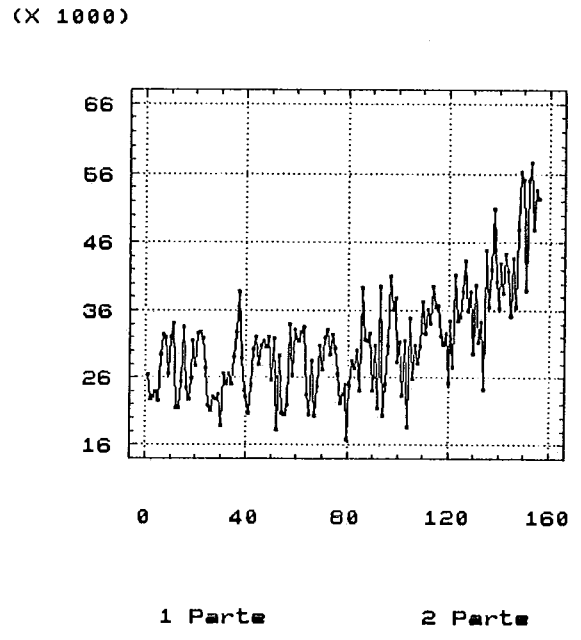

Figura V.9: Vendas de Equipamento Elétrico

dendo a um ano (apêndice A, série A).

Esta nova série apresenta um grau de dificuldade maior para a previsão do que a anterior, segundo Vicent A. [20], pois representa os dados de venda de um único produto com tendêcia crescente e periodicidade decrescente.

Como no exemplo de consumo de energia de Ohio, dividimos esta série em duas partes. A primeira, de 1961 a 1967 será utilizada para desenvolvimento do modelo, enquanto que a segunda, 1968 a 1972, será usada como comparação para as previsões obtidas pelo modelo desenvolvido.

Analisando a fig.V.9 verificamos que no período de 1961 a 1969 a série apresenta um comportameto estacionário, e a seguir uma tendência de alta. Além disso, o comportamento sazonal tende a desaparecer a partir de 1968.

A fig.V.10 nos fornece as funções de autocorrelação para algumas combinações da "diferença regular" com a "diferença sazonal de comprimento 13". O gráfico da fig.V.10a (d=0 e dp=0) apresenta um padrão semelhante ao senoidal. Analisando a função de autocorrelação parcial apresentada na fig.V.11a verificamos

Fig.V.10a - d = 0 e dp = 0

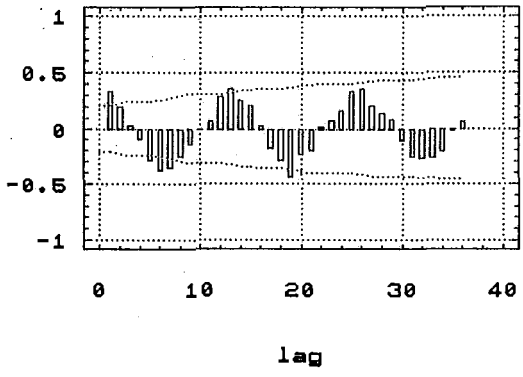

Fig.V.10b - d = 1 e dp =0

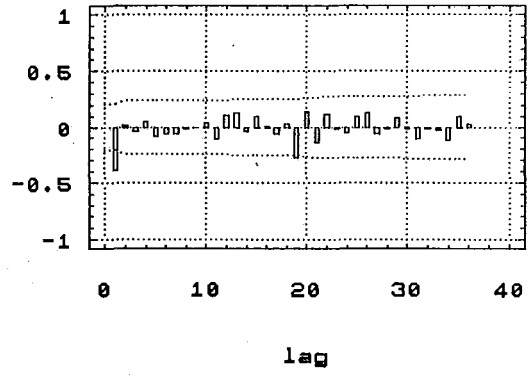

Fig.V.10c - d = 0 e dp =1

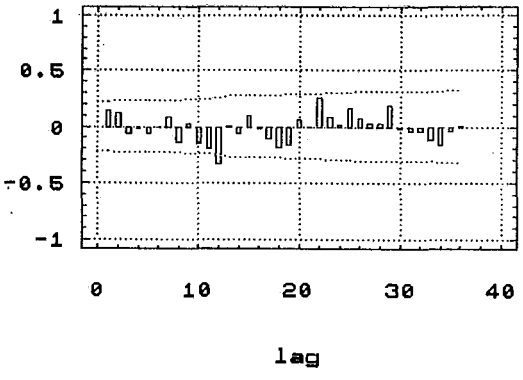

Fig.V.10d - d = 1 e dp = 1

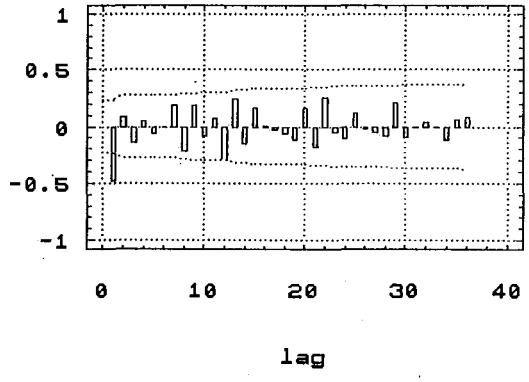

Figura V.10: Funções de Autocorrelação da Série de Vendas de Equipamento Elétrico

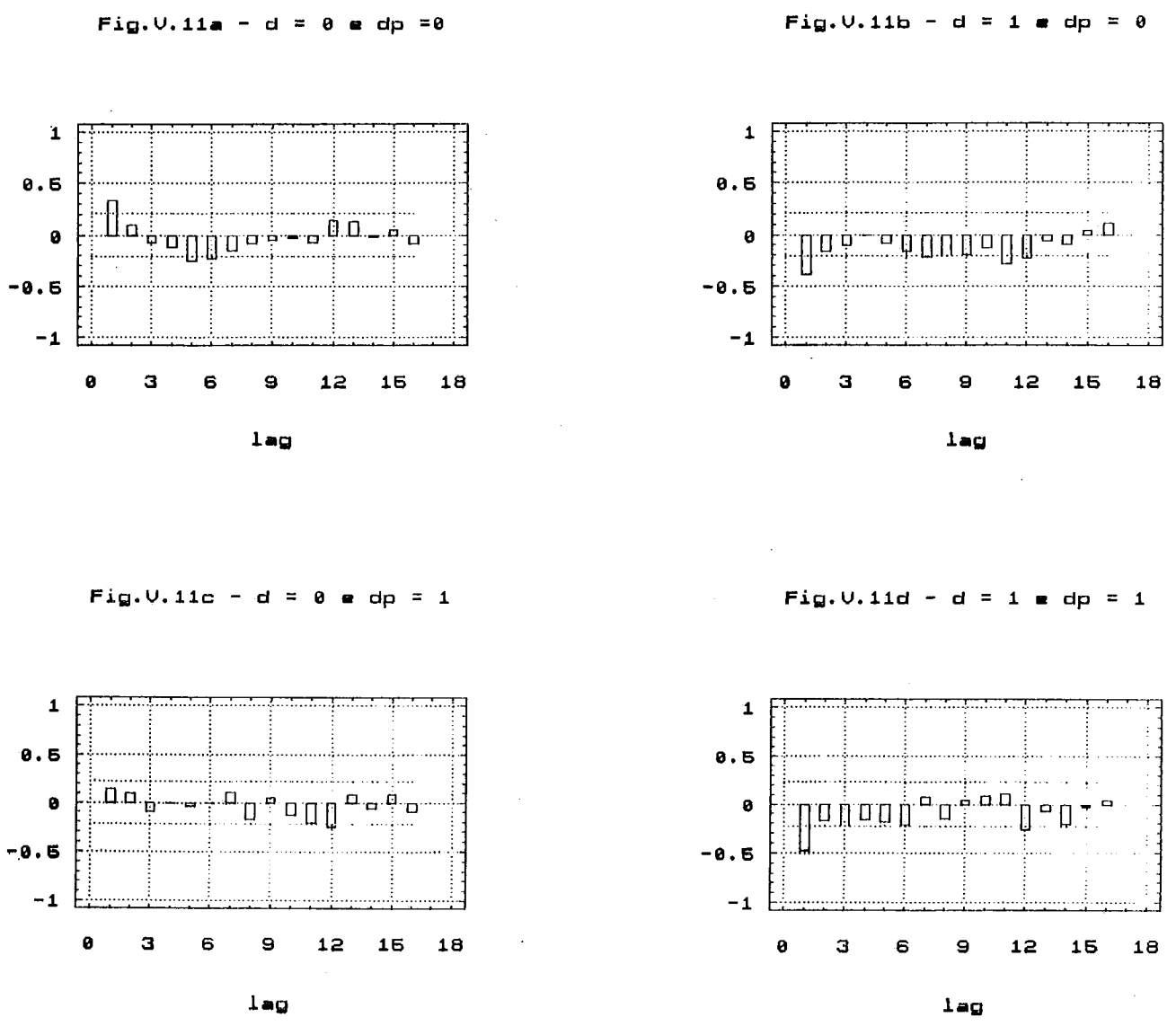

Figura V.11: Funções de Autocorrelação Parcial da Série de Vendas de Equipamento Elétrico

que o "lag" 1 é um pouco maior que os outros. Por esse motivo tomaremos $p = 1$. Um candidato ao número de *unidades de entrada* será $N_1 = 0 + 0 * 13 + 1 = 1$. Mas como $N = máximo\{N_1, c\}$, o número de unidades de entrada será 13.

Na tabela V.5 comparamos os resultados obtidos utilizando a rede $13 : 2 : 1$ com duas aplicações do modelo

$$(1 - \phi_1 B - \phi_2 B^2 - \phi_{12} B^{12})(Z_t - \mu) = a_t \qquad \text{(V.8)}$$

desenvolvido por *Vicent, A.* [20]. A primeira aplicação foi realizada com a série original, ou seja, os parâmetros do modelo proposto foram estimados utilizando os

dados da série original, enquanto que a segunda aplicação ("logged") foi realizada utilizando uma nova série obtida a partir da aplicação do logaritmo neperiano a cada elemento da série original.

Foram realizados cinco anos de previsões, de 1968 a 1972. Primeiramente foram utilizados os dados de 1961 até 1967 para se treinar a rede e estimar os parâmetros do modelo. Realizadas as previsões para 1968, os parâmetros são novamente estimados e é realizado um novo treinamento para a rede, utilizando os dados de 1961 à 1968. Este procedimento é repetido para todos os outros anos.

Tabela V.5:

Ano	*13 : 2 : 1*			*Box-Jenkins*	
	Menor	Maior	Média	Original	"Logged"
1968	15.2 %	15.9 %	15.44 %	16.6 %	15.4 %
1969	13.1 %	13.7 %	13.38 %	11.5 %	10.5 %
1970	10.2 %	10.6 %	10.54 %	9.9 %	11.1 %
1971	11.8 %	12.3 %	12.0 %	14.1 %	11.7 %
1972	12.0 %	14.8 %	13.8 %	20.0 %	16.6 %
Média	12.72 %	13.28 %	13.03 %	14.42 %	13.06 %

Podemos verificar que mesmo o pior resultado da rede neuronal foi superior ao modelo, desenvolvido pela metodologia Box-Jenkins, aplicado a série original. Em comparação a série "Logged", a rede, na média, foi um pouco superior a metodologia Box-Jenkins.

V.4.3 Vôos internacionais

A fig.V.12 apresenta os números mensais de passageiros de vôos internacionais de 1949 à 1960 (apêndice, *série C*). A série apresenta um comportamento sazonal marcante de comprimento 12, e uma tendência crescente.

A primeira parte da série, 1949 à 1957, foi utilizada para o treinamento da rede e o ajuste do modelo *ARIMA*, realizado por Box e Jenkins [18]. Os dados de 1958 à 1960 foram usados para comparações com as previsões obtidas através dos modelos propostos.

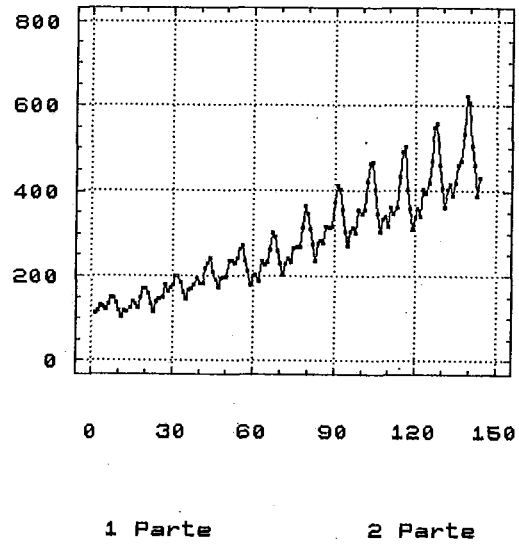

Figura V.12: Dados sobre Vôos Internacionais

A fig.V.13 nos mostra que a combinação que mais se aproxima de um comportamento estacionário é o da "diferença sazonal" de primeira ordem, *d=0* e *dp=1*. O gráfico da fig.V.13c apresenta um decaimento senoidal, tornando necessária uma análise da função de autocorrelação parcial, apresentada na fig.V.14.Para *d=0* e *dp=1* vemos que a função de autocorrelação parcial apresenta um "pico" no "lag" 1, destoando dos demais. Daí$p = 1$. Um candidato para o número de unidades de entrada é então $N_1 = 0 + 1 * 12 + 1 = 13$, tornando $N = 13$, pois o comprimento sazonal é $c = 12$. A rede, portanto, será $13 : 2 : 1$.

Os resultados da rede $13 : 2 : 1$ são comparados com o modelo

$$(1 - B)(1 - B^{12})z_t = (1 - \theta_1 B)(1 - \theta_{12} B^{12}) \tag{V.9}$$

identificado por Box-Jenkins [18], onde os parâmetros foram estimados através do software "Statgraphics" [25], obtendo-se $\theta_1 = 0.35$ e $\theta_{12} = 0.61$.

As previsões foram realizadas para os 36 meses dos anos de 1958, 1959 e 1960, utilizando-se sempre as próprias **previsões como dados de entrada.**

A tabela V.6 mostra que o desempenho da rede $13 : 2 : 1$ foi bem

Fig.V.13a - d = 0 e dp = 0

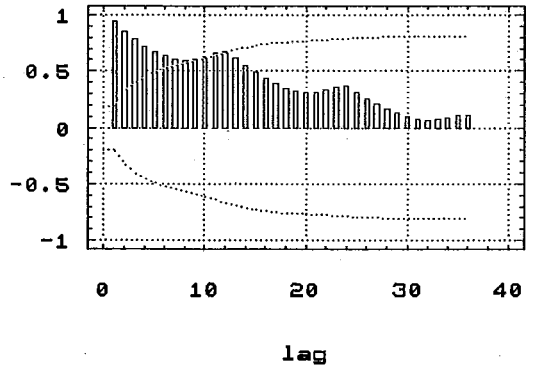

Fig.V.12b - d = 1 e dp = 0

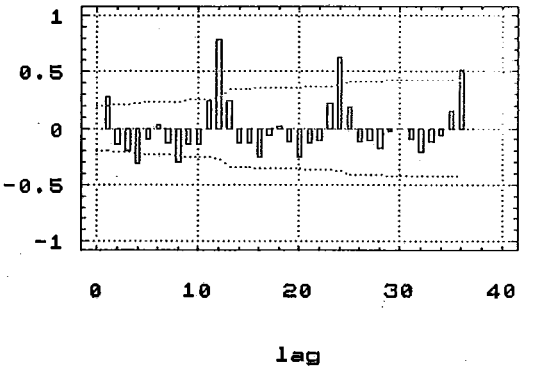

Fig.V.12c - d = 0 e dp = 1

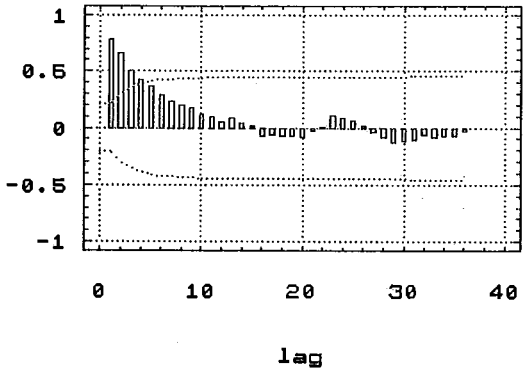

Fig.V.12d - d = 1 e dp = 1

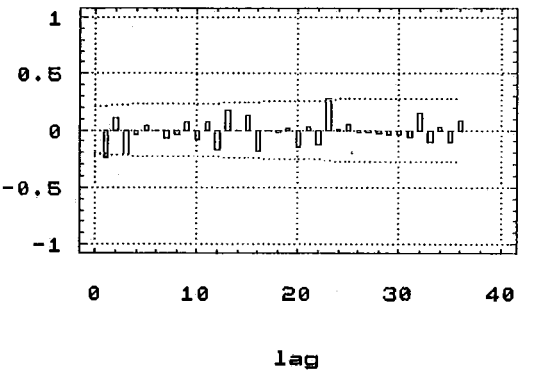

Figura V.13: Funções de Autocorrelação da Série de Vôos Internacionais

Fig.V.14a - d = 0 e dp = 0

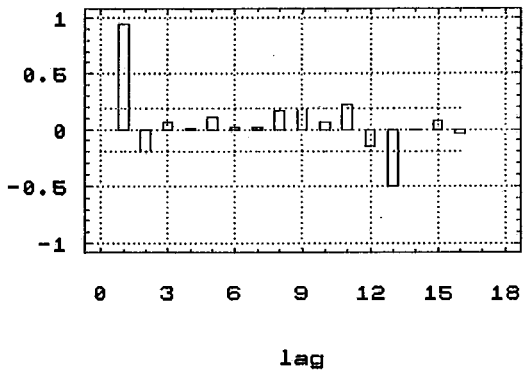

Fig.V.14b - d = 1 e dp =0

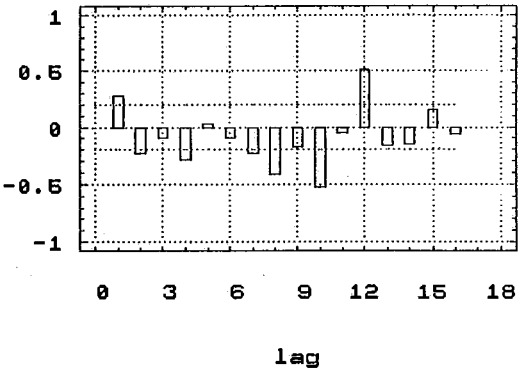

Fig.V.14c - d = 0 e dp = 1

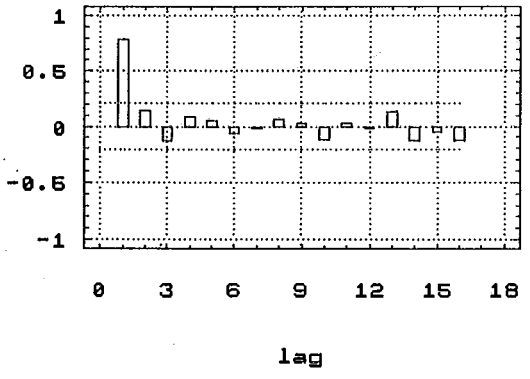

Fig.V.14d - d = 1 e dp = 1

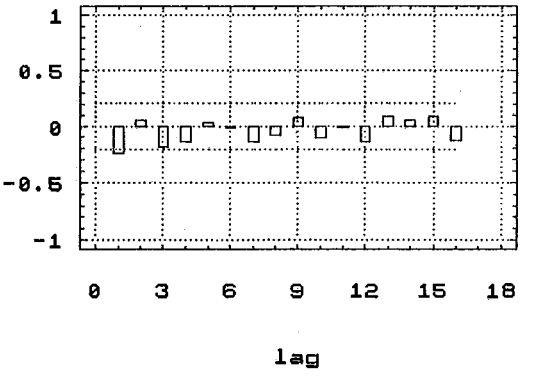

Figura V.14: Funções de Autocorrelação Parcial da Série de Vôos Internacionais

Tabela V.6:

Ano	*13 : 2 : 1*			*Box-Jenkins*
	Menor	Maior	Média	
1958	6.05 %	6.96 %	6.44 %	7.75 %
1959	4.53 %	5.38 %	4.90 %	10.86 %
1960	7.29 %	8.60 %	7.93 %	15.55 %
Média	5.96 %	6.98 %	6.42 %	11.39 %

Tabela V.7:

	Média			
Ano	*12 : 2 : 1*	*13 : 2 : 1*	*14 : 2 : 1*	*Box-Jenkins*
1958	7.63 %	6.44 %	5.00 %	7.75 %
1959	6.32 %	4.90 %	7.75 %	10.86 %
1960	7.39 %	7.93 %	11.91 %	15.55 %
Média	7.11 %	6.42 %	7.79 %	11.39 %

superior ao modelo obtido pela metodologia Box-Jenkins, absorvendo melhor os erros das previsões, obtendo assim uma previsão mais precisa a longo prazo.

Agora faremos uma comparação entre três redes com números de unidades de entrada diferentes, verificando se mais uma vez a análise gráfica realizada através do procedimento proposto foi realmente relevante ou não. A tabela 7 exibe os resultados obtidos pelas redes neuronais *12 : 2 : 1*, *13 : 2 : 1*, *14 : 2 : 1* e o do modelo obtido pela metodologia Box-Jenkins.

A rede *13 : 2 : 1* foi superior à rede *12 : 2 : 1* tanto na média geral dos erros das previsões como nas médias dos anos 1958 e 1959, só perdendo no 1960. Em realação a rede *14 : 2 : 1* houve um ganho ainda maior na média, perdendo apenas no ano de 1958. Podemos constatar que o aumento do número das unidades de entrada leva a uma maior acurácia na previsões a curto prazo, e uma perda de precisão nas de longo prazo. Como a diferença entre a rede *13 : 2 : 1* e as redes *12 : 2 : 1* e *14 : 2 : 1* foi razoável, acreditamos que a análise da série temporal,para se obter a topologia mais adequada da rede, é relevante e necessária para se realizar uma boa modelagem da série, podendo-se assim, obter previsões dos valores futuros

mais próximas da realidade.

Capítulo VI

Conclusões

Os testes mostraram que o procedimento baseado na análise gráfica da série, proposto para a obtenção do número de unidades de entrada, foi eficiente, pois tanto os testes realizados com a série de consumo de energia de Ohio como os realizados com a série de vôos internacionais, mostraram que a inclusão ou exclusão de apenas uma unidade de entrada pode modificar bastante o resultado da previsão.

Na série de consumo de energia testamos três redes diferentes quanto ao número de unidades de entrada, *12 : 2 : 1*, *13 : 2 : 1* e *14 : 2 : 1*. A análise realizada mostrou que os números de unidades de entrada mais adequados para a série em questão eram 13 e 14. Observou-se que a rede *13 : 2 : 1* teve um bom rendimento se comparado ao modelo desenvolvido pela metodologia Box-Jenkins, enquanto que a exclusão de uma unidade de entrada acarretou em uma queda na performance em relação as previsões realizadas, como observado na rede *12 : 2 : 1*. Em relação a rede *14 : 2 : 1*, obtivemos resultados equivalentes a *13 : 2 : 1*, mostrando que a análise estava correta, pois mesmo aumentando o número de unidades os resultados continuaram estáveis.

Para a série de vôos internacionais, a análise sugeriu 13 unidades de entrada para a rede neuronal. Aplicamos novamente os testes com três redes diferentes, *12 : 2 : 1*, *13 : 2 : 1* e *14 : 2 : 1*. A rede sugerida pela análise teve mais uma vez um rendimento superior, e nesse caso, bem superior ao modelo desenvolvido pela metodologia Box-Jenkins. A fim de verificar a relevância dessa análise, retiramos uma unidade da camada de entrada da rede, e constatamos que

os resultados da rede *12 : 2 : 1* pioraram sensivelmente. Com o objetivo de verificar se o fato de apenas aumentar o número de unidades acarreta em uma melhora na performance ou pelo menos mantem-se equivalente, como na série de consumo de Ohio, incluímos uma unidade de entrada à rede. Na simulação, a rede *14 : 2 : 1* obteve também resultados inferiores a rede *13 : 2 : 1* sugerida pela análise da série, mostrando mais uma vez que o procedimento para a determinação do número de unidades de entrada que mais se adapte a série foi relevante, e eficiente.

Conclui-se, a partir das simulações, que o desempenho do modelo "Backpropagation", quando avaliado para a realização de previsões, é bastante sensível à topologia da rede. Portanto, para cada série é necessário um estudo para a obtenção de uma topologia que mais se adapte a série em questão. Ou seja, cada série induz a determinação de uma topologia específica para a rede com o intuito de melhor modela-la matematicamente, tendo como principal objetivo a capacidade de generalização (interpolação ou extrapolação), para se obter previsões com uma maior acurácia possível.

A realização desses testes induzem a pensar que o modelo "Backpropagation", se bem utilizado, é uma boa ferramenta para a realização de previsões em séries temporais.

Apêndice A

Tabela .1: Série A - Vendas de Equipamento Elétrico

Período	1961	1962	1963	1964	1965	1966
1	26411	25566	23239	23196	29165	28418
2	22899	33551	22862	20829	20829	20424
3	23145	24506	23677	24895	20620	24794
4	24000	22776	18874	30195	22170	30729
5	22658	25925	26562	32095	33997	27093
6	29400	31479	25044	27836	26486	32150
7	32527	27641	26565	30684	33289	33289
8	31992	32620	25176	31414	32003	29457
9	26592	32772	29006	30577	31482	32486
10	32121	31900	32783	32047	32891	30304
11	34068	27291	38834	25599	33483	26566
12	21587	21968	28863	31813	23534	22205
13	21556	21188	24203	18254	20662	23649
Período	1967	1968	1969	1970	1971	1972
1	25019	21471	34962	30963	39746	41890
2	16736	39538	25846	32660	31358	35201
3	24860	20403	30675	25312	34316	43779
4	28431	25078	28020	34424	24315	36674
5	27313	30701	30685	27477	44774	47803
6	30065	41026	37270	41257	36533	56422
7	24068	36203	32620	34577	42131	55269
8	39488	37980	36254	35140	50904	39087
9	31702	28332	34181	38909	40289	55121
10	31483	31377	39680	43314	36485	57747
11	32629	23466	36536	36099	42923	47948
12	24207	31414	36828	38794	38697	53798
13	30826	18642	32252	29705	44237	52449

Tabela .2: Série B - Consumo de Energia de Ohio

Ano	Janeiro	Fevereiro	Março	Abril	Maio	Junho
1954	226705	206998	232202	218260	222989	223850
1955	269679	241588	285170	271828	283323	275758
1956	373499	342117	364302	343677	348255	342601
1957	382966	337531	368615	350038	353502	336454
1958	377395	334631	364840	339673	354958	342003
1959	419325	379322	413931	382444	389134	403207
1960	448025	428626	446652	399422	402203	395718
1961	409234	361229	400510	388159	392716	379395
1962	465063	407799	440558	418149	425788	417154
1963	473874	428516	455636	447072	449772	437582
1964	487303	440832	468165	442546	457999	462126
1965	500828	450747	497278	457820	460504	462007
1966	501665	468958	519319	484456	481817	497593
1967	560572	522916	545742	516294	536430	541045
1968	617747	587046	586097	540829	567503	603869
1969	650387	613735	634894	621336	632423	646633
1970	689626	616488	676767	612048	624052	651663

Ano	Julho	Agosto	Setembro	Outubro	Novembro	Dezembro
1954	215966	231139	227034	243624	255703	274755
1955	291888	313815	306543	337649	342701	366698
1956	318590	349833	336575	371337	371025	376397
1957	346738	365611	349416	373400	362110	370140
1958	351529	361280	358349	385891	376232	429165
1959	390373	396642	386770	389306	402078	448236
1960	393719	418406	398758	431902	292340	422172
1961	397140	411101	425239	422996	405604	454618
1962	400512	433831	412572	432379	439149	451799
1963	445563	451678	433395	461069	462409	497878
1964	460533	450313	445164	458087	461694	461664
1965	458618	480103	473550	484278	483624	511386
1966	503168	515626	489971	521223	535843	556692
1967	539120	566032	539487	569215	570595	579917
1968	611445	639375	562923	634066	622171	648651
1969	717611	699046	638600	642998	632139	709500
1970	698192	715772	651048	652476	-	-

Tabela .3: Série C - Número de Passageiros de Vôos Internacionais

Ano	Janeiro	Fevereiro	Março	Abril	Maio	Junho
1949	112	118	132	129	121	135
1950	115	126	141	135	125	149
1951	145	150	178	163	172	178
1952	171	180	193	181	183	218
1953	196	196	236	235	229	243
1954	204	188	235	227	234	264
1955	242	233	267	269	270	278
1956	284	277	317	313	318	306
1957	315	301	356	348	355	336
1958	340	318	362	348	363	337
1959	360	342	406	396	420	405
1960	417	391	419	461	472	535
Ano	Julho	Agosto	Setembro	Outubro	Novembro	Dezembro
1949	148	148	136	119	104	118
1950	170	170	158	133	114	140
1951	199	199	184	162	146	166
1952	230	242	209	191	172	194
1953	264	272	237	211	180	201
1954	302	293	259	229	203	229
1955	315	364	347	312	274	237
1956	374	413	405	355	306	271
1957	422	465	467	404	347	305
1958	435	491	505	404	359	310
1959	472	548	559	463	407	362
1960	622	606	508	461	390	432

Referências Bibliográficas

[1] KARNIN, Ehud, D.. A Simple procedure For Pruning Backpropagation Trained Networks. IEEE Trasactions on Neural Networks, V.1, n.2, p.239-242, Jun.1990.

[2] PINEDA, Fernando J. Generalization of Backpropagation To Recurrent Neural Networks. Physical Review Letters, v.59, n.19, p.2229 - 2232, nov. 1987.

[3] COWAN, Jack D. & SHARP. Neural Net and Artificial Intelligence.

[4] MONTGOMERYM, Johnson. Forecasting and time series analysis.

[5] SIESTMA, J, & DOWN, R. J. F. Neural Net Pruning - Why and How.

[6] TAKASHI, K. et al. Stock Maket Prediction System With Modular Neural Networks.

[7] KAMIJO, KEN-ICHI & TANIGAWA, Tetsuji. stock Price Pattern Recognition: A Recurrent Neural Networks Approach. In: International Joint Conference. Neural networks, 1990, california, v.1, p.215-221.

[8] WILSON III, James M. Backpropagation Neural Networks: A Comparison of Selected Algorithms and Methods of Improving Performance. In: Proceedings of the Workshop on Neural Networks, 2, 1991, Alabama, p.39-46.

[9] TANG; ALMEIDA, Chrys de & FISHWICK, P.A..Time Series Forecating Using Neural Networks vs. Box-Jenkins Methodology. In: Proceedings of the Workshop on Neural Networks, 1, 1990, Alabama, p.95-100.

[10] RICAHRDSON, M. et al. The Application of Neural Networks to Forecasting Agricutural Commodity Prices. In: International Symposium on Forecasting, 11, 1991, New York, p.27.

[11] TURKEY, Y. W. Discussion emphatsizing the connection between analysis of variance and espectrum analysis, Technometrics, 3, 191, 1961.

[12] MINSKY, M. & PAPERT, S. Perceptrons. Cambridge, MA: MIT Press, 1969.

[13] NEWLBOLD, P. Model checking in times series analysis. proceedings of ASA-AENSUS-NBER conference on Applied time series analysis of Economic data, 1982, Ed. A. Zellner.

[14] HEBB, D. O.. Organazation of Behavior, Science Editions, Inc., New York, 1949.

[15] McCLELLAND, J. L. & RUMELHART, D. E. Parallel distributed processing: exploration in the microstructure of cognition. MIT, Cambridge, Massachussets, 1988, v.1.

[16] McCLELLAND, J. L. & RUMELHART, D. E. Exporation in Parallel Distributed Processing. MIT, Cambridge, Massachussets, 1988.

[17] HECHT-NIELSEN, Robert. Theory of the Backpropagation Neural Network. p I.593-605.

[18] BOX, G. E. P. & JENKINS, G. M. Time series analysis: Forecasting and Control. 1970, Hoden Day.

[19] CARVALHO, L. A. V. de. et al. Redes Neuronais Artificiais: a volta do cérebro eletrônico. Ciência Hoje, v.12, n 12, p 12-21, Jan./fev. 1991.

[20] MABERT, V. A. An Introduction to Short Term Forecasting Using The Box-Jenkins Methodology. Purdue University, 1975.

[21] CARVALHO, Luiz Alfredo V. de . Redes Neuronais e a Tradição Conexionista da Inteligência Artificial. rio de Janeiro, [UFRJ], 1988.

[22] PEREIRA, B. de B. & SANT'ANA, Anibal Panacho. Análise Econométrica de séries Temporais. Rio de Janeiro, UFRJ. (Estudos e Comunicações do IM, 41).

[23] CARDADOR, D. M. Representação de Conhecimento: Modelos Classicos e Conexionistas. Rio de Janeiro, IME, 1990. (Tese de Mestrado).

[24] CARVALHO, L. A. Vidal de. Síntese de Redes Neuronais Com Aplicações à Representação do Conhecimento e a Otimização. Rio de Janeiro, COPPE/UFRJ, 1989. (Tese de Doutorado).

[25] STATGRAPHICS. Statistical Graphics System. Version 2.6. Copyright 1985, 1986, 1987 STSC, Inc and Statistical Graphics Corporation.

[26] WIDROW, B. & Hoff, M. E. Adaptative switching circuits., Institute of Radio Engineers, 1960. IRE Wescon Convention record, Part 4, 96.