

unesp

UNIVERSIDADE ESTADUAL PAULISTA
FACULDADE DE ENGENHARIA DE ILHA SOLTEIRA
PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

DISSERTAÇÃO DE MESTRADO
EM
ENGENHARIA ELÉTRICA

1210001168



Nº 58

DESENVOLVIMENTO DE UM SISTEMA PREVISOR
DE CARGAS ELÉTRICAS VIA REDES NEURAIIS
ALUNA: MARA LÚCIA MARTINS LOPES
ORIENTADOR: CARLOS ROBERTO MINUSSI

Te.1168

MARA LÚCIA MARTINS LOPES

DESENVOLVIMENTO DE UM SISTEMA PREVISOR DE CARGAS ELÉTRICAS VIA REDES NEURAIIS

Dissertação submetida à Universidade Estadual
Paulista – UNESP, Câmpus de Ilha Solteira, para a
obtenção do título de Mestre em Engenharia Elétrica.

Orientador: Prof. Dr. Carlos Roberto Minussi



UNESP - "CAMPUS DE ILHA SOLTEIRA"
SERVIÇO TÉCNICO DE BIBLIOTECA E DOCUMENTAÇÃO

DATA DE CHEGADA	DATA DE TOMBO
REGISTRADO POR	TOMBO
AQUISIÇÃO	CLASSIFICAÇÃO

31/03/01
Lélia
1168
Doacop
R\$ 10,00
autor
L864d

AR. 01/01/01 - Rec. 01/01/01

ILHA SOLTEIRA – SP, DEZEMBRO DE 2000

3040205 - sistemas eletrônicos de medida
e de controle

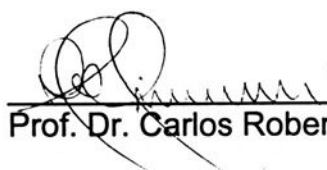
“Desenvolvimento de um Sistema Previsor de Cargas Elétricas Via Redes Neurais”

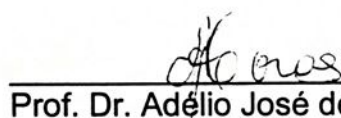
Mara Lúcia Martins Lopes

DISSERTAÇÃO SUBMETIDA À FACULDADE DE ENGENHARIA DE ILHA SOLTEIRA – UNESP – COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO TÍTULO DE MESTRE EM ENGENHARIA ELÉTRICA (ME).


Prof. Dr. Antonio Padilha Feltrin – Coordenador

COMISSÃO EXAMINADORA:


Prof. Dr. Carlos Roberto Minussi – orientador


Prof. Dr. Adélio José de Moraes


Prof. Dr. Rubén Augusto Romero Lázaro

Ilha Solteira – SP, dezembro de 2000



Resumo

É muito importante conhecer o comportamento de carga com antecedência em problemas de análise, planejamento e operação de sistemas de energia elétrica com o propósito de garantir o fornecimento aos consumidores de forma econômica, segura e de confiança ininterrupta. O objetivo deste trabalho é o desenvolvimento de uma metodologia para previsão de carga elétrica baseado em uma rede neural com treinamento realizado utilizando-se o algoritmo *Back-Propagation* com controlador nebuloso. Esta metodologia resulta em um treinamento rápido, quando comparado à formulação convencional do algoritmo *Back-Propagation*. Os resultados apresentados utilizam dados de uma companhia elétrica brasileira como forma de testar a metodologia proposta.



Abstract

Knowing the load behavior in advance is very important in planning, analysis and operation of power systems to maintain an uninterrupted reliable, secure and economic electric energy providing. The objective of this work is development of a methodology for electric load forecasting based on a neural network. Here, it is used a *Back-Propagation* algorithm with an adaptive process based on fuzzy logic. This methodology results in fast training, when compared to the conventional formulation of *Back-Propagation* algorithm. Results are presented using data from a Brazilian electric utilities and the performance is very good for the proposal objective.



Aos meus pais, Manoel e Lourdes.



Agradecimentos

Agradeço a todos aqueles que me incentivaram e apoiaram na elaboração deste trabalho, especialmente minha família: minha mãe, meu pai, meus irmãos e ao Valter pela paciência e compreensão. Agradeço aos meus colegas que compartilharam comigo as derrotas e vitórias, em especial Elizete Amorim, Giovani Faria, Marcelo Almeida (Cilim) e Marilene Mantovani pela força e apoio na realização do meu trabalho. Agradeço ao técnico do laboratório de computação Deoclécio Mitsuiti Kosaka pelas muitas vezes em que dispôs de seu tempo e me auxiliou de certa forma no desenvolvimento do meu trabalho. Agradeço a Prof^a. Anna Diva Plasencia Lotufo, pelo seu apoio e suporte, que permitiu a participação do meu trabalho em um congresso internacional importante, e ao meu orientador Prof. Carlos Roberto Minussi, que sempre esteve presente em todos os momentos, dando-me empenho para prosseguir nas etapas difíceis e sendo mais que tudo um amigo. Agradeço também a Fapesp por seu apoio financeiro, através de uma bolsa de estudos e reserva técnica, que permitiu que eu pudesse me dedicar integralmente ao desenvolvimento desta dissertação. Agradeço acima de tudo a Deus, que me concedeu força e perseverança para continuar caminhando sempre em busca de meus sonhos.



Simbologia

RNA	=	Rede Neural Artificial
B-P	=	<i>Back-Propagation</i>
RH	=	Rede Hopfield
γ	=	taxa de treinamento
$z_1, z_2, \dots, z_t$	=	valores observados das variáveis $Z_1, Z_2, \dots, Z_t$
$Z_1, Z_2, \dots, Z_t$	=	variáveis aleatórias
z_{t+l}	=	valor da série temporal no período de tempo l no futuro
$\hat{z}_t(l)$	=	previsão de z_{t+l}
μ	=	média
$e_t(l)$	=	erro de previsão
$\sigma^2[e_t(l)]$	=	variância do erro da previsão
ACF	=	Função de Autocorrelação Amostral
PACF	=	Função de Autocorrelação Parcial Amostral
r_k	=	autocorrelação amostral de atraso k
$\bar{z}$	=	média amostral da série temporal
N	=	número de observações da série temporal
ρ_k	=	autocorrelação teórica
ρ_{kk}	=	autocorrelação parcial teórica
r_{kk}	=	autocorrelação parcial amostral



AR	=	Modelo <i>Auto Regressive</i>
MA	=	Modelo <i>Moving Average</i>
ARMA	=	Modelo <i>Auto Regressive Moving Average</i>
A_t	=	impulsos aleatórios
θ_0	=	termo constante
$\phi_1, \dots, \phi_p$	=	coeficientes auto regressivos
p	=	ordem do modelo AR
q	=	ordem do modelo MA
$\theta_1, \dots, \theta_p$	=	coeficientes de média móvel
$\hat{\sigma}_z$	=	desvio padrão amostral
$e_{t-1}(1)$	=	erro das previsões a um passo adiante
$\hat{a}_t$	=	resíduo para o tempo t
$\bar{\hat{a}}_t$	=	média residual
z_t^*	=	série temporal transformada
w_t	=	série temporal diferenciada
B	=	operador que expressa a série temporal diferenciada em termos da original
d	=	número de diferenças regulares
ARI	=	Modelo <i>Auto Regressive Integrated</i>
IMA	=	Modelo <i>Integrated Moving Average</i>
ARIMA	=	Modelo <i>Auto Regressive Integrated Moving Average</i>
X	$\triangleq$	conjunto de vetores padrão
	=	$[x_1, x_2, \dots, x_n]^T$
W	$\triangleq$	pesos sinápticos associado a cada vetor padrão
	=	$[w_1, w_2, \dots, w_n]^T$



T	=	transposto
Σ	=	bloco somatório
w_0	=	peso “bias”
$x_0 = +1$	=	controle do nível de saída do neurônio
s_i	$\triangleq$	saída do neurônio
	=	$\mathbf{X}^T \mathbf{W}$
ε^2	=	erro quadrático da rede neural
ε_i	=	erro instantâneo de cada neurônio alocado na última camada
d_i	=	saída desejada do i-ésimo elemento da última camada da rede
y_i	=	saída do i-ésimo elemento da última camada da rede
ns	=	número de neurônios da última camada da rede
$\mathbf{V}_i$	$\triangleq$	vetor contendo os pesos do neurônio i
	=	$[w_{0i} \ w_{1i} \ w_{2i} \ \dots \ w_{ni}]^T$
$\nabla_i(h)$	=	gradiente do erro quadrático com relação aos pesos do neurônio i avaliado em h
h	=	índice de iteração
λ	=	constante que determina a inclinação da curva y_i
$sgm'(s_i)$	=	derivada parcial de y_i com relação a s_i
v_{ij}	=	peso correspondente à interligação entre o i-ésimo e j-ésimo neurônio
η	=	constante momento
σ_j	=	derivada da função sigmóide com relação a s_j
j	=	elemento (neurônio) da rede
$R(j)$	=	conjunto dos índices dos elementos que se encontram na fileira seguinte à fileira do elemento j e que estão interligados ao elemento j
$\mu_A(x)$	=	valor da função de pertinência do conjunto nebuloso A correspondente ao elemento x



A, B	=	conjuntos nebulosos
X	=	coleção de objetos
$\mu_{A'}(x)$	$\triangleq$	valor da função de pertinência do complemento de A correspondente ao elemento x
	=	$1 - \mu_A(x)$
C	=	$A \cap B$
$\mu_C(x)$	=	valor da função de pertinência da interseção de A e B correspondente ao elemento x
D	=	$A \cup B$
$\mu_D(x)$	=	valor da função de pertinência da união de A e B correspondente ao elemento x
MAN	=	Memória Associativa Nebulosa
δ_i	=	valor do conjunto que possui um valor de pertinência μ_i
n	=	número de regras nebulosas
$f(x)$	=	conjunto nebuloso contínuo
u	=	ação de controle
ε_g	=	erro global da rede neural
$\Delta \varepsilon_g$	=	variação do erro global da rede neural
np	=	número de vetores padrão da rede neural
$\Delta \gamma$	=	ação de controle que deve ser exercida sobre o sistema
q	=	índice que indica a iteração corrente
α	=	número positivo arbitrário;
$\Delta \psi^q$	=	acréscimo precedente do controlador nebuloso no instante q
lu	=	limite de variação de $\Delta \gamma$
NG	=	Negativo Grande



NP	=	Negativo Pequeno
ZE	=	Próximo de Zero
PMP	=	Positivo Muito Pequeno
PP	=	Positivo Pequeno
ME	=	Médio
PG	=	Positivo Grande
PMG	=	Positivo Muito Grande
m	=	dimensão do vetor X
$L(h-p)$	=	valor da carga p horas anterior à hora corrente h
$L(h+1)$	=	valor da carga elétrica correspondente à hora subsequente a hora corrente h
t	=	vetor de tempo referente aos dados históricos (mês, dia da semana, feriado, hora, etc.) representado de modo similar ao código binário $(-1,+1)$
GC	=	Ganho de Ciclo
GT	=	Ganho de Tempo
NBP	=	número de ciclos do B-P
NBPN	=	número de ciclos do B-P nebuloso
TBP	=	tempo de processamento via B-P
TBPN	=	tempo de processamento via B-P nebuloso
$L(h)$	=	valor da carga real referente à hora h
$\underline{L}(h)$	=	valor da carga estimada pela rede neural referente à hora h
N	=	número total de horas
R	=	campo dos números reais.



Índice

1	Introdução.....	01
2	Previsão de Cargas.....	04
2.1	Cargas Elétricas.....	04
2.2	Previsão para Modelos de Série Temporal Estacionária.....	07
2.2.1	Função de Autocorrelação e Autocorrelação Parcial Amostral.....	07
2.2.2	Formas Gerais dos Modelos ARMA.....	09
2.2.2.1	Modelo Ruído Branco.....	09
2.2.2.2	Modelo <i>Auto Regressive</i> (AR(p)).....	10
2.2.2.3	Modelo <i>Moving Average</i> (MA(q)).....	10
2.2.2.4	Modelo <i>Auto Regressive Moving Average</i> (ARMA(p,q)).....	10
2.3	Previsão para Série Temporal Estacionária.....	11
2.4	Previsão para Série Temporal Não-Estacionária.....	21
2.5	Classe de Modelos para uma Série Temporal Não-Estacionária.....	23
2.5.1	Modelo Walk Aleatório.....	23
2.5.2	Modelo <i>Auto Regressive Integrated Moving Average</i>	23
2.6	Conclusão.....	25
3	Redes Neurais.....	26
3.1	Modelo Neural Biológico.....	26
3.2	Modelo Neural Artificial.....	27
3.3	Estrutura da Rede Neural.....	30
3.4	Treinamento da Rede Neural.....	30
3.5	Algoritmo <i>Back-Propagation</i>	31



3.6	Algoritmo <i>Back-Propagation</i> com Momento.....	36
3.7	Vantagem do Algoritmo <i>Back-Propagation</i>	39
3.8	Desvantagem do Algoritmo <i>Back-Propagation</i>	39
3.9	Conclusão.....	40
4	Lógica Nebulosa.....	41
4.1	Conjuntos Nebulosos.....	41
4.2	Funções de Pertinência.....	44
4.3	Controle Nebuloso.....	45
4.3.1	Nebulização.....	46
4.3.2	Inferência.....	47
4.3.3	Desnebulização.....	49
4.4	Conclusão.....	50
5	Previsão de Carga via Redes Neurais com Controlador Nebuloso.....	51
5.1	Rede Neural Nebulosa.....	51
5.2	Conclusão.....	55
6	Aplicação – Previsão de Carga.....	56
6.1	Sistema Previsor de Cargas Elétricas.....	56
6.2	Outros Testes.....	66
6.3	Conclusão.....	68
7	Conclusão e Sugestões para Trabalhos Futuros.....	69
7.1	Conclusão.....	69
7.2	Sugestões para Trabalhos Futuros.....	70
	Referências Bibliográficas.....	71
	Apêndice A.....	76
	Apêndice B.....	77
	Apêndice C.....	79



Lista de Figuras

2.1	Série temporal estacionária.....	06
2.2	Série temporal não-estacionária.....	06
2.3	Estágios na aproximação de Box-Jenkins.....	12
3.1	Componentes de um neurônio biológico.....	27
3.2	Componentes de um neurônio artificial.....	28
3.3	Modelo de um neurônio artificial.....	28
3.4	Funções de ativação.....	29
3.5	Treinamento de um neurônio situado na última camada.....	37
3.6	Treinamento de um neurônio situado nas demais camadas.....	38
3.7	O algoritmo gradiente descendente no espaço de pesos.....	38
4.1	Conjuntos nebulosos.....	42
4.2	Operador AND.....	43
4.3	Operador OR.....	43
4.4	Operador NOT.....	44
4.5	Funções de pertinência.....	45
4.6	Diagrama de bloco de um sistema nebuloso.....	45
4.7	Sistema nebuloso.....	46
4.8	Conjunto de regras e MAN.....	48
5.1	Funções de pertinência para as variáveis ϵg , $\Delta \epsilon g$ e $\Delta \psi$	53
6.1	Resultado da previsão de carga para rede neural 1, aplicação 1.....	61
6.2	Resultado da previsão de carga para rede neural 2, aplicação 1.....	62
6.3	Resultado da previsão de carga para rede neural 1, aplicação 2.....	62
6.4	Resultado da previsão de carga para rede neural 2, aplicação 2.....	63



6.5 Resultado da previsão de carga para rede neural 1, aplicação 3..... 63

6.6 Resultado da previsão de carga para rede neural 2, aplicação 3..... 64



Lista de Tabelas

2.1	Definições dos seis modelos ARMA.....	11
2.2	Propriedades características dos seis modelos ARMA.....	14
2.3	Erro padrão estimado dos quadrados mínimos dos modelos ARMA.....	17
2.4	Propriedades características dos seis modelos ARIMA(p,1,q).....	25
5.1	Regras do controlador nebuloso.....	55
6.1	Especificação da rede neural referente à aplicação 1.....	60
6.2	Especificação da rede neural referente à aplicação 2.....	60
6.3	Especificação da rede neural referente à aplicação 3.....	61
6.4	Resultados comparativos para a aplicação 1.....	65
6.5	Resultados comparativos para a aplicação 2.....	65
6.6	Resultados comparativos para a aplicação 3.....	66
6.7	Especificação da rede neural.....	67
6.8	Resultados comparativos.....	67



Planejamento de expansão, fluxo de potência, operação econômica, análise e controle de segurança de Sistemas de Energia Elétrica são alguns exemplos de estudos que efetivamente dependem do conhecimento futuro do comportamento do perfil da carga do sistema (previsão de carga) ([10]), ou seja, da previsão de informações futuras de uma série temporal por extração do conhecimento de valores passados. Por conseguinte, a previsão de carga constitui-se num item muito importante de investigação.

Na literatura técnica podem-se encontrar muitos métodos utilizados para previsão de carga, dentre eles tem-se: técnica de regressão linear simples ou múltipla, alisamento exponencial, estimação de estado, filtro de Kalman e ARIMA (*AutoRegressive Integrated Moving Average*) de Box & Jenkins ([01]).

Todos estes métodos necessitam que seja efetuada uma modelagem prévia da carga para aplicação posterior. Os fatores que influenciam na modelagem da carga podem ser condições meteorológicas (nebulosidade, velocidade do vento, variações bruscas de temperatura) e fatores diversos como efeito de dias atípicos (feriados, greves, etc.). O conhecimento da influencia de tais fatores permitem representar da melhor maneira o comportamento da carga para então servir como entrada no algoritmo de previsão escolhido. Bons resultados têm sido obtidos apesar de, às vezes, serem requeridos grandes esforços computacionais, além de técnicas de modelagem complexas. Outro tipo de resultado tem sido obtido, também, através da introdução de sistemas inteligentes, que se tem mostrado eficientes, principalmente, para o caso dos dias especiais. Recentemente, o uso de redes neurais ([15], [16], [25], [33]) tem despertado a atenção de muitos pesquisadores.

As Redes Neurais Artificiais (RNA) são concepções em *hardware* e/ou *software* baseadas no mecanismo de funcionamento do cérebro. A neurocomputação surgiu, basicamente, em 1943 através da proposição de um neurônio artificial por McCulloch e Pitts. A partir deste modelo, foram desenvolvidas várias concepções de RNA, porém, os resultados



eram pouco significativos. Somente após os anos 70 é que houve considerável avanço, principalmente, com o advento do algoritmo *Back-Propagation* (B-P) ([34]) e da rede de Hopfield (RH) ([33]) e, sobretudo, pelo avanço da microeletrônica a partir dos anos 80, foi que permitiram a idealização de estruturas em *hardware* de modo eficiente.

O algoritmo B-P tem por finalidade o aprendizado supervisionado (regra baseada no binômio estímulo de entrada – saída) de RNA compostas por várias fileiras de neurônios dispostos em paralelo e funções de ativação contínuas. Com esta estrutura pode-se executar tarefas como o reconhecimento de padrão, diagnósticos complexos, previsão, etc.

Este trabalho visa, então, investigar o potencial do emprego do conceito de redes neurais artificiais ([15], [16], [25], [33], [35]) para a solução do problema da previsão de carga, com aprendizado efetuado via *Back-Propagation* (B-P). O algoritmo B-P é considerado, na literatura especializada, um *benchmark* em precisão, contudo sua convergência é muito lenta. Então, o propósito deste trabalho é ajustar a taxa de treinamento γ durante o processo de convergência, com o propósito de reduzir o tempo de execução. O ajuste γ é feito usando um controlador nebuloso. Usa-se também uma função exponencial decrescente que estabiliza a atuação do regulador no tempo de treinamento inicial e evita a instabilidade no processo de convergência.

Pretende-se, deste modo, propor um treinamento mais rápido, se comparado com a formulação convencional do B-P, conseqüentemente tornando o sistema predictor, a princípio, mais competitivo em comparação aos principais métodos encontrados na literatura especializada.

Visando, então, reduzir este tempo de treinamento, será desenvolvida uma versão do algoritmo B-P, utilizando-se um processo adaptativo baseado na lógica nebulosa, com o propósito de acelerar o processo de convergência, via observação do erro global da rede durante o treinamento. A característica principal desta rede, espera-se, é demandar um baixo esforço computacional para a execução do treinamento. Estes são os objetivos iniciais. Porém, à medida que foram obtidos os resultados com a implementação da rede neural, observou-se que, além da maior rapidez, os resultados eram mais precisos. Este fato passou, então, a ser o maior destaque da pesquisa.

Ressalta-se, ainda, que a rede neural nebulosa será bastante útil para a resolução de outros problemas, além da previsão de carga.



Este texto está organizado da seguinte forma: O capítulo 2 trata da teoria da previsão de carga e os modelos de Box-Jenkins utilizados para aproximar uma série temporal. No capítulo 3 é feita uma abordagem geral sobre redes neurais sendo que foi elaborado um estudo minucioso com relação ao algoritmo *Back-Propagation*. O capítulo 4 descreve as principais teorias da lógica nebulosa introduzidas por Zadeh ([30]). O desenvolvimento do sistema previsor de cargas elétricas utilizando o algoritmo *Back-Propagation*, com adaptação feita através de um controlador nebuloso, é discutido no capítulo 5. O capítulo 6 apresenta uma aplicação da rede neural nebulosa no campo de previsão de cargas elétricas. E finalmente, no capítulo 7 são apresentadas as conclusões e as sugestões para trabalhos futuros.



Neste capítulo, serão apresentados os principais conceitos relacionados com a previsão de cargas elétricas e os tipos de métodos utilizados para realizar essa previsão. Uma vez que a evolução da carga pode ser representada por uma série temporal, serão mostrados, de forma geral, os modelos de previsão de uma série temporal estacionária e não-estacionária, baseados na aproximação de Box-Jenkins. Trata-se, essencialmente, de um capítulo que relaciona os resultados mais relevantes sobre os conceitos e métodos abordados na literatura. Deve-se destacar que a principal referência usada neste capítulo é a ([10]) que é, certamente, um dos principais textos nesta área.

2.1 Cargas Elétricas

As cargas elétricas são compostas por milhares de aparelhos e dispositivos que, quando consideradas em conjunto, apresentam tendências que podem ser estatisticamente previstas. Tais tendências são influenciadas por fatores que interferem na forma de modelar a carga. Esses fatores podem ser divididos em: 1) condições meteorológicas, tais como, nebulosidade, velocidade do vento, variação brusca de temperatura, umidade; 2) fatores diversos como efeitos de dias atípicos (feriados) e dias especiais (sábados e domingos), semana típica onde corresponde ao ciclo de trabalho-descanso e 3) fatores sazonais, onde o consumo varia nas diferentes estações do ano. Tais fatores influenciam da melhor maneira o comportamento da carga para então efetuar-se a previsão.

Tem sido proposta, na literatura, uma variedade de métodos previsionais, de modo a estabelecer previsões com certo grau de precisão. Os métodos necessitam, basicamente, de duas condições fundamentais ([01]):



1. informações sobre o passado quantificadas em forma de dados;
2. assumir que o comportamento ocorrido no passado de certa forma irá se repetir no futuro.

Os métodos utilizados para previsão de carga podem ser classificados em três classes ([10]):

1. **Métodos Intuitivos:** Encontram-se nesta categoria os métodos de previsão que não utilizam um modelo matemático formal para descrever o comportamento da carga.
2. **Métodos Autônomos:** Compreendem os métodos de previsão que utilizam modelos baseados somente na extrapolação de observações passadas da carga. A técnica usada consiste na análise da evolução da carga no tempo, podendo, de acordo com determinados padrões de comportamento, ser classificada em horizontal, sazonal e tendenciosa. No padrão horizontal os dados da carga flutuam em torno de um determinado valor médio constante no tempo. Para o padrão tendencioso, os dados flutuam em torno de uma curva de tendência (média variável no tempo). No padrão sazonal, a carga é influenciada por fatores temporais, tais como: estações do ano, dias da semana. Já para o padrão cíclico as flutuações ocorrem variando em amplitude e duração.
3. **Métodos Causais:** Estes métodos assumem que a carga exibe uma relação de causa e efeito com uma ou mais variáveis independentes, que irão descrever seu comportamento. As variáveis utilizadas para justificar a tendência de crescimento da carga são: índices sócio-econômicos, crescimento populacional, etc.

A determinação do padrão de comportamento da carga é utilizado na escolha do modelo adequado visando descrever a evolução da carga no tempo e através dele pode-se dividir a série temporal (evolução da carga no tempo) em duas classes ([10]):

1. **Estacionária:** Uma série temporal é dita ser estacionária se os dados da carga tiverem flutuações em torno de uma média constante no tempo. Na Figura 2.1, tanto o item (a)



como o item (b), possuem dados que mantêm uma média constante no tempo, o que indica que a série temporal é estacionária.

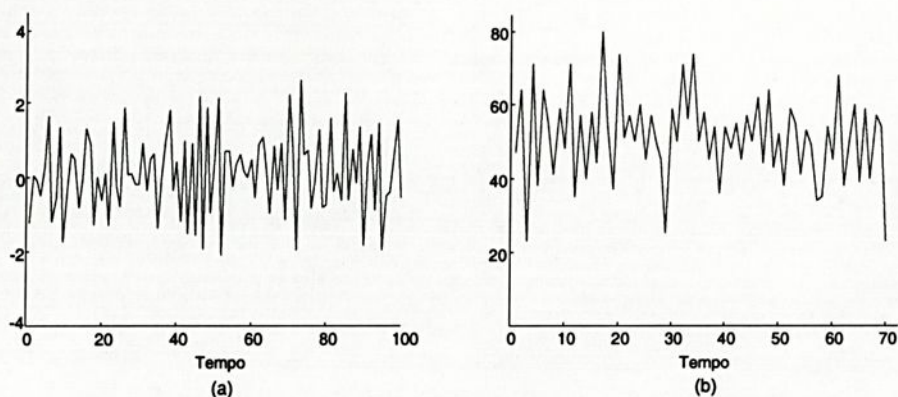


Figura 2.1: Série temporal estacionária.

2. **Não-Estacionária:** Se a série temporal apresentar o padrão tendencioso e/ou o padrão sazonal, ela é dita ser não-estacionária. Na Figura 2.2 nota-se que o item (a) apresenta um padrão sazonal sobreposto semelhante a um padrão horizontal e o item (b) exibe um padrão sazonal com aparente tendência para cima caracterizando assim uma série temporal não-estacionária.

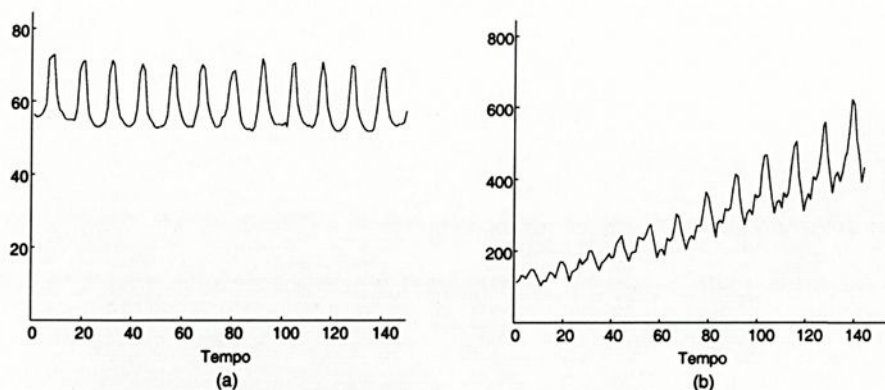


Figura 2.2: Série temporal não-estacionária.

Após selecionar a classe de modelos que melhor descreve os padrões de comportamento, é preciso escolher um modelo adequado desta classe que melhor descreve a série temporal com o propósito de realizar previsões para valores futuros da série temporal.

2.2 Previsão para Modelos de Série Temporal Estacionária

Uma série temporal consiste de uma seqüência ordenada de valores de uma variável para intervalos de tempo igualmente espaçados ([24]). A série temporal de t observações, $z_1, z_2, \dots, z_t$, são consideradas como valores observados das variáveis aleatórias $Z_1, Z_2, \dots, Z_t$ e o valor dela no período de tempo l no futuro é expresso por z_{t+l} (valor aleatório da variável Z_{t+l}). A previsão de z_{t+l} , indicada por $\hat{z}_t(l)$, é baseada nos valores observados da série temporal $z_1, z_2, \dots, z_t$ com origem em t e avanço de tempo l ([10]).

Se a série temporal é estacionária, as variáveis $Z_1, Z_2, \dots, Z_t$ possuem a mesma média μ , logo a previsão de z_{t+l} é exatamente μ , o valor esperado da variável Z_{t+l} . Assim, o erro de previsão é:

$$e_t(l) = Z_{t+l} - \hat{z}_t(l)$$

Através da distribuição normal de Z_{t+l} verifica-se que a probabilidade do valor futuro z_{t+l} estar situado entre os limites:

$$\hat{z}_t(l) \pm (1.96) (\sigma^2[e_t(l)])^{1/2}$$

é de 95%, sendo $\sigma^2[e_t(l)]$ a variância do erro de previsão. O intervalo entre estes limites é dito ser 95% do intervalo de predição para z_{t+l} .

Na prática, a previsão e a variância do erro de previsão dependerá de parâmetros desconhecidos do modelo da série temporal e estes serão estimados dos valores da série temporal observada $z_1, z_2, \dots, z_t$ ([10]).

2.2.1 Função de Autocorrelação e Autocorrelação Parcial Amostral



A Função de Autocorrelação Amostral (*Sample AutoCorrelation Function* - ACF) e Função de Autocorrelação Parcial Amostral (*Sample Partial AutoCorrelation Function* - PACF) representa um passo importante no estágio de identificação do processo de Box-Jenkins.

Uma função de autocorrelação é a correlação entre o comportamento anterior da carga e o fator que se espera influenciar a carga no futuro. Assim, a autocorrelação amostral de atraso k , r_k para $k = 1, 2, \dots$, é dada da seguinte forma ([10]):

$$r_k = \frac{\sum_{t=1}^{N-k} (z_t - \bar{z}) (z_{t+k} - \bar{z})}{\sum_{t=1}^N (z_t - \bar{z})^2}$$

sendo $\bar{z}$ a média amostral da série temporal.

Para obter valores úteis de autocorrelações amostrais, necessita-se de pelo menos 50 observações e elas serão calculadas apenas para $k = 1, \dots, K$ em que K não será maior que $N/4$, onde N corresponde ao número de observações da série. É importante ressaltar que as autocorrelações amostrais são somente estimativas de autocorrelações teóricas, ρ_k , logo tendem ao mesmo padrão, de maneira que se pode concluir muitas propriedades do processo estocástico subjacente a partir de um estudo da função de autocorrelação amostral ([10]).

A autocorrelação amostral é de fundamental importância, mas infelizmente o ACF amostral nem sempre permite especificar o modelo apropriado. Exige-se, então, outra característica da série temporal que é a função de autocorrelação parcial amostral expressa por r_{kk} .

O PACF amostral é também, por sua vez, uma estimativa da autocorrelação parcial teórica, ρ_{kk} , calculada dos valores da série temporal observada. A autocorrelação



parcial teórica é definida como sendo a autocorrelação entre quaisquer duas variáveis Z_t e Z_{t+k} , separadas k atrasos de tempo, sendo que as variáveis $Z_{t+1}, Z_{t+2}, \dots, Z_{t+k-1}$ são eliminadas. As autocorrelações parciais amostrais possuem a seguinte forma ([10]):

$$r_{11} = r_1$$

$$r_{22} = \frac{r_2 - r_1^2}{1 - r_1^2}$$

sendo que as equações de r_{33} por diante, tornam-se cada vez mais complicadas.

Da mesma forma ocorrida com o ACF amostral, o PACF amostral tende ao mesmo padrão que o PACF teórico de tal forma, que se pode utilizar o PACF amostral para identificar o modelo apropriado no processo estocástico.

2.2.2 Formas Gerais dos Modelos ARMA

A classe de modelos que melhor apresenta as características da série temporal estacionária é a classe de modelos ARMA ([09], [10], [24]). Dentre eles podem-se citar os seguintes modelos: o modelo Ruído Branco, o modelo *Auto Regressive* (AR), o modelo *Moving Average* (MA) e finalmente o modelo *Auto Regressive Moving Average* (ARMA).

2.2.2.1 Modelo Ruído Branco ([10], [24])

O modelo é definido por ([10], [24]):

$$Z_t = \theta_0 + A_t$$

em que A_t são os *random shocks*, $A_t \sim \text{IN}(0, \delta_A^2)$, e θ_0 é o termo constante.

2.2.2.2 Modelo Auto Regressive (AR(p)) ([10], [24])

O modelo é definido pela equação ([10], [24]):

$$Z_t = \theta_0 + \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \dots + \phi_p Z_{t-p} + A_t$$

sendo que A_t representa os *random shocks*, $A_t \sim \text{IN}(0, \delta_A^2)$, θ_0 é o termo constante, $\phi_1, \dots, \phi_p$ são coeficientes autoregressivos e p é a ordem do modelo.

2.2.2.3 Modelo Moving Average (MA(q)) ([10], [24])

O modelo é definido da seguinte forma ([10], [24]):

$$Z_t = \theta_0 + A_t - \theta_1 A_{t-1} - \theta_2 A_{t-2} - \dots - \theta_q A_{t-q}$$

no qual A_t representa os *random shocks*, sendo $A_t \sim \text{IN}(0, \delta_A^2)$, θ_0 é o termo constante, $\theta_1, \dots, \theta_p$ são coeficientes de média móvel e q é a ordem do modelo.

2.2.2.4 Modelo Auto Regressive Moving Average (ARMA(p,q)) ([10], [24])

Este modelo combina no modelo ambos termos autoregressive e de moving average. O modelo é definido da seguinte maneira ([10], [24]):

$$Z_t = \theta_0 + \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \dots + \phi_p Z_{t-p} + A_t - \theta_1 A_{t-1} - \theta_2 A_{t-2} - \dots - \theta_q A_{t-q}$$

sendo que A_t representa os *random shocks*, em que $A_t \sim \text{IN}(0, \delta_A^2)$, θ_0 é o termo constante, $\phi_1, \dots, \phi_p$ e $\theta_1, \dots, \theta_p$ são simultaneamente, os coeficientes autoregressivos e de média móvel e p e q são as ordens do modelo.



Na Tabela 2.1, estão relacionados os seis modelos da classe de modelos ARMA e suas respectivas propriedades características.

Tabela 2.1: Definições dos seis modelos ARMA.

Modelo	Notação ARMA	Definição do Modelo	Condições de Inversibilidade e Estacionaridade
Ruído Branco	ARMA(0,0)	$Z_t = \theta_0 + A_t$	-
AR(1)	ARMA(1,0)	$Z_t = \theta_0 + \phi_1 Z_{t-1} + A_t$	$-1 < \phi_1 < 1$
AR(2)	ARMA(2,0)	$Z_t = \theta_0 + \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + A_t$	$\phi_2 + \phi_1 < 1$ $\phi_2 - \phi_1 < 1$ $-1 < \phi_2 < 1$
MA(1)	ARMA(0,1)	$Z_t = \theta_0 + A_t - \theta_1 A_{t-1}$	$-1 < \theta_1 < 1$ $\theta_2 + \theta_1 < 1$
MA(2)	ARMA(0,2)	$Z_t = \theta_0 + A_t - \theta_1 A_{t-1} - \theta_2 A_{t-2}$	$\theta_2 - \theta_1 < 1$ $-1 < \theta_2 < 1$
Auto Regressive Moving Average	ARMA(1,1)	$Z_t = \theta_0 + \phi_1 Z_{t-1} + A_t - \theta_1 A_{t-1}$	$-1 < \phi_1 < 1$ $-1 < \theta_1 < 1$

2.3 Previsão para Série Temporal Estacionária

Na análise da série temporal pode-se utilizar a aproximação de Box-Jenkins ([10]) a qual é dividida em três estágios:

1. escolha de uma classe de modelos convenientes para o ajuste da série temporal observada;
2. adaptação do modelo apropriado desta classe para série temporal observada;
3. utilização do modelo adaptado para realizar previsões de valores futuros da série temporal observada.

A Figura 2.4 mostra a forma geral da aproximação de Box-Jenkins ([10]).



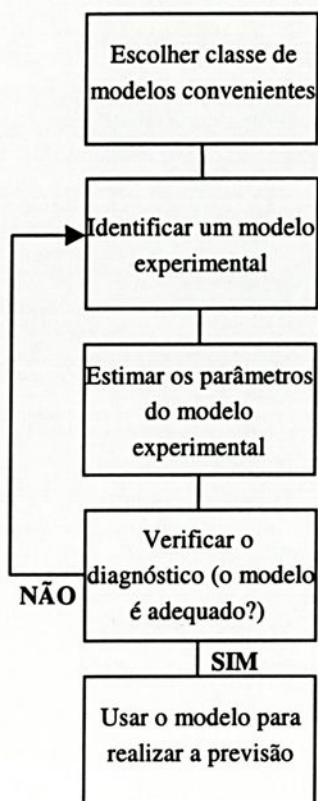


Figura 2.3: Estágios na aproximação de Box-Jenkins.

O segundo estágio pode ser subdividido em três partes ([10]): *identificação*, *estimativa* e *verificação de diagnósticos*. Estas três partes são descritas a seguir.

1. Identificação de um Modelo Experimental

A técnica básica no estágio de identificação é a comparação da ACF Amostral e PACF Amostral da série temporal com os padrões de ACF e PACF teóricos de modelos da classe escolhida.

Quando dois ou mais modelos forem selecionados para o estágio de identificação, utilizar-se-á o que Box and Jenkins chamam de *princípio de parametrização parcimoniosa* ([10]), ou seja, o modelo escolhido dentre eles será aquele que apresentar o menor número de parâmetros. O fato se deve por dois motivos: 1) a dificuldade de estimar precisamente os parâmetros do modelo se ele contém muitos parâmetros e 2) se o modelo for complicado, a

detecção de diagnóstico pode não descobrir se ele contém mais parâmetros que o necessário para representar adequadamente a série temporal observada.

Um passo importante, através das autocorrelações e autocorrelações parciais amostrais, é identificar se as autocorrelações, r_k , e autocorrelações teóricas, ρ_k , para todo $k > q$, são ou não iguais a zero. Para isso, utilizam-se os chamados testes de significância para as autocorrelações e autocorrelações parciais, ou seja:

Se a amostra de autocorrelação r_k encontrar-se fora dos limites ([10]):

$$\pm \frac{1.96}{N^{1/2}} (1 + 2 \sum_{i=1}^k r_i^2)^{1/2}$$

então, ela é significativamente diferente de zero e, assim, $\rho_k \neq 0$. Se r_k não é significativamente diferente de zero, então, $\rho_k = 0$.

Se a amostra de autocorrelação parcial r_{kk} encontrar-se fora dos limites:

$$\pm \frac{1.96}{N^{1/2}}$$

ela é significativamente diferente de zero e, então, $\rho_{kk} \neq 0$. Se r_{kk} não é significativamente diferente de zero, então, $\rho_{kk} = 0$.

sendo N o número de observações na realização.

Após selecionar um modelo da classe de modelos ARMA, é importante verificar se o termo constante θ_0 deve ou não ser incluído no modelo. Através do modelo ARMA(p,q) o termo constante θ_0 é relacionado ao processo de média μ como mostra a Equação (2.3.1) ([10]):

$$\theta_0 = \mu (1 - \phi_1 - \phi_2 - \dots - \phi_p) \quad (2.3.1)$$

Da Equação (2.3.1) observar-se que o termo constante θ_0 será zero, se e somente se, o processo de média μ é zero. Como a média μ do processo estocástico gerado pela série



temporal observada é estimada da média amostral $\bar{z}$, pode-se, então, ter o seguinte teste de significância para θ_0 :

Se a média amostral $\bar{z}$ é grande comparada com o desvio padrão amostral $\hat{\sigma}_z$, incluir o termo constante no modelo.

ou seja, se a média amostral $\bar{z}$ difere de zero, o termo constante θ_0 deve ser incluído no modelo.

A Tabela 2.2 resume de que maneira as funções de autocorrelação e autocorrelação parcial amostral são utilizadas na identificação do modelo ([09], [10], [24]).

Tabela 2.2: Propriedades características dos seis modelos ARMA.

Modelo	ACF Teórico	PACF Teórico
Ruído Branco	Todas autocorrelações são zeros	Todas autocorrelações parciais são zero
AR(1)	Decai para zero	Termina após o atraso 1
AR(2)	Decai para zero	Termina após o atraso 2
MA(1)	Termina após o atraso 1	Decai para zero
MA(1)	Termina após o atraso 2	Decai para zero
ARMA(1,1)	Decai para zero	Decai para zero

2. Estimação dos Parâmetros do Modelo Experimental

Após o modelo experimental ser selecionado, o próximo passo será estimar os parâmetros desconhecidos que fazem parte deste modelo. O erro das previsões a um passo adiante do modelo ARMA(p,q) ([10]):

$$e_{t-1}(1) = Z_t - \hat{z}_{t-1}(1) \tag{2.3.2}$$



é exatamente o random shock A_t . Então, o valor observado do erro da previsão a um passo adiante:

$$z_t - \hat{z}_{t-1}(1) \quad (2.3.3)$$

é igual a a_t , a realização do random shock A_t para o tempo t .

A soma dos quadrados é dada por ([10]):

$$\sum_{t=1}^N (z_t - \hat{z}_{t-1}(1))^2 \quad (2.3.4)$$

cujos valores dependem de parâmetros desconhecidos no modelo, $\theta_0, \phi_1, \dots, \phi_p, \theta_1, \dots, \theta_q$, assim como nas observações $z_1, z_2, \dots, z_N$.

Seguindo o princípio dos mínimos quadrados, a estimativa dos parâmetros desconhecidos $\theta_0, \phi_1, \dots, \phi_p, \theta_1, \dots, \theta_q$ são valores no qual a soma dos quadrados dada pela Equação (2.3.4) é mínima. A soma dos quadrados depende das observações $z_1, z_2, \dots, z_N$ assim como os valores dos parâmetros desconhecidos $\theta_0, \phi_1, \dots, \phi_p, \theta_1, \dots, \theta_q$ para o qual a soma dos quadrados é mínima também dependerá das observações $z_1, z_2, \dots, z_N$.

Então, cada estimativa dos mínimos quadrados destes parâmetros é uma função das observações $z_1, z_2, \dots, z_N$, denotadas por $\hat{\theta}_0, \hat{\phi}_1, \dots, \hat{\phi}_p, \hat{\theta}_1, \dots, \hat{\theta}_q$.

No geral, não é possível encontrar expressões para estimar os mínimos quadrados dos parâmetros dos modelos ARMA(p,q) em termos das observações $z_1, z_2, \dots, z_N$. Porém, é possível, através de um algoritmo de computação dos mínimos quadrados não-linear, encontrar a função soma dos quadrados, Equação (2.3.4), para localizar o mínimo. Entretanto, a estimativa dos mínimos quadrados requer estimativas preliminares desses parâmetros. Em geral, é necessário verificar se estas estimativas preliminares satisfazem os conceitos de estacionaridade e inversibilidade exigidas pelo modelo.

Deve-se, então, estabelecer estimativas preliminares de todos os parâmetros com exceção do termo constante θ_0 que é igual a um número pequeno, por exemplo 0,1. Isso



ocorre porque as estimativas preliminares devem cumprir as condições de estacionaridade e inversibilidade exigidas pelo modelo. Para um modelo ARMA(p,q) a estimativa preliminar do termo constante θ_0 , Equação (2.3.1), é dada por:

$$\bar{z}(1 - \hat{\phi}_1 - \dots - \hat{\phi}_p) \quad (2.3.5)$$

em que $\hat{\phi}_1, \dots, \hat{\phi}_p$ são respectivamente estimativas preliminares de $\phi_1, \dots, \phi_p$.

Existem dois métodos para encontrar as estimativas dos quadrados mínimos:

Método 1:

Se o termo constante θ_0 for incluído no modelo, então, a estimativa preliminar de θ_0 é obtida utilizando a Equação (2.3.5). Por conseguinte, assumindo que o parâmetro θ_0 tem o valor dado por esta estimativa, as estimativas dos quadrados mínimos são criadas para outros parâmetros no modelo. Uma estimativa final de θ_0 é obtida substituindo a estimativa dos quadrados mínimos correspondente na Equação (2.3.1) ([10]).

Método 2:

Se o termo constante θ_0 for incluído no modelo, as estimativas dos mínimos quadrados serão encontradas para todos os parâmetros do modelo, incluindo o termo constante θ_0 ([10]).

Geralmente as estimativas encontradas usando o método 1 são muito boas e dão previsões muito similares as estimativas obtidas pelo método 2. A vantagem do método 1 é que reduz a um, o número de parâmetros a ser estimado.

3. Verificação do Diagnóstico do Modelo Experimental



A próxima etapa, é verificar se as estimativas dos parâmetros satisfazem as condições de estacionaridade e inversibilidade exigidas pelo modelo.

Deve-se observar, primeiramente, se as estimativas dos parâmetros são significativamente diferentes de zero. Uma maneira de verificar isso é calcular a razão-*t* ([10]):

$$\text{razão-}t = \frac{\text{parâmetro estimado}}{\text{erro padrão estimado}}$$

(2.3.6)

Se esta razão-*t*, Equação (2.3.6), estiver fora dos limites $\pm 1,96$, o parâmetro estimado é significativamente diferente de zero.

sendo que a expressão do erro padrão estimado dependerá do modelo a ser ajustado, como mostra a Tabela 2.3.

Tabela 2.3: Erro padrão estimado dos quadrados mínimos dos modelos ARMA.

Modelo	Parâmetro Estimado	Erro Padrão Estimado
AR(1)	$\hat{\phi}_1$	$\left[\frac{1 - (\hat{\phi}_1)^2}{N} \right]^{1/2}$
AR(2)	$\hat{\phi}_1, \hat{\phi}_2$	$\left[\frac{1 - (\hat{\phi}_2)^2}{N} \right]^{1/2}$
MA(1)	$\hat{\theta}_1$	$\left[\frac{1 - (\hat{\theta}_1)^2}{N} \right]^{1/2}$
MA(1)	$\hat{\theta}_1, \hat{\theta}_2$	$\left[\frac{1 - (\hat{\theta}_2)^2}{N} \right]^{1/2}$
ARMA(1,1)	$\hat{\phi}_1$	$\left[\frac{(1 - (\hat{\phi}_1)^2)(1 - \hat{\phi}_1 \hat{\theta}_1)^2}{N (\hat{\phi}_1 - \hat{\theta}_1)^2} \right]^{1/2}$
ARMA(1,1)	$\hat{\theta}_1$	$\left[\frac{(1 - (\hat{\theta}_1)^2)(1 - \hat{\phi}_1 \hat{\theta}_1)^2}{N (\hat{\phi}_1 - \hat{\theta}_1)^2} \right]^{1/2}$



Estas regras devem ser consideradas como rígidas, já que um parâmetro estimado pode não ser significativo em consequência do pequeno número de observações na série temporal. Mas, para um número grande de observações, a estimativa deste parâmetro pode ser significativo.

Se um parâmetro estimado é significativo, isto indica que o termo correspondente deve ser incluído no modelo, por outro lado, se ele não é significativo pode-se abandonar o termo correspondente deste parâmetro no modelo.

O principal teste de modelos adequados é baseado em um estudo dos *resíduos* do modelo ajustado. Se os parâmetros desconhecidos são substituídos por suas estimativas dos mínimos quadrados no cálculo da previsão $\hat{z}_{t-1}(1)$, o erro da previsão a um passo adiante pode ser expresso pela Equação (2.3.3) sendo chamado *resíduo* para o tempo t (diferença entre o valor observado para o tempo t e o previsto utilizando seu modelo ajustado) e denotado por $\hat{a}_t$ ([10]).

O valor mínimo da soma dos quadrados, Equação (2.3.4), que é obtido quando as estimativas dos mínimos quadrados são usadas para os parâmetros desconhecidos, é exatamente a soma dos *resíduos* ao quadrado:

$$\sum_1^N (\hat{a}_t)^2 \quad (2.3.7)$$

conhecido como *soma dos resíduos quadrados*.

O estudo dos *resíduos* $\hat{a}_t$ baseia-se na verificação de uma realização do processo de Ruído Branco. Se o modelo ajustado é certo, então os *resíduos* são uma realização do processo de Ruído Branco e, portanto, as autocorrelações e autocorrelações parciais amostrais dos *resíduos* devem todas estar próxima de zero.

Considere $r_l(\hat{a})$ a autocorrelação amostral de atraso l para os *resíduos* e

$$Q = N \sum_{l=1}^k (r_l(\hat{a}))^2$$

em que N é o número de resíduos para a série temporal observada.



Se Q é muito grande, o modelo será inadequado, pois tendo Q um valor grande, as autocorrelações $r_i(\hat{a})$ são também grandes para os *resíduos* assemelhar-se a uma realização de um processo de Ruído Branco. Se o modelo é adequado, a variável aleatória Q possui uma distribuição de probabilidade denominada *chi-quadrado* com $(k-r)$ graus de liberdade, em que r é o número total de parâmetros do modelo ajustado ([10]).

Das tabelas de distribuição de probabilidade, é possível encontrar um valor crítico tal que a probabilidade de um valor aleatório de Q exceder este valor crítico é de 0,05. Daí pode-se utilizar o seguinte teste conhecido como teste de ajuste da *portmanteau lack* sendo Q conhecido como *chi-quadrado de Box-Pierce*:

Se o valor calculado de Q excede o valor crítico calculado das tabelas *chi-squared*, este valor de Q é muito grande e o modelo é julgado inadequado. Caso contrário, o modelo é considerado como sendo adequado.

O estudo da autocorrelação amostral de *resíduos* pode indicar os modelos inadequados e sugerir que um novo parâmetro seja incluído no modelo. Se o teste de diagnóstico não detectar qualquer inadequações no modelo ajustado, pode-se considerar um parâmetro adicional no modelo com a finalidade de que o novo modelo seja mais adequado se comparado ao modelo anterior. Este teste é conhecido como *overfitting* ([10]).

Quando o novo modelo é ajustado, se o parâmetro estimado do novo parâmetro não é significativamente diferente de zero, então, não vale a pena inclui-lo no modelo. Neste caso, a estimativa do outros parâmetros no novo modelo será similar as estimativas correspondentes no modelo anterior.

Para verificar se o novo modelo experimental tem um melhor ajuste que o modelo experimental anterior, calcula-se a variância *residual* para cada modelo da seguinte forma

$$\frac{1}{N-1} \sum_{i=1}^N (\hat{a}_i - \bar{\hat{a}})^2$$

sendo



$$\bar{\hat{a}}_t = \frac{1}{N} \sum_{t=1}^N \hat{a}_t$$

a média *residual*.

Se a média residual está próxima de zero e N é grande, a variância residual é ([10]):

$$\frac{1}{N} \sum_{t=1}^N (\hat{a}_t)^2 \quad (2.3.8)$$

Comparando as Equações (2.3.7) e (2.3.8), tem-se que a variância residual é exatamente o valor mínimo da soma da função quadrada dividida por N . Para modelos ARMA, a variância residual é uma estimativa de σ_A^2 (variância do *random shocks* A_t , sendo os *random shocks* componentes imprevisíveis, pode-se dizer que o modelo preferido é aquele com menor variância residual).

Supõe-se que se dois ou mais modelos adequados foram escolhidos e possuem a mesma variância residual, deve-se então utilizar neste momento o princípio de *parametrização parcimoniosa*, em que o modelo escolhido será aquele que apresentar menor número de parâmetros. Se os modelos possuem o mesmo número de parâmetros, tem-se que os modelos são realmente similares e darão previsões a curto prazo similares. Se os modelos são similares porém produzem previsões diferentes, deve-se utilizar os seguintes passos:

1. supor uma série temporal com N observações. Ajustar os modelos alternativos para as primeiras $0,9*N$ observações, e calcular as previsões para as próximas $0,1*N$ observações usando cada método;
2. comparar as previsões com valores reais das próximas $0,1*N$ observações e assim calcular o erro da previsão;
3. a medida da previsão de cada modelo é obtida calculando o erro quadrado médio das previsões que é exatamente a soma dos quadrados dos erros da previsão, dividida pelo número de previsões;
4. o modelo com mínimo erro quadrado médio é escolhido como modelo para realizar a previsão, e seus parâmetros são estimados novamente para as N observações avaliadas.
5. este modelo adaptado é utilizado para prever valores futuros da série temporal.



2.4 Previsão para Série Temporal Não-Estacionária

Uma série temporal pode ser não-estacionária na média ou na variância ou até em ambas. Ela pode ser não-estacionária na média devido a um padrão tendencioso ou sazonal e não-estacionária na variância se a média e o desvio padrão de cada subconjunto calculado da série temporal são proporcionais ou independentes um do outro.

Uma série temporal não-estacionária na variância pode ser transformada em uma série temporal estacionária na variância. Uma transformação adequada é a transformação logarítmica em que ([10]):

$$z_t^* = \log_e(z_t)$$

consiste de logaritmos naturais dos termos da série temporal original.

Outra transformação bastante útil é a transformação da raiz quadrada na qual considera-se a seguinte série temporal observada ([10]):

$$z_t^* = (z_t)^{1/2}$$

Em seguida, deve-se descobrir qual série temporal transformada produzirá a previsão mais precisa. Então, as previsões para série temporal transformada são convertidas em previsões para série temporal original.

Se uma série temporal é não-estacionária na média, pode-se utilizar um mecanismo conhecido como *differencing* com o propósito de transformar a série temporal em uma série temporal transformada estacionária na média. Utiliza-se, então, uma série temporal determinística, em que seus valores futuros são obtidos de uma função matemática.

Supondo-se, então, uma série determinística linear, e desta série uma segunda série obtida subtraindo valores consecutivos da série determinística. Esta segunda série temporal é dita consistir de primeiras diferenças regulares da série temporal original, e é denotada por ([10]):

$$w_t = z_t - z_{t-1} \quad (2.4.1)$$



O número de observações na série temporal de primeiras diferenças regulares é menor do que o número de observações na série temporal original.

Se a série determinística utilizada for quadrática, e desta série uma segunda série for obtida subtraindo valores consecutivos da série determinística, sendo que esta segunda série temporal obtida não é estacionária. Subtraindo os termos consecutivos desta série temporal obtida, obtêm-se uma terceira série temporal que é dita consistir de segundas diferenças regulares da série temporal original, e é denotada por ([10]):

$$w_t = [z_t - z_{t-1}] - [z_{t-1} - z_{t-2}] \quad (2.4.2)$$

e nota-se que o número de observações na série temporal de segundas diferenças regulares é duas vezes menor que o número de observações na série temporal original.

Freqüentemente utiliza-se uma notação para expressar a série temporal diferenciada em termos da série original, que usa o deslocamento *Backward* (sentido inverso) no operador B , de uma forma geral tem-se ([10]):

$$B^k z_t = z_{t-k} \quad (2.4.3)$$

Substituindo a Equação (2.4.3) nas Equações (2.4.1) e (2.4.2), tem-se uma idéia de como obter a série temporal diferenciada w_t relacionada a série temporal original z_t para d diferenças regulares, sendo d o número de diferenças regulares necessário para obter a estacionaridade da série temporal. Desta forma, a série temporal diferenciada é dada da seguinte forma ([10]):

$$w_t = (1 - B)^d z_t$$

portanto, utilizando um número adequado de diferenças regulares da série temporal original, a série temporal não-estacionária na média pode ser transformada em uma série temporal estacionária na média.

O número de d diferenças regulares exigidas será determinado pelo ACF amostral da série temporal original e de cada série temporal diferenciada derivada dela. Se o ACF amostral decai gradual e extremamente devagar, tem-se que a série temporal é não-estacionária. Logo a série necessita das primeiras diferenças regulares. Em seguida faz-se a análise do ACF da série temporal diferenciada. Se ela não for estacionária, utilizam-se as



segundas diferenças regulares e assim por diante, até que a série temporal diferenciada seja estacionária. Ocasionalmente podem ser exigidas duas diferenças regulares, mas raramente requer mais que duas.

O processo de *overdifferencing*, no qual tomam-se mais diferenças regulares que o necessário para obter a estacionaridade, deve ser evitado, pois este processo é realizado de forma a obter um modelo mais adequado que o anterior, quer dizer com menor variância, e com o aumento das diferenças regulares a variância crescerá cada vez mais.

2.5 Classe de Modelos para uma Série Temporal Não-Estacionária

Os modelos ARIMA(p,d,q) são as classes mais comuns de modelos que melhor representam uma série temporal não-estacionária, no qual pode ser estacionarizado através da transformação *differencing* ([09], [10], [24]). Dentre elas tem-se:

2.5.1 Modelo Walk Aleatório ([10])

O modelo é definido por ([10]):

$$Z_t = Z_{t-1} + \theta_0 + A_t$$

em que A_t são os *random shocks*, com $A_t \sim \text{IN}(0, \delta_A^2)$ e θ_0 é o termo constante. Este modelo é caracterizado pelo fato de que as primeiras diferenças regulares desse modelo forma um modelo de Ruído Branco. Neste modelo, a previsão para todos os períodos de tempo futuro é exatamente o último valor observado da série temporal.

2.5.2 Modelos Auto Regressive Integrated Moving Average (ARIMA) ([10])

São caracterizados pelas primeiras diferenças regulares formarem modelos ARMA, tais como AR(1), AR(2), MA(1) e MA(2) ([10]).



O primeiro destes é conhecido como modelo *Auto Regressive Integrated* ARI(1,1) sendo definido como ([10]):

$$Z_t = \theta_0 + (1 + \phi_1)Z_{t-1} - \phi_1 Z_{t-2} + A_t$$

em que as primeiras diferenças regulares forma um modelo AR(1). O modelo ARI(2,1) é encontrada de forma similar e as primeiras diferenças regulares forma um modelo AR(2).

O modelo IMA(1,1) é definido como ([10]):

$$Z_t = \theta_0 + Z_{t-1} - \theta_1 A_{t-1} + A_t$$

no qual as primeiras diferenças regulares formam um modelo MA(1). O modelo IMA(1,2) é encontrado, de forma similar, sendo que as primeiras diferenças regulares formam um modelo MA(2).

O último modelo a ser considerado é o modelo ARIMA(1,1,1) sendo que as primeiras diferenças regulares deste modelo formam um modelo ARMA(1,1). Este modelo é descrito como ([10]):

$$Z_t = \theta_0 + (1 + \phi_1)Z_{t-1} - \phi_1 Z_{t-2} + A_t - \theta_1 A_{t-1}$$

e de forma geral, os modelos considerados são todos casos particulares de modelos ARIMA (p,1,q) cujas primeiras diferenças regulares formam modelos ARMA(p,q).

Os modelos ARIMA(p,d,q) são aqueles cujas d -ésimas diferenças regulares formam modelos ARMA(p,q). É importante lembrar que na prática modelos ARIMA(p,d,q) com $d \geq 2$ não ocorrem.

Para um modelo ARIMA(p,1,q), se o termo constante θ_0 é zero a previsão tende a uma linha reta horizontal, e se θ_0 não é zero, as previsões tendem a uma linha reta com inclinação μ , em que ([10]):

$$\mu = \frac{\theta_0}{1 - \phi_1 - \phi_2 - \dots - \phi_p}$$

sendo que se θ_0 for positivo a inclinação será para cima e se θ_0 for negativo a inclinação será para baixo.

A Tabela 2.4 indica as propriedades características dos modelos da classe de modelos ARIMA.

Tabela 2.4: Propriedades Características dos seis modelos ARIMA(p,1,q).

Modelo	Notação ARIMA	ACF Teórico	ACF Teórico Primeiras Diferenças Regulares	PACF Teórico Primeiras Diferenças Regulares
Walk Aleatório	ARIMA(0,1,0)	Decai lentamente	Todas autocorrelações são zeros	Todas autocorrelações parciais são zeros
ARI(1,1)	ARIMA(1,1,0)	Decai lentamente	Decai para zero	Termina após o atraso 1
ARI(2,1)	ARIMA(2,1,0)	Decai lentamente	Decai para zero	Termina após o atraso 2
IMA(1,1)	ARIMA(0,1,1)	Decai lentamente	Termina após o atraso 1	Decai para zero
IMA(1,2)	ARIMA(0,1,2)	Decai lentamente	Termina após o atraso 2	Decai para zero
ARIMA(1,1,1)	ARIMA(1,1,1)	Decai lentamente	Decai para zero	Decai para zero

2.6 Conclusão

Neste capítulo, tratou-se, de maneira geral, do conceito de cargas elétricas. Como ela pode ser representada por uma série temporal, deu-se ênfase à compreensão das séries temporais. Estudou-se desde a identificação da estacionaridade da série temporal até as características das classes de modelos de Box-Jenkins, tanto para uma série temporal estacionária, quanto para uma série temporal não-estacionária. Também foi visto, como a análise da série temporal é feita pela aproximação de Box-Jenkins e a partir dela, como são realizadas as previsões da série temporal estacionária e da série temporal não-estacionária.



No Capítulo anterior foram apresentados os principais conceitos e métodos para modelagem e previsão de cargas elétricas convencionalmente abordados na literatura. A grande dificuldade destes procedimentos é a necessidade de se desenvolver a modelagem prévia da carga para aplicação posterior. Então, a qualidade das soluções depende de uma boa escolha do modelo do comportamento da carga, baseado, principalmente, na determinação das influências de cada variável adotada no modelo considerado. Esta atividade é bastante complexa. Exige, portanto, um grande empreendimento de esforços, conhecimento e habilidade dos profissionais da área no trato deste tipo de problema. Contudo, as redes neurais têm se mostrado de forma eficiente no desenvolvimento de séries temporais, principalmente, porque não há necessidade de se conhecer, *a priori*, o modelo da carga. As correlações entre variáveis são estabelecidas de forma autônoma. Esta particularidade torna-se bastante importante para o problema de previsão de cargas elétricas.

Deve-se ressaltar, porém, para que se obtenha os resultados satisfatórios com o uso de redes neurais é indispensável efetuar um pré-processamento do conjunto de dados na fase de treinamento que é uma atividade que pode ser realizada sem consumir grandes quantidades de cálculos computacionais. Assim sendo, nos capítulos subsequentes será proposto um sistema para a previsão de cargas elétricas baseado no uso de redes neurais.

Neste capítulo, serão apresentados os conceitos sobre redes neurais artificiais, com o propósito de desenvolver o mecanismo para a previsão de cargas elétricas. Apresentar-se-ão a estrutura da rede neural *feedforward*, seus componentes e o seu funcionamento. Também, serão abordadas as formas de treinamento, dando ênfase ao algoritmo *Back-Propagation* Convencional e ao *Back-Propagation* com Momento.

3.1 Modelo Neural Biológico



As Redes Neurais Artificiais são técnicas computacionais que apresentam um modelo matemático inspirado na estrutura neural de organismos inteligentes e que adquirem conhecimento através da experiência ([21]).

O cérebro humano é composto por bilhões de neurônios que executam papel essencial na determinação do funcionamento e comportamento do corpo humano e do raciocínio. Na Figura 3.1 são mostrados os vários componentes de um neurônio biológico mais comum ([18], [21]).

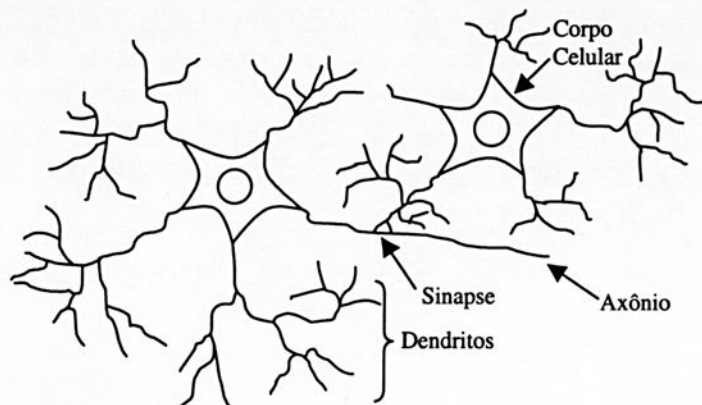


Figura 3.1: Componentes de um neurônio biológico.

No cérebro humano, os neurônios captam os sinais entre si através de inúmeras estruturas finas chamadas *dendritos* (terminais de entrada) e enviam atividades elétricas através de um longo e fino suporte conhecido como *axônio* (terminal de saída) ([21], [28]).

Os neurônios são interligados por meio das *sinapses*, região na qual dois neurônios entram em contato e transmitem impulsos nervosos. Através desta conectividade é atribuída a inteligência ([18]).

3.2 Modelo Neural Artificial

Com o desenvolvimento da inteligência artificial, surgiram idéias de como representar, através de determinados programas, o funcionamento do processo de aprendizagem do cérebro humano. A tentativa de emular a rede neural do cérebro deu origem à chamada Rede Neural Artificial ([18]).

Semelhante ao cérebro humano, a rede neural possui um neurônio artificial que foi designado a imitar as características do neurônio biológico ([18], [33]). A Figura 3.2 mostra os componentes de um neurônio artificial proposto por McCulloch e Pitts ([28]).

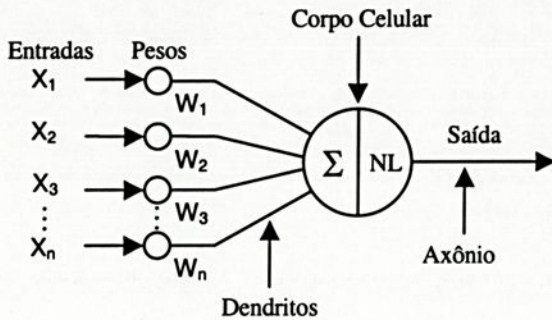


Figura 3.2: Componentes de um neurônio artificial.

Um conjunto de entradas X , $x_1, x_2, \dots, x_n$, são aplicados ao neurônio artificial. Estas entradas correspondem aos sinais dentro das sinapses de um neurônio biológico. Cada sinal é multiplicado por um peso associado W , $w_1, w_2, \dots, w_n$, que adiante é aplicado a um bloco somatório, Σ , denominado soma, que corresponde ao corpo celular biológico. Cada peso está relacionado com a “resistência” de uma única ligação sináptica biológica ([33]).

O modelo básico de um neurônio artificial com n entradas é apresentado na Figura 3.3.

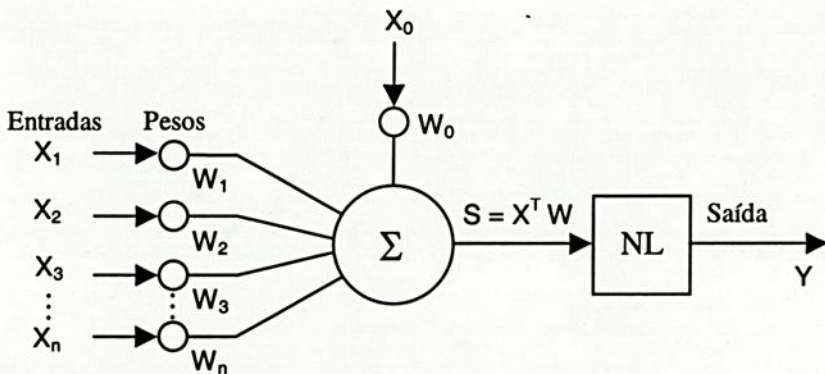


Figura 3.3: Modelo de um neurônio artificial.

sendo:

NL = Função Não-Linear denominada Função de ativação.

Cada entrada X é ponderada por um peso sináptico W . Se a soma ponderada das entradas for maior que o valor de w_0 , denominado *bias*, o neurônio é ativado provocando um pulso na saída, como mostra a Equação (3.2.1). Caso contrário, o neurônio não é ativado. O peso *bias* pondera uma entrada $x_0 = +1$, que atua como controle do nível de saída do neurônio ([29], [35]). A saída intermediária do neurônio é, então, dada por:

$$s_i = \sum_j^n w_{ij} x_j \quad (3.2.1)$$

A ativação do neurônio artificial é feita por uma função denominada função de ativação. A Figura 3.4 mostra alguns exemplos de funções de ativação ([17]).

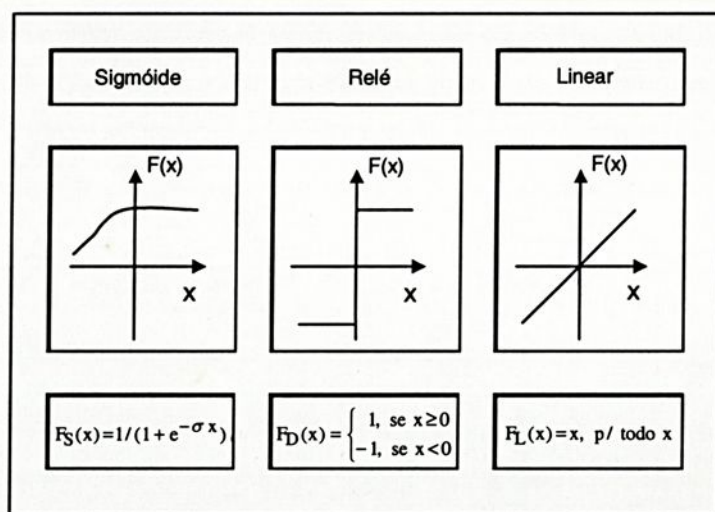


Figura 3.4: Funções de ativação.

As funções do tipo relé são apropriadas para sistemas binários, enquanto que as funções sigmoidais podem ser empregadas tanto para sistemas contínuos como binários. A função sigmoidal (*threshold*) é a mais utilizada, por ser contínua, apresentar uma derivada simples e ser a única a ter uma similaridade com o neurônio biológico ([11], [25], [29], [35]).

3.3 Estrutura da Rede Neural

As redes neurais artificiais podem ser definidas como conjuntos bem estruturados de unidades de processamento, interligadas por canais de comunicação, cada qual tendo um determinado peso correspondente a um valor numérico. Elas consistem de unidades de processamento (neurônios) conectadas entre si formando uma disposição estrutural de camadas e conexões entre as camadas ([18]).

Com relação a propagação dos dados, as redes neurais podem ser divididas em duas classes ([16]):

1. **Redes *Feedforward*:** Os dados fluem das unidades de entrada para as unidades de saída. São estritamente *feedforward*. O processamento dos dados pode se estender sobre múltiplas unidades (camadas), porém, as conexões *feedback* não estão presentes, isto é, as conexões são estendidas das unidades de saída para as unidades de entrada na mesma camada ou camadas anteriores ([16]).
2. **Redes Recorrentes:** Contém conexões *feedback*. Contrário à rede *feedforward*, as propriedades dinâmicas da rede são importantes. Em alguns casos, os valores de ativação das unidades submetem-se a um processo de relaxação tal que a rede evoluirá para um estado estável no qual estas ativações não variará mais ([16]).

Em outras aplicações, a variação dos valores de ativação dos neurônios de saída são significantes, de tal forma que o comportamento dinâmico constitui a saída da rede.

A rede recorrente desenvolve uma memória a longo prazo nos neurônios internos, enquanto que na rede *feedforward* a saída é exclusivamente determinada em função das entradas e dos valores dos pesos, o que caracteriza uma rede que não possui memória ([32], [33]).

3.4 Treinamento da Rede Neural



Dentre todas as características das redes neurais, uma das mais importantes é a habilidade de aprender e com isso melhorar seu desempenho. A rede é treinada de maneira que um conjunto de entrada produza o conjunto de saída desejada, ou pelo menos consistente ([21], [33]).

O treinamento torna-se completo a partir do momento em que aplicando um conjunto de entradas (vetores), seqüencialmente, os pesos da rede se ajustam de acordo com o procedimento predeterminado. Os pesos da rede convergem gradualmente a determinados valores, à medida que o vetor de entrada produz o vetor de saída desejado ([33]).

Quanto ao tipo de treinamento, pode-se classificá-lo em dois grupos distintos:

1. **Treinamento Supervisionado ou Treinamento Associativo:** A rede é treinada proporcionada pela combinação dos padrões de entrada e saída. Este treinamento exige a presença de um agente externo (tutor ou professor) na fase de aprendizado, que avalia e informa à rede sobre seu desempenho. O agente externo adquire conhecimento da rede, na forma de mapeamentos (padrões) de entrada-saída ([18]).
2. **Treinamento Não-Supervisionado (*Self-Organization*):** O conjunto de treinamento consiste exclusivamente de vetores de entrada. O treinamento não-supervisionado não possui tutor, ou seja, a rede é autônoma. Trabalha com os dados que são apresentados à rede.

Neste paradigma, o sistema é suposto como sendo capaz de descobrir, estatisticamente, características relevantes dos dados de entrada. Ao contrário do paradigma do treinamento supervisionado, não existe, *a priori*, um conjunto de categorias no qual os padrões serão classificados. Antes o sistema deve desenvolver sua representação própria ao estímulo de entrada ([16], [18], [33]).

3.5 Algoritmo *Back-Propagation*

Em 1974, Werbos apresentou a primeira concepção do algoritmo *Back-Propagation* ([34]). Porém, foi a partir de 1986 que Rumelhart, Hinton e Williams,



informados do trabalho de Werbos, tiveram uma idéia de como ajustar os pesos de uma rede neural das unidades de entrada para a unidade de saída. Os erros observados nas unidades da camada escondida são determinados pelo erro propagado na direção inversa das unidades da camada de saída. Por esta razão, o método é chamado algoritmo de aprendizagem B-P (*Back-Propagation*) ([16]). O B-P é um procedimento sistemático de adaptação de pesos de redes neurais baseado no método do gradiente descendente ([25], [33]).

O algoritmo *Back-Propagation* consiste na adaptação de pesos, tal que, o erro quadrático da rede é minimizado. A soma do erro quadrático instantâneo de cada neurônio alocado na última camada (saída da rede) é dada por ([35]):

$$\varepsilon^2 = \sum_{i=1}^{ns} \varepsilon_i^2 \quad (3.5.1)$$

sendo:

$$\varepsilon_i = d_i - y_i;$$

d_i = saída desejada do i -ésimo elemento da última camada da rede;

y_i = saída do i -ésimo elemento da última camada da rede;

ns = número de neurônios da última camada da rede.

Considerando-se o neurônio de índice i da rede, e utilizando-se o método do gradiente descendente ([25], [35]), o ajuste de pesos pode ser formulado como ([11], [35]):

$$V_i(h+1) = V_i(h) - \theta_i(h) \quad (3.5.2)$$

sendo:

$$\theta_i(h) = \gamma [\nabla_i(h)];$$

γ = parâmetro de controle da estabilidade ou taxa de treinamento;

h = representa o índice de iteração;

$\nabla_i(h)$ = gradiente do erro quadrático com relação aos pesos do neurônio i avaliado em h ;

V_i = vetor contendo os pesos do neurônio i

$$= [w_{0i} \ w_{1i} \ w_{2i} \ \dots \ w_{ni}]^T.$$

Na Equação (3.5.2), a direção adotada para minimizar a função objetivo do erro quadrático corresponde à direção contrária ao gradiente. O gradiente $\nabla_i(h)$ pode ser expresso por ([16]):

$$\nabla_i(h) = \frac{\partial \epsilon_i^2}{\partial V_i(h)} = 2 \epsilon_i \frac{\partial \epsilon_i}{\partial V_i(h)} \quad (3.5.3)$$

A função de ativação adotada é a função sigmóide definida por ([25],[35],[11]):

$$y_i = \frac{1 - e^{(-\lambda s_i)}}{1 + e^{(-\lambda s_i)}} \quad (3.5.4)$$

ou

$$y_i = \frac{1}{1 + e^{(-\lambda s_i)}} \quad (3.5.5)$$

sendo:

λ = constante que determina a inclinação da curva y_i .

Diferenciando a Equação (3.5.1) com relação ao vetor V_i , obtém-se:

$$\frac{\partial \epsilon_i}{\partial V_i} = - \frac{\partial y_i}{\partial V_i} = \frac{\partial y_i}{\partial s_i} \frac{\partial s_i}{\partial V_i} \quad (3.5.6)$$

Da Equação (3.5.6), resulta-se:

$$\frac{\partial \epsilon_i}{\partial V_i} = - sgm'(s_i) \frac{\partial s_i}{\partial V_i} \quad (3.5.7)$$

em que:

$sgm'(s_i) = \frac{\partial y_i}{\partial s_i}$, derivada parcial de y_i com relação a s_i .

Observa-se que:

$$\frac{\partial s_i}{\partial V_i} = X_i \quad (3.5.8)$$

sendo:

X_i $\triangleq$ vetor padrão;

$$= [x_{0i} \ x_{1i} \ x_{2i} \ \dots \ x_{ni}]^T.$$

Substituindo a Equação (3.5.8) na Equação (3.5.7), tem-se:

$$\frac{\partial \epsilon_i}{\partial V_i} = -sgm'(s_i) \ X_i \quad (3.5.9)$$

Inserindo-se a Equação (3.5.9) na Equação (3.5.3), resulta em:

$$\nabla_i(h) = -2 \ \epsilon_i \ sgm'(s_i) \ X_i \quad (3.5.10)$$

Utilizando o gradiente estimado no método do gradiente descendente, obtém-se o seguinte algoritmo ([35]):

$$V_i(h+1) = V_i(h) + 2 \ \gamma \ \epsilon_i \ sgm'(s_i) \ X_i \quad (3.5.11)$$

A derivada parcial da função sigmóide dada pela Equação (3.5.4) pode ser expressa por:

$$\begin{aligned} \frac{\partial y_i}{\partial s_i} &= \frac{\partial}{\partial s_i} \left\{ \frac{1 - e^{-\lambda s_i}}{1 + e^{-\lambda s_i}} \right\} \\ &= \frac{2 \lambda e^{-\lambda s_i}}{(1 + e^{-\lambda s_i})^2} \\ &= \frac{1}{2} \lambda \left\{ \frac{(1 + e^{-\lambda s_i})^2}{(1 + e^{-\lambda s_i})^2} - \frac{(1 - e^{-\lambda s_i})^2}{(1 + e^{-\lambda s_i})^2} \right\} \\ &= 0.5 \lambda (1 - y_i^2) \end{aligned} \quad (3.5.12)$$

E a derivada parcial da função sigmóide dada pela Equação (3.5.5) pode ser descrita por:

$$\begin{aligned}
 \frac{\partial y_i}{\partial s_i} &= \frac{\partial}{\partial s_i} \left\{ \frac{1}{1 + e^{-\lambda s_i}} \right\} \\
 &= \frac{1}{(1 + e^{-\lambda s_i})^2} (-\lambda e^{-\lambda s_i}) \\
 &= \frac{1}{(1 + e^{-\lambda s_i})} \frac{\lambda e^{-\lambda s_i}}{(1 + e^{-\lambda s_i})} \\
 &= \lambda y_i (1 - y_i)
 \end{aligned} \tag{3.5.13}$$

Utilizando a função sigmóide descrita pela Equação (3.5.4), pode-se escrever o algoritmo B-P da seguinte forma ([35]):

$$V_i(h+1) = V_i(h) + 2 \gamma \varepsilon_i (0.5 \lambda (1 - y_i^2)) X_i$$

e usando a função sigmóide dada pela Equação (3.5.5), obtém-se o algoritmo B-P conforme a equação abaixo ([35]):

$$V_i(h+1) = V_i(h) + 2 \gamma \varepsilon_i (\lambda y_i (1 - y_i)) X_i$$

O treinamento, via B-P, é iniciado pela apresentação de um padrão **X** à rede, o qual produzirá uma saída **Y**. Em seguida calcula-se o erro de cada saída (diferença entre o valor desejado e saída).

O próximo passo consiste na determinação do erro propagado no sentido inverso, através da rede associada à derivada parcial do erro quadrático de cada elemento com relação aos pesos e, finalmente, ajustando os pesos de cada elemento. Um novo padrão é apresentado – assim, o processo é repetido, para todos os padrões, até a convergência ($|\text{erro}| \leq \text{tolerância preestabelecida}$). Os pesos iniciais são normalmente adotados como números randômicos, visto que a iniciação nula quase sempre proporciona baixa convergência ([35]). Os pesos da rede são iniciados randomicamente considerando-se o intervalo $\{0,1\}$.



O treinamento via B-P pode ser efetuado, basicamente, de duas formas ([22]):

- **Convergência por Padrão:** Consiste em ajustar os pesos da rede (considerando-se todas as camadas), fazendo-se com que haja convergência para cada padrão, até que se complete o conjunto de padrões de entrada. O processo deverá ser repetido até a total convergência, isto é, o erro quadrático seja inferior a uma tolerância preestabelecida para os padrões considerados ([22]).
- **Uma Iteração por Padrão:** Este procedimento é idêntico ao primeiro, porém, fazendo-se somente uma iteração (ajuste de pesos) por padrão ([22]).

3.6 Algoritmo *Back-Propagation* com Momento

O algoritmo *Back-Propagation* é abordado na literatura sob várias formas com o propósito de torná-lo mais rápido computacionalmente. Uma formulação bastante interessante é o algoritmo B-P *com momento* ([35]).

O processo de aprendizagem exige com que a mudança nos pesos seja proporcional a $\partial \epsilon_i / \partial V_i$. O gradiente descendente requer que uma infinidade de passos sejam tomados. Considera-se por exemplo, uma taxa de treinamento grande o possível para que haja oscilação sem direção. Assim, um meio de evitar a oscilação para γ grande, é tomar uma variação no peso dependente da variação do peso passado acrescentando o termo momento. Este procedimento se constitui no algoritmo *Back-Propagation* com momento ([16]).

A adaptação dos pesos, neste caso, pode ser formulada da seguinte forma ([35]):

$$v_{ij}(h+1) = v_{ij}(h) + \Delta v_{ij}(h) \quad (3.6.1)$$

sendo:

$$\Delta v_{ij}(h) = 2 \gamma (1 - \eta) \beta_j x_i + \eta \Delta v_{ij}(h-1); \quad (3.6.2)$$



- v_{ij} = peso correspondente à interligação entre o i-ésimo e j-ésimo neurônio;
- γ = taxa de treinamento;
- η = constante momento ($0 \leq \eta < 1$) ([35]).

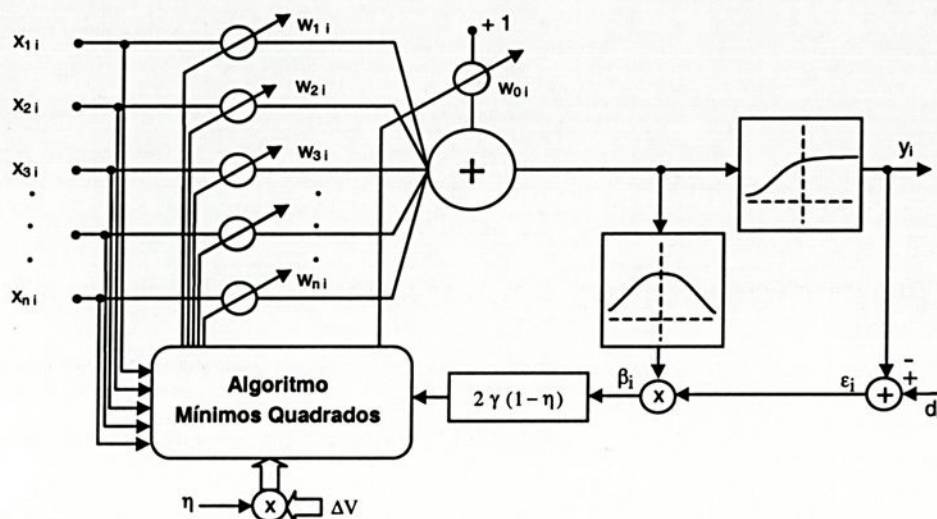


Figura 3.5: Treinamento de um neurônio situado na última camada.

Se o elemento j encontrar-se na última camada, como mostra a Figura 3.5, então:

$$\beta_j = \sigma_j \varepsilon_j \quad (3.6.3)$$

em que:

σ_j = derivada da função sigmoide dada pelas Equações (3.5.12) ou (3.5.13), respectivamente, com relação a s_j . No caso das Figuras 3.5 e 3.6, a derivada da função sigmoide é dada pela Equação (3.5.13).

Se o elemento j encontrar-se nas demais camadas, como mostra a Figura 3.6, tem-se:

$$\beta_j = \sigma_j \sum_{k \in R(j)} w_{jk} \beta_k \quad (3.6.4)$$

sendo:

$R(j)$ = conjunto dos índices dos elementos que se encontram na fileira seguinte à fileira do elemento j e que estão interligados ao elemento j .

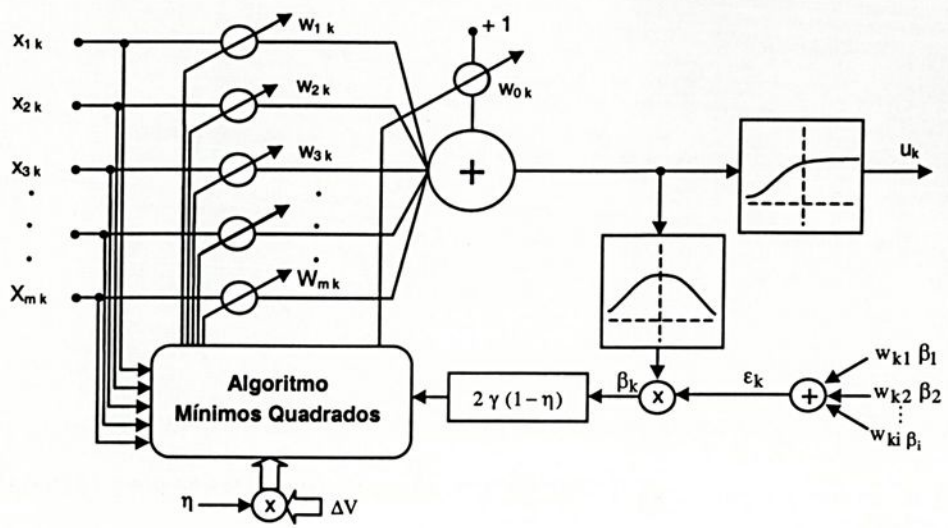


Figura 3.6: Treinamento de um neurônio situado nas demais camadas.

O parâmetro γ , que serve como controle de estabilidade do processo iterativo, é dependente de λ .

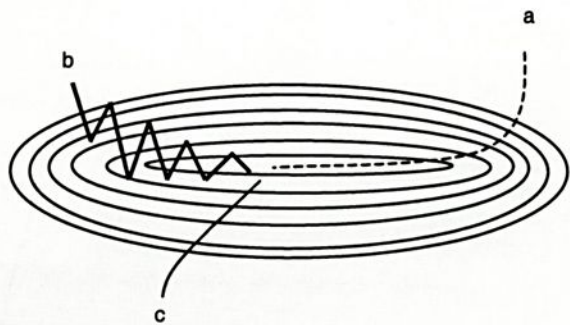


Figura 3.7: O algoritmo gradiente descendente no espaço de pesos, sendo o termo (a) taxa de treinamento pequena, o termo (b) taxa de treinamento grande (notar as oscilações) e o termo (c) taxa de treinamento grande mas com o termo momento já adicionado.

A Figura 3.7 mostra o significado do termo momento. Quando o termo momento não é utilizado, o algoritmo toma um longo tempo até o mínimo ser alcançado

com uma taxa de treinamento baixa, enquanto que para uma taxa de treinamento alta o mínimo nunca é alcançado em consequência das oscilações. Com o momento, o mínimo será obtido rapidamente, a princípio ([16]).

O termo momento promove movimentos em uma direção fixa, de maneira que se vários passos fossem tomados na mesma direção, o algoritmo “ganha velocidade” dando a ele capacidade (algumas vezes) de escapar dos mínimos locais e deslocar-se rapidamente sobre lugares planos e planaltos ([27]).

3.7 Vantagem do Algoritmo *Back-Propagation*

Dependendo da natureza da aplicação e da coerência dos padrões de dados internos, em geral, pode-se contar com uma rede que treine muito bem. As redes neurais são aplicadas a problemas de relações dinâmicas ou não-lineares. Elas suprem uma alternativa analítica que as técnicas convencionais limitam-se, muitas vezes, por suposições estritas de normalidade, linearidade, independência de variável, etc. ([12]).

A rede neural pode capturar muitos tipos de informações que permitem ao usuário, com maior rapidez e facilidade, explicar os problemas que, por outro lado, podem ter sido difíceis ou impossíveis de serem resolvidos ([12]). Dentre tantas vantagens da utilização das redes neurais artificiais, pode-se citar a inferência de múltiplas variáveis, a grande tolerância a falhas, a modelagem direta do problema e o paralelismo inerente ([18]).

3.8 Desvantagem do Algoritmo *Back-Propagation*

Apesar do sucesso aparente do algoritmo de aprendizado *Back-Propagation*, existem alguns problemas que afetam o seu desempenho. A maior parte dos problemas é que o processo de treinamento é longo, e pode causar um resultado de momento e uma taxa de treinamento não ótimos. Os fracassos do treinamento geralmente surgem de duas fontes: paralisia da rede e mínimo local.



1. **Paralisia da Rede:** Com o treinamento da rede, os pesos podem ser ajustados por valores muito grandes. A entrada total de uma unidade (camada) escondida ou unidade de saída pode, portanto, alcançar valores muito altos (ou positivo ou negativo) e, em consequência da função de ativação sigmoideal, a unidade terá uma ativação muito próxima de zero ou muito próxima de um. Portanto, os pesos ajustados estarão próximos de zero, e o processo de treinamento pode chegar a uma paralisação (ou efeito de paralisia) ([16]).
2. **Mínimos Locais:** O erro de superfície de uma rede complexa é cheio de “morros” e “vales”. Em consequência do gradiente descendente, a rede pode ficar confinada em um mínimo local quando existir um mínimo próximo muito profundo. Os métodos probabilísticos podem ajudar a evitar esta armadilha, mas eles tendem a ser lentos. Outra possibilidade sugerida é aumentar o número de unidades escondidas, que funcionará por causa da dimensionalidade maior do espaço de erro. Isto ocorre porque existem alguns limites superiores do número de unidades escondidas que, quando excedida, novamente resultam no sistema sendo apanhado no mínimo local ([16]).

Outra desvantagem da utilização das redes neurais artificiais é o fato delas trabalharem com um número reduzido (centenas) de neurônios artificiais, enquanto as redes neurais biológicas trabalham com milhões de neurônios ([18]).

3.9 Conclusão

Neste capítulo foram apresentados os conceitos de redes neurais artificiais e os modelos de neurônio biológico e artificial, identificando em cada um deles os principais componentes funcionais. Foram mostradas, também, as estruturas das redes neurais, desde as subdivisões dos modelos de conexões entre as unidades de processamento e a propagação dos dados, até as distinções dos tipos de treinamento, no qual desenvolve a característica mais importante das redes neurais. O estudo principal deste capítulo foi o enfoque do algoritmo *Back-Propagation*, versão com momento, que é o instrumento fundamental utilizado na pesquisa deste trabalho. Para concluir o capítulo, foram abordados os problemas encontrados no treinamento do algoritmo *Back-Propagation* e possíveis maneiras de contorná-los, tornando mais viável a compreensão do algoritmo.



Neste capítulo, serão apresentados os conceitos sobre lógica nebulosa, com o objetivo de desenvolver um mecanismo de adaptação de pesos da rede neural *Back-Propagation* para previsão de cargas elétricas. Serão apresentadas as definições de conjuntos nebulosos, funções de pertinência e o conceito fundamental da lógica nebulosa, o controle nebuloso, no qual engloba basicamente três fases: nebulização, inferência e desnebulização, completando assim a chamada ação de controle.

4.1 Conjuntos Nebulosos

A lógica nebulosa foi estruturada em 1965 pelo Dr. Lofti A. Zadeh da Universidade da Califórnia para tratar e representar incertezas ([18]). Ela torna-se importante pelo fato de que o mundo não é constituído por fatos absolutamente verdadeiros ou falsos e permite representar valores de pertinência (grau de verdade) intermediários entre os valores verdadeiros e falsos da lógica clássica ([18]).

A lógica clássica é aplicada às informações consideradas completamente verdadeiras, cujo valor de verdade é igual a 1, ou informações completamente falsas, cujo valor de verdade é igual a 0, enquanto que a lógica nebulosa tem a vantagem de poder ser aplicada às informações que não são completamente verdadeiras ou falsas, podendo variar entre 0 e 1 ([18]).

A lógica nebulosa é, basicamente, uma lógica de multi-valores que permite valores intermediários serem definidos entre avaliações convencionais como *sim/não*, *verdadeiro/falso*, são as chamadas variáveis lingüísticas, objetos ou palavras lingüísticas, que possuem uma representação melhor que os números ([14]). As variáveis nebulosas são



adjetivos que modificam a variável, de tal forma que noções como *bastante morno* ou *bastante frio* sejam formuladas matematicamente e processadas por computador ([07], [14]).

Um conjunto nebuloso A em X , coleção de objetos, é definido como sendo o conjunto de pares ordenados:

$$A = \{(x, \mu_A(x)) \mid x \in X\} \quad (4.1.1)$$

sendo:

$\mu_A(x)$ = valor da função de pertinência do conjunto nebuloso A correspondente ao elemento x .

Supondo-se que A e B são dois conjuntos nebulosos com funções de pertinência designadas, respectivamente, por $\mu_A(x)$ e $\mu_B(x)$ definidas pela Figura 4.1, pode-se introduzir os operadores básicos mais importantes dos conjuntos nebulosos definidos por Zadeh da seguinte maneira ([03], [04], [07], [15], [30]):

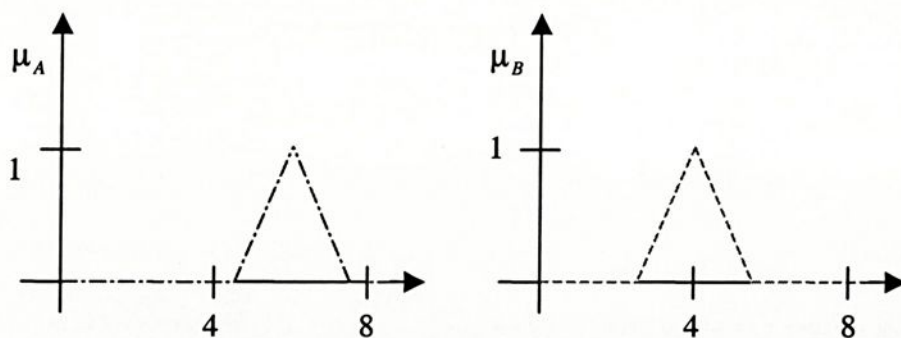


Figura 4.1: Conjuntos nebulosos.

- a) *Operador AND ou interseção de dois conjuntos*: A função de pertinência da interseção destes dois conjuntos nebulosos ($C = A \cap B$) é definida por:

$$\mu_C(x) = \min\{\mu_A(x), \mu_B(x)\}, x \in X \quad (4.1.2)$$

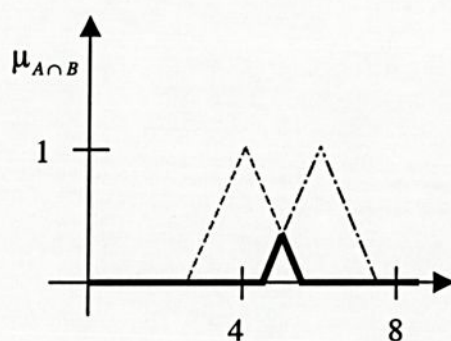


Figura 4.2: Operador AND.

Na Figura 4.2, a linha cheia define a interseção dos dois conjuntos nebulosos A e B ([06], [07]).

b) *Operador OR ou união entre dois conjuntos*: A função de pertinência da união destes conjuntos nebulosos ($D = A \cup B$) é definida por:

$$\mu_D(x) = \max\{\mu_A(x), \mu_B(x)\}, x \in X \quad (4.1.3)$$

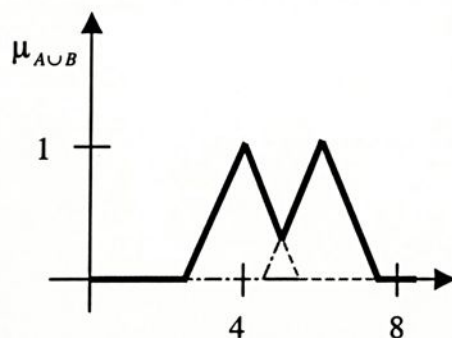


Figura 4.3: Operador OR.

Neste caso, a linha cheia mostrada na Figura 4.3 define a união dos dois conjuntos nebulosos A e B ([06], [07]).

c) *Operador NOT ou o complemento de um conjunto nebuloso*: A função de pertinência do complemento de A, A' é definida por:

$$\mu_{A'}(x) = 1 - \mu_A(x), x \in X \quad (4.1.4)$$

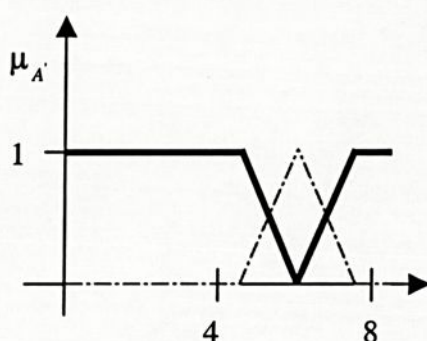


Figura 4.4: Operador NOT.

Na Figura 4.4 o complemento do conjunto A é dado pela linha cheia ([06], [07]).

4.2 Funções de Pertinência

A função de pertinência é uma representação gráfica da magnitude de participação de cada entrada. Ela associa um peso a cada entrada processada, definindo uma superposição funcional entre as entradas e, finalmente, determinando uma resposta de saída. As regras utilizam os valores de pertinência da entrada como fator peso para determinar sua influência na saída conjuntos nebulosos da conclusão final de saída. Uma vez as funções concluídas, pesadas e combinadas, elas são desnebulizadas em uma saída real (*crisp*) que conduz o sistema ([14]).

As funções de pertinência nebulosas podem ter diferentes formas associadas com cada entrada e resposta de saída. A função de pertinência triangular é a mais comum, mas as demais funções de pertinência (trapezoidal, gaussiana e exponencial) podem ser usadas de acordo com a preferência ou experiência do projetista ([13], [14]).

A Figura 4.5 mostra as características da função de pertinência, na qual a altura ou magnitude é normalizada em 1. O grau de pertinência é determinado pela projeção vertical do parâmetro de entrada do eixo horizontal no limite mais alto da função de pertinência, o qual tem valores compreendidos entre 0 e 1 ([08], [14]).

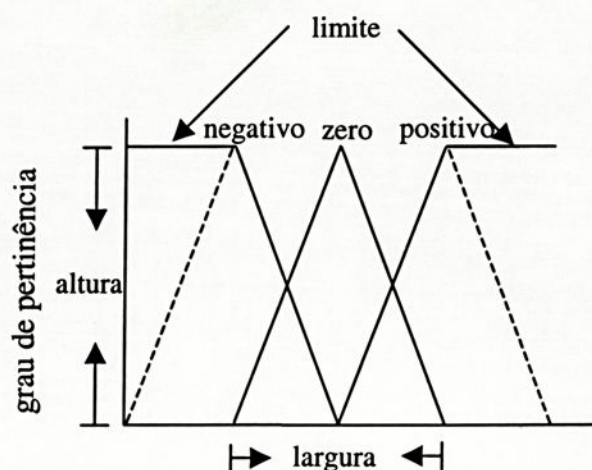


Figura 4.5: Funções de pertinência.

4.3 Controle Nebuloso

Os problemas complexos normalmente exigem soluções complexas, que por sua vez, necessitam de tempo e esforço proporcional ao nível de complexidade. Os sistemas nebulosos utilizam soluções simples para resolver estes tipos de problemas, pois em vez de controlar e/ou modelar um processo através do uso de relações complicadas entre as variáveis, o sistema nebuloso leva em conta o controle e/ou modelagem do processo através do uso de um conjunto simples de vários tipos de regras de senso comum. O controle nebuloso utiliza estas regras baseado em um procedimento proposto nos anos 70 por Ebrahim Mamdani ([05], [31]).

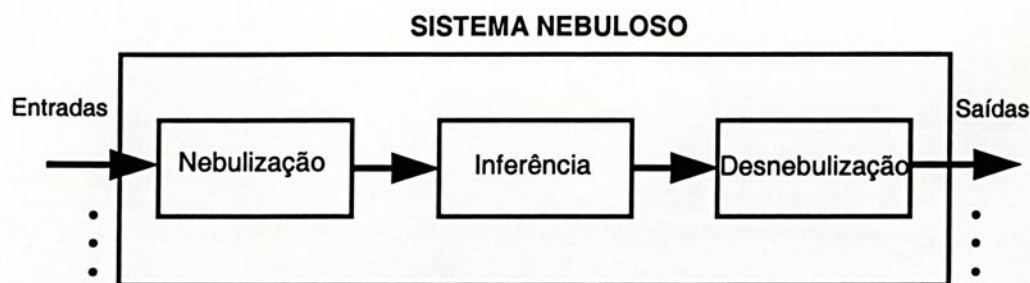


Figura 4.6: Diagrama de bloco de um sistema nebuloso.

Esse mecanismo é constituído, basicamente, por três partes como mostra a Figura 4.6: *nebulização* que converte variáveis reais (*crisp*) em variáveis lingüísticas;

inferência que consiste na manipulação de base de regras utilizando declarações *if-then* e, ainda, operações nebulosas, como definidas anteriormente (equações (4.1.2)–(4.1.4)) e *desnebulização* que converte o resultado obtido (variáveis lingüísticas) em variáveis reais (*crisp*), as quais constituem a ação de controle ([03], [13]).

4.3.1 Nebulização

A nebulização é um processo que decompõe um sistema de entrada e/ou saída em um ou mais conjuntos nebulosos.

Cada conjunto nebuloso cobre uma região particular de um sistema de entradas ou valores de saída e, através da função de pertinência, produz um grau de pertinência com valores compreendidos entre 0 e 1 ([31]). O resultado dos conjuntos nebulosos representa o grau para o qual um sistema de entrada ou um valor de saída é um membro daquele conjunto nebuloso particular.

Na Figura 4.7 mostra-se que uma entrada de temperatura variável pode ser nebulizada em vários conjuntos nebulosos chamados “fria”, “normal” ou “quente”.

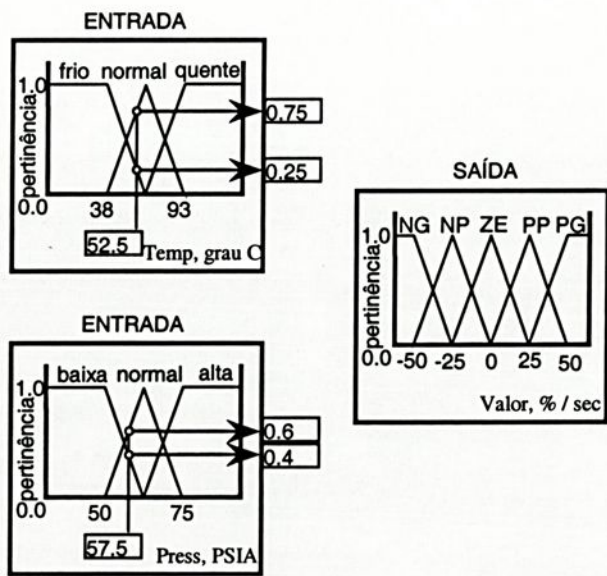


Figura 4.7: Sistema Nebuloso.

em que:



NG = Negativo Grande

NP = Negativo Pequeno

ZE = Zero

PP = Positivo Pequeno

PG = Positivo Grande

Uma entrada de temperatura é considerada “*normal*” quando está entre 38 e 93 graus centígrados, a função de pertinência “*normal*” deve ser uma curva que gera um valor entre 0 e 1 quando a temperatura está entre 38 e 93 graus centígrados. Observa-se que a superposição entre os limites dos conjuntos nebulosos é necessária para prover uma operação suave do sistema. O processo de nebulização permite que as entradas e saídas do sistema sejam expressas em termos lingüísticos de maneira que as regras possam ser aplicadas de uma maneira simples para expressar um sistema complexo ([31]).

4.3.2 Inferência

Após as entradas e saídas do sistema terem sido decompostas em seus respectivos grupos de conjuntos nebulosos, um conjunto de regras associada a estes conjuntos nebulosos devem ser definidas para reger o comportamento do sistema ([31]).

As regras nebulosas são uma série de declarações *if-then* que consistem de uma condição e uma ação definidas como sendo uma função da entrada dos conjuntos nebulosos e uma função da saída dos conjuntos nebulosos, respectivamente ([04], [05], [31]):

IF CONDIÇÃO THEN AÇÃO

sendo:

CONDIÇÃO = f (entrada de conjuntos nebulosos)

AÇÃO = f (saída de conjuntos nebulosos)

f = função



Normalmente, um sistema nebuloso exige mais de uma regra para descrever completamente toda a ação necessária. Na prática, um conjunto de regras que abrange todas combinações de todos os conjuntos nebulosos possíveis para todas as entradas do sistema, pode ser indicado por uma matriz chamada *Memória Associativa Nebulosa* (MAN). Na Figura 4.8 duas entradas do sistema (temperatura e pressão) com três conjuntos nebulosos cada uma, forma uma base com nove regras na MAN ([31]). Desta forma, as regras ficam armazenadas no MAN como se fossem uma matriz, assim:

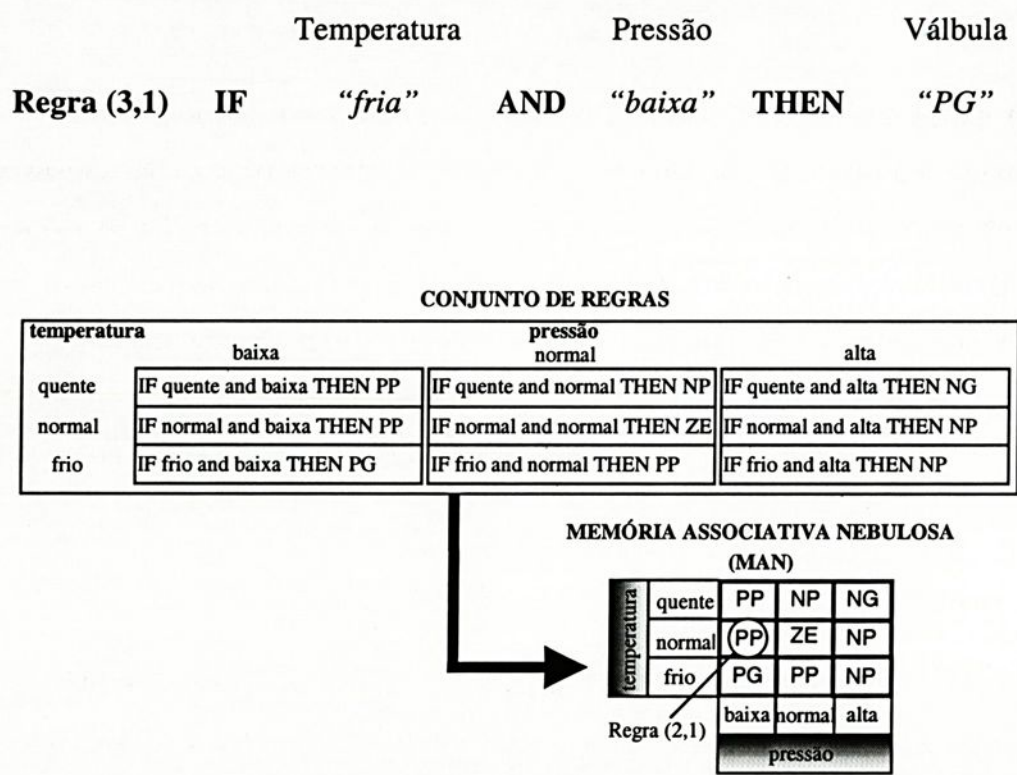


Figura 4.8: Conjunto de regras e MAN.

Cada regra é avaliada para surtir uma ação baseada na firmeza da condição. A regra é dita ser “fraca” quando um resultado não zero retorna sobre a avaliação da regra.

O MAN é uma matriz que representa todas as possíveis combinações de todos conjuntos nebulosos de ambas entradas nebulosas. Os conjuntos nebulosos são colocados dentro desta matriz de forma a apresentar o melhor ajuste do tipo de controle e/ou modelagem que este sistema nebuloso está designado a realizar. Na prática, este arranjo da



saída dos conjuntos nebulosos dentro do MAN dá ao usuário uma capacidade de ajuste poderoso ([31]).

4.3.3 Desnebulização

Desnebulização é um processo que converte um conjunto nebuloso ou um número nebuloso em um valor ou número real (*crisp*). Este processo é utilizado na modelagem e no controle da lógica nebulosa para converter as saídas nebulosas dos sistemas em valores reais ([36]).

Existem numerosas técnicas para desnebulizar um conjunto nebuloso, dentre elas as três principais encontradas na literatura são:

1. **critério *max*** : Encontra o ponto no qual a função de pertinência é um máximo ([13]).
2. **média do máximo** : Toma a média dos pontos em que a função de pertinência está para um valor máximo ([13]).
3. **centro de área** : Encontra o centro de gravidade da solução dos conjuntos nebulosos ([03], [13]).

Dentre eles, o método mais comum é o método de centro de área também conhecido como centróide ([03], [30]). O valor de saída é calculado pela somatório do produto da pertinência do conjunto de entrada, para cada entrada, pelo valor de saída determinado através de um conjunto de regras.

Este resultado é dividido pelo somatório da pertinência do conjunto de entrada para cada entrada, resultando assim o valor da ação de controle ([31]). Desta forma, para um conjunto nebuloso discreto tem-se ([03], [30]):

$$u = \frac{\sum_{i=1}^n \mu_i \delta_i}{\sum_{i=1}^n \mu_i} \quad (4.3.3.1)$$



sendo:

δ_i = valor do conjunto que tem um valor de pertinência μ_i ;

n = número de regras nebulosas.

Para conjuntos nebulosos contínuos o cálculo do centróide é a divisão do momento dos conjuntos nebulosos pela sua área. Considerando $f(x)$ um conjunto nebuloso, então ([04], [15]):

$$u = \frac{\int x f(x) dx}{\int f(x) dx} \quad (4.3.3.2)$$

sendo:

$f(x)$ = conjunto nebuloso. Os limites da integração correspondem ao universo inteiro de X ([04]).

O valor u calculado pela Equação (4.3.3.1) ou (4.3.3.2) corresponde a projeção do centro de inércia da figura definida pelo conjunto de regras sobre o eixo da variável de controle.

4.4 Conclusão

Foram abordados, neste capítulo, os conceitos sobre conjuntos nebulosos, ilustrando de forma geral, as operações básicas exercidas sobre estes conjuntos. Também abordaram-se função de pertinência e controle nebuloso, que é basicamente a parte fundamental da lógica nebulosa. Dentro do controle nebuloso foram enfocados os seguintes temas: nebulização, inferência e desnebulização, sendo que estas três fases formam a ação de controle da lógica nebulosa, que é a parte principal para montagem do controlador nebuloso utilizado como controlador da taxa de aprendizado da rede neural *Back-Propagation* a fim de reduzir o tempo de execução do método convencional.

5

Previsão de Carga via Redes Neurais com Controlador Nebuloso

Neste capítulo, será apresentada a rede neural controlada, *Back-Propagation* com controlador nebuloso, que é o algoritmo fundamental deste trabalho utilizado para realizar a previsão de cargas elétricas.

5.1 Rede Neural Nebulosa

A idéia básica da metodologia consiste na determinação do estado do sistema definido como sendo o erro global (ϵ_g) e a variação do erro global ($\Delta\epsilon_g$). O objetivo é obter uma estrutura de controle que consiga com que o erro tenda a zero em um número mínimo de iterações, quando comparado ao procedimento convencional, neste caso o algoritmo *Back-Propagation*.

O controle é formulado utilizando os conceitos de lógica nebulosa ([30]). O erro global (ϵ_g) e sua variação ($\Delta\epsilon_g$) são os componentes do estado do sistema, e $\Delta\gamma$ é a ação de controle que deve ser exercida sobre o sistema.

O erro global é definido como sendo:

$$\epsilon_g = \sum_{j=1}^{np} \sum_{i=1}^{ns} \epsilon_i^2 \quad (5.1.1)$$

em que:

ϵ_g = erro global da rede neural;

np = número de vetores padrão da rede;



ns = número de neurônios da última camada da rede.

O erro global corresponde ao cálculo de erros de todas as saídas (neurônios de saída), considerando-se todos os vetores padrões da rede. O treinamento deve ser executado utilizando uma iteração por padrão e, então, o erro global é calculado em cada iteração e ajustado o parâmetro γ , através de um acréscimo $\Delta\gamma$ determinado via lógica nebulosa.

O estado do sistema e ação de controle são definidos como ([19]):

$$E^q = [\varepsilon g^q \ \Delta\varepsilon g^q]^T \quad (5.1.2)$$

$$u^q = \Delta\gamma^q \quad (5.1.3)$$

sendo:

q = o índice que indica a iteração corrente.

Para um padrão X de entrada da rede muito grande, o εg e a $\Delta\varepsilon g$ podem entrar em saturação. Então, o controle adaptativo é feito através do uso de uma função exponencial decrescente aplicada à resposta do controlador nebuloso. Desta forma, o controle adaptativo é dado por ([20]):

$$\Delta\gamma^q = \exp(-\alpha q) \Delta\psi^q \quad (5.1.4)$$

em que:

α = um número positivo arbitrário;

$\Delta\psi^q$ = acréscimo procedente do controlador nebuloso no instante q .

Este parâmetro será utilizado para ajustar o conjunto de pesos da rede referente à iteração subsequente. O processo deverá ser repetido até que o treinamento seja concluído. Trata-se de um procedimento muito simples, cujo sistema de controle requer um esforço



adicional bastante reduzido, tendo em vista que o controlador possui duas variáveis de entrada e uma única de saída ([03]).

Cada variável de estado deverá ser representada entre 3 e 7 conjuntos nebulosos. A variável de controle também deverá ser representada com o mesmo número de conjuntos nebulosos.

A variável ϵ_g deverá ser normalizada considerando-se como fator de escala, o primeiro erro global gerado pela rede, ou seja, de índice $q = 0$. Com esta representação, o intervalo de variação de ϵ_g deverá estar compreendido entre 0 e +1. Se a heurística de adaptação estiver devidamente sintonizada, a convergência do processo deverá ser exponencial decrescente ([20]).

A variável $\Delta \epsilon_g$ deverá variar entre -1 e +1. Se o processo de convergência for exponencial decrescente os valores de $\Delta \epsilon_g$ deverão ser sempre negativos. Neste caso, embora a escala de $\Delta \epsilon_g$ esteja entre -1 e +1, deve-se empregar, no conjunto de regras, um ajuste fino entre -1 e 0. No outro intervalo (0,+1], o ajuste poderá ser mais relaxado.

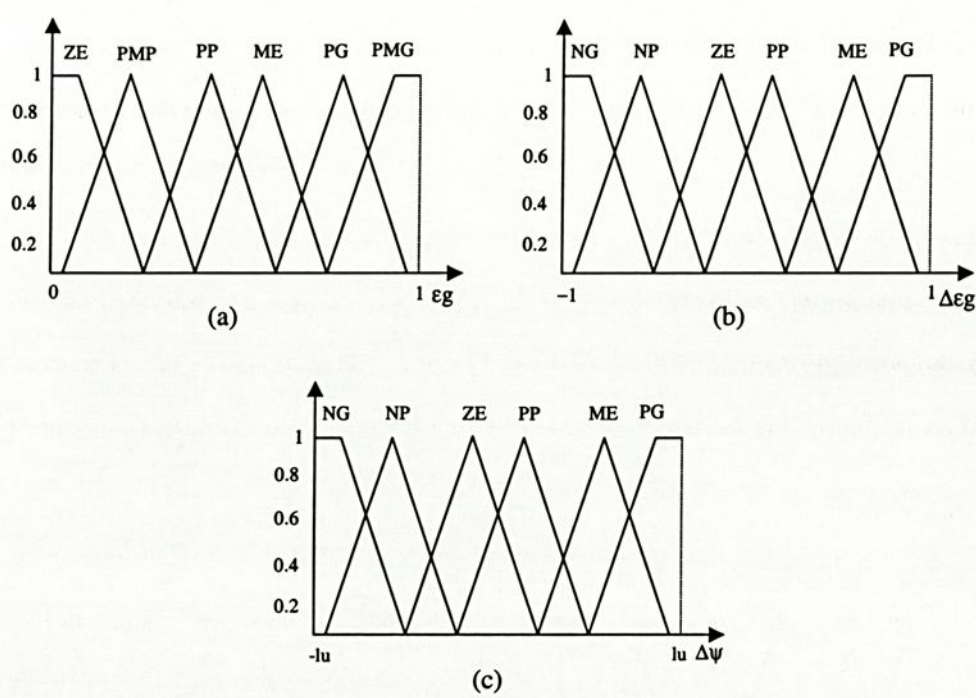


Figura 5.1: Funções de pertinência para as variáveis ϵ_g , $\Delta \epsilon_g$ e $\Delta \psi$ (ação de controle).

sendo:

NG = Negativo Grande;

NP = Negativo Pequeno;

ZE = Próximo de Zero;

PMP = Positivo Muito Pequeno;

PP = Positivo Pequeno;

ME = Médio;

PG = Positivo Grande;

PMG = Positivo Muito Grande;

lu = limite de variação de $\Delta\psi$.

As funções de pertinência dos conjuntos nebulosos para várias variáveis envolvidas no treinamento da rede neural com *Back-Propagation*, via controlador nebuloso, são mostradas na Figura 5.1, sendo que o item (a) representa as funções de pertinência para o erro global ε_g , o item (b) as funções de pertinência para a variação do erro global $\Delta\varepsilon_g$ e o item (c) as funções de pertinência para a ação de controle $\Delta\psi$.

No controlador nebuloso, as regras são codificadas na forma de uma tabela de decisões. Cada entrada representa o valor da variável nebulosa $\Delta\psi$ dados os valores do erro global ε_g e a variação do erro global $\Delta\varepsilon_g$. O parâmetro γ (taxa de treinamento) deve ser arbitrado em função de λ (inclinação da curva). Da mesma forma, as variações de γ também devem seguir o mesmo procedimento. Deste modo, o parâmetro (lu) deve ser ajustado para atender as especificações de cada problema em função de λ .

Na Tabela 5.1 é apresentado o conjunto de regras nebulosas, totalizando 30 regras. O número de regras poderá ser aumentado visando melhorar o desempenho da rede no treinamento.



Tabela 5.1: Regras do controlador nebuloso.

$\Delta \epsilon_g$	ϵ_g					
	ZE	PMP	PP	ME	PG	PMG
NG	ZE	ZE	NP	PG		PG
NP	NG	ZE	ME	NG	ME	
ZE	PP	NG	NG		PP	NP
PP	PG	ME		PG	NP	ZE
ME	ZE		ZE	PP	ZE	NG
PG		ZE	NG	ME	ME	ZE

5.2 Conclusão

Neste capítulo, foi mostrado todo o desenvolvimento da rede neural *Back-Propagation* com um controlador adaptativo da taxa de treinamento feito através de um controlador nebuloso. É importante salientar que o algoritmo apresentado neste capítulo ressalta contribuições tais como: melhoria do algoritmo *Back-Propagation* e uma proposta de controlador nebuloso original (um conjunto de regras e o uso de uma função exponencial decrescente aplicada à resposta do controlador).



6

Aplicação – Previsão de Carga

Neste capítulo, serão apresentados os principais resultados obtidos através do algoritmo neuro-nebuloso desenvolvido visando a resolução do problema de previsão de carga a curto prazo (24 horas à frente). Para a fase de treinamento da rede neural, serão utilizados os dados históricos de uma companhia do setor elétrico brasileiro. Com o objetivo de testar a eficiência da metodologia da rede neuro-nebulosa proposta, os resultados obtidos serão comparados com os obtidos com uma rede neural treinada, via *Back-Propagation* convencional. A precisão deverá ser verificada calculando-se o erro médio e erro máximo ocorrido, considerando-se o horizonte de previsão. O erro será calculado tomando-se a diferença entre a série prevista e os valores conhecidos das cargas (valores de referência).

6.1 Sistema Previsor de Cargas Elétricas

No desenvolvimento do trabalho existem características importantes que devem ser comentadas e analisadas com a finalidade de compreender o desempenho do algoritmo proposto com relação a previsão de cargas elétricas. Uma delas é verificar qual deve ser a função sigmóide utilizada durante o processo de treinamento da rede. Para isso deve-se analisar a saída desejada da rede. Se a saída está compreendida entre -1 e $+1$, a função sigmóide é dada pela Equação (3.5.4) e se a saída possuir grandezas positivas, a função sigmóide utilizada será dada pela Equação (3.5.5), sendo que a saída terá valores compreendidos entre 0 e $+1$.

Como o problema a ser abordado corresponde à previsão de cargas elétricas, a função não-linear adotada é a função sigmóide definida pela Equação (3.5.5) ([11], [25], [35]).



Para trabalhar com modelos binários deve-se adotar valores grandes para o parâmetro λ ([35]), enquanto que para modelos contínuos ([22], [23]) é recomendável adotar valores de menor magnitude ([22]). A derivada σ tem forte semelhança com a curva de distribuição normal (de Gauss). Constitui-se de um *corpo central* e suas laterais designadas, como forma de simplificação, *caudas esquerda e direita* ([22]). Nota-se que a diminuição de λ produz uma suavização na função sigmóide y , enquanto que sua derivada σ tende a diminuir a amplitude da curva e alongar as caudas.

No algoritmo *Back-Propagation*, a adaptação de pesos, que concorrem a cada neurônio, é efetuada utilizando-se, basicamente, o erro propagado no sentido inverso, multiplicado por σ e pela entrada no referido neurônio. Deste modo, os pesos têm efetivo ajuste somente para valores de s situados no corpo central da função σ . À medida que os pesos se tornarem significativos, há desaceleração dos ajustes dos mesmos, tendendo para a completa paralisia, justamente onde se encontram os terminais das caudas tanto direita como esquerda ([22]).

Em vista disto, no caso específico do problema de determinação de tempos críticos, utiliza-se a função sigmóide com λ relativamente pequeno, proporcionando caudas mais longas. Isto permitirá a escolha de pesos da rede de forma menos restritiva, se comparada às adotadas na bibliografia, que de certo modo diminui a possibilidade da ocorrência de paralisia e aumenta a velocidade de convergência do algoritmo B-P ([22]).

Porém, com a diminuição de λ , a magnitude dos pesos tende a crescer. Por conseguinte, a escolha de λ deve ser feita com critério, levando-se em conta a experiência com treinamento.

Nos resultados apresentados, por conveniência, o parâmetro γ (taxa de treinamento) é redefinido do seguinte modo ([19]):

$$\gamma = (2 \gamma^* / \lambda) \quad (\text{para sigmóide definida pela Equação (3.5.4)}) \quad (6.1.1)$$

$$\gamma = (\gamma^* / \lambda) \quad (\text{para sigmóide definida pela Equação (3.5.5)}) \quad (6.1.2)$$



Substituindo-se as expressões de γ (Equação (6.1.1) e (6.1.2)) na Equação (3.6.2), será “anulada” a dependência da amplitude de σ com relação a λ . A amplitude de σ será mantida constante para qualquer valor de λ . Esta alternativa torna-se mais conveniente tendo em vista que λ atuará somente nas caudas esquerda e direita de σ . Assim, arbitrando-se um valor inicial para γ^* , este parâmetro será ajustado via controlador nebuloso, substituindo-se $\Delta\gamma$ por $\Delta\gamma^*$ ([19]).

A previsão de carga a curto-prazo (previsão diária) é realizada de acordo com o seguinte esquema: implementação de recorrência na saída em um determinado instante será utilizada como entrada no instante subsequente. São considerados os dados históricos horários dentro de um intervalo preestabelecido, *e.g.*, mensal. A entrada da rede, para uma determinada hora h , é definida como sendo os valores da carga, extraídos dos dados históricos em quatro instantes (o valor corrente, uma, duas e três horas anteriores), a temperatura, etc., e os dados referentes ao tempo (mês, dia da semana, feriado e hora, etc.). A saída da rede corresponde ao valor da carga referente à hora $(h+1)$.

O conjunto que compõe a entrada / saída é definido considerando esta estratégia até completar todo o intervalo de tempo da série temporal. Este esquema pode ser modificado visando obter melhores resultados, através da introdução de outras variáveis (dias nublados, etc.). Assim, o vetor de entrada e a saída, respectivamente, são definidos da seguinte forma ([20]):

$$X(h) = [t^T L(h-3) L(h-2) L(h-1) L(h)]^T, \quad X \in R^m \quad (6.1.3)$$

$$Y(h) = [L(h+1)], \quad Y \in R^1 \quad (6.1.4)$$

sendo:

m = dimensão do vetor X ;

$L(h-p)$ = valor da carga p horas anterior à hora corrente h ;

$L(h+1)$ = valor da carga elétrica correspondente à hora subsequente a hora corrente h ;

t = vetor de tempo referente aos dados históricos (mês, dia da semana, feriado, hora, etc) representados de modo similar ao código binário $(-1,+1)$.



A escolha desta representação binária é preferível em relação à representação (0,+1), tendo em vista que o componente "0" de entrada à rede não proporciona modificação nos pesos. Deste modo, a representação (-1,+1) produz uma convergência mais rápida, conseqüentemente sendo mais eficiente.

As cargas elétricas $L(h-3)$, ..., $L(h-1)$ representam o laço de realimentação, com atraso, da saída. Deste modo, esta rede é do tipo recorrente.

Os dados para compor o treinamento, empregados neste trabalho, são (para cada vetor): os dados do tempo (dia do mês, dia da semana (se é feriado ou não) e a hora do dia), as cargas horárias correntes e valores de cargas considerando-se três horas anteriores. A carga futura (uma hora à frente) constitui a saída da rede. Não foram considerados os dados de temperatura em conseqüência da não disponibilidade dos mesmos. Considerando-se a representação binária, o vetor t terá dimensão 20 que, juntamente com os dados da carga, serão 30 componentes ($m = 30$). Os dados históricos foram obtidos de cargas elétricas de uma companhia do setor elétrico brasileiro. Estes dados contêm valores das cargas horárias do ano de 1998, estando eles relacionados a dias atípicos (feriados), dias especiais (sábados e domingos) e dias de uma semana típica.

Neste experimento são considerados três aplicações, sendo que o banco de dados possui, respectivamente, dados históricos de 21 (Apêndice A), 42 (Apêndice B) e 84 dias (Apêndice C). De um modo geral, utiliza-se um vetor de entrada de dimensão 12, no qual tem-se os dados do dia da semana e da hora do dia codificados na representação binária, mais a carga horária corrente e os valores de cargas considerando-se três horas anteriores.

A primeira aplicação composta por dados históricos de 21 dias está compreendida entre o dia 8 de julho de 1998 e o dia 28 de julho de 1998, num total de 504 vetores de entrada. Para a segunda aplicação, composta por 42 dias, os dados históricos estão compreendidos entre o dia 17 de junho de 1998 e o dia 28 de julho de 1998 resultando num banco de entradas formado por 1008 vetores. E, finalmente, a terceira aplicação que contém 84 dias, compreendidos entre o dia 4 de maio de 1998 e o dia 27 de julho de 1998, dando origem ao vetor de entrada composto por 2016 vetores. É importante ressaltar que o dia 11 de junho de 1998 foi excluído do banco de dados, pelo fato de ser um dia atípico (feriado), cujos dados contêm imprecisões grosseiras que se utilizado interferem consideravelmente nos resultados da previsão.



Os valores das cargas são, inicialmente, normalizados uma vez que a saída somente pode variar entre 0 e +1. A série prevista será, então, finalmente, convertida aos valores reais, ou seja, multiplicados pelo valor da base preestabelecida.

Em cada aplicação, utiliza-se duas redes neurais, rede neural 1 e rede neural 2, que possuem os mesmos parâmetros da rede (número de camadas, número de neurônios por camadas, tolerância, taxa de treinamento, momento e inclinação da função sigmóide), diferenciadas apenas no valor do parâmetro α (decaimento exponencial, imposto pelo controlador nebuloso para evitar a saturação da rede neural), os quais produzem resultados diferentes.

Tabela 6.1: Especificação da rede neural referente a aplicação 1.

Item	Valor	
	Rede Neural 1	Rede Neural 2
Número de vetores padrão	504	504
Número de camadas	3	3
Número de neurônios por camada	12-30-1	12-30-1
Tolerância	0,02	0,02
Taxa de treinamento γ^*	4,5	4,5
Termo momento η	0,9	0,9
Inclinação da função sigmóide λ	0,3	0,3
Parâmetro α	0,003	0,002

Tabela 6.2: Especificação da rede neural referente a aplicação 2.

Item	Valor	
	Rede Neural 1	Rede Neural 2
Número de vetores padrão	1008	1008
Número de camadas	3	3
Número de neurônios por camada	12-30-1	12-30-1
Tolerância	0,025	0,025
Taxa de treinamento γ^*	5,5	5,5
Termo momento η	0,9	0,9
Inclinação da função sigmóide λ	0,3	0,3
Parâmetro α	0,81	0,47



Tabela 6.3: Especificação da rede neural referente a aplicação 3.

Item	Valor	
	Rede Neural 1	Rede Neural 2
Número de vetores padrão	2016	2016
Número de camadas	3	3
Número de neurônios por camada	12-30-1	12-30-1
Tolerância	0,03	0,03
Taxa de treinamento γ^*	6,0	6,0
Termo momento η	0,9	0,9
Inclinação da função sigmóide λ	0,3	0,3
Parâmetro α	0,06	0,08

Nas Tabelas 6.1, 6.2 e 6.3 são apresentados os principais parâmetros referentes às redes neurais utilizadas e ao treinamento. Os parâmetros referentes ao *algoritmo Back-Propagation* convencional é semelhante aos parâmetros das redes neurais 1 e 2, sendo que o parâmetro α assume valor igual a 0.

Nas Figuras 6.1 e 6.2 são mostrados os resultados da previsão de carga (via B-P convencional e B-P via controlador nebuloso) referentes ao dia 29 de julho de 1998.

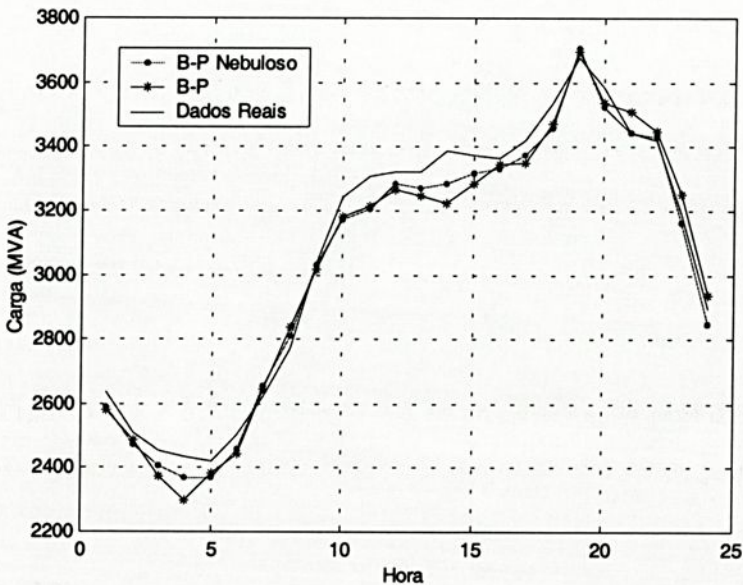


Figura 6.1: Resultado da previsão de carga para rede neural 1, aplicação 1.



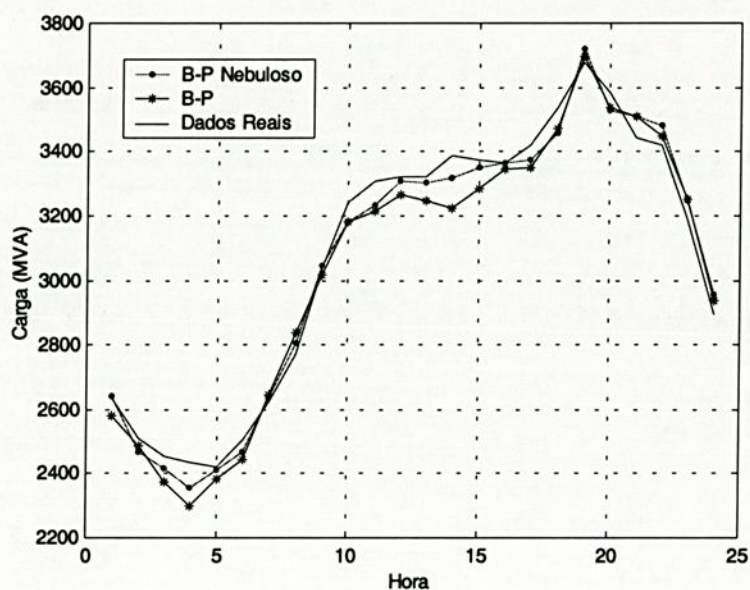


Figura 6.2: Resultado da previsão de carga para rede neural 2, aplicação 1.

Nas Figuras 6.3 e 6.4 são mostrados os resultados da previsão de carga (via B-P convencional e B-P via controlador nebuloso) referentes ao dia 29 de julho de 1998.

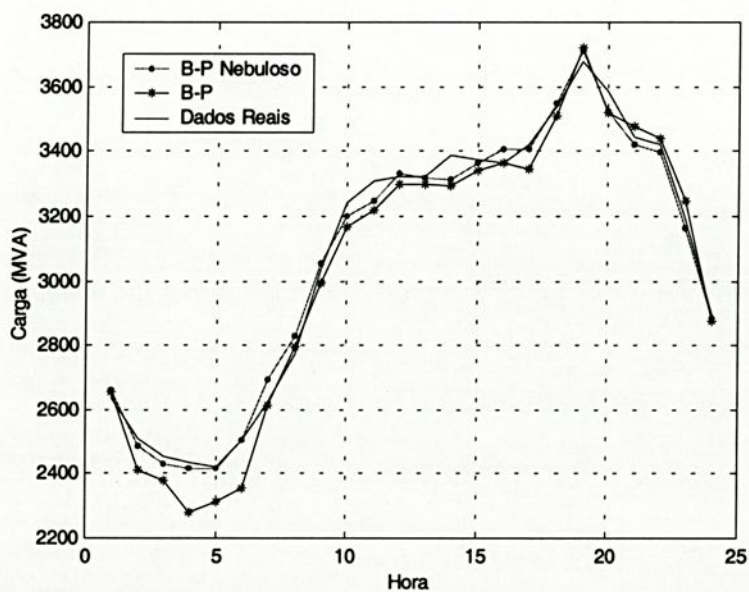


Figura 6.3: Resultado da previsão de carga para rede neural 1, aplicação 2.

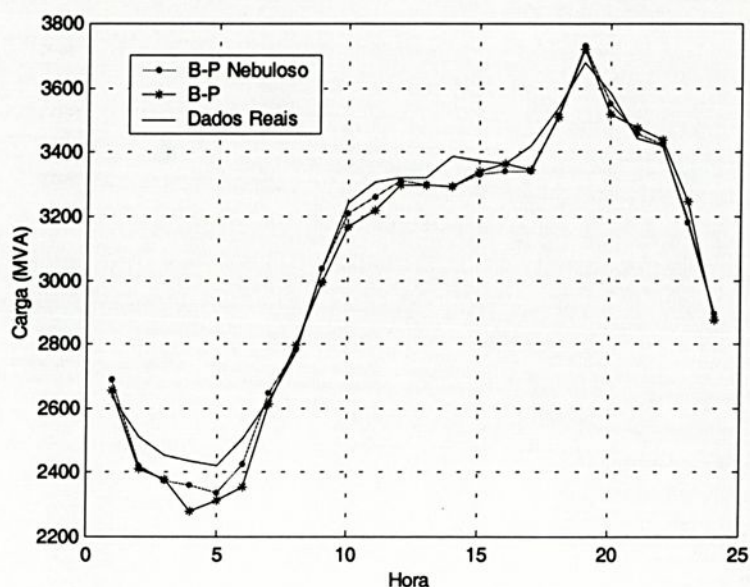


Figura 6.4: Resultado da previsão de carga para rede neural 2, aplicação 2.

Nas Figuras 6.5 e 6.6 são mostrados os resultados da previsão de carga (via B-P convencional e B-P via controlador nebuloso) referentes ao dia 28 de julho de 1998.

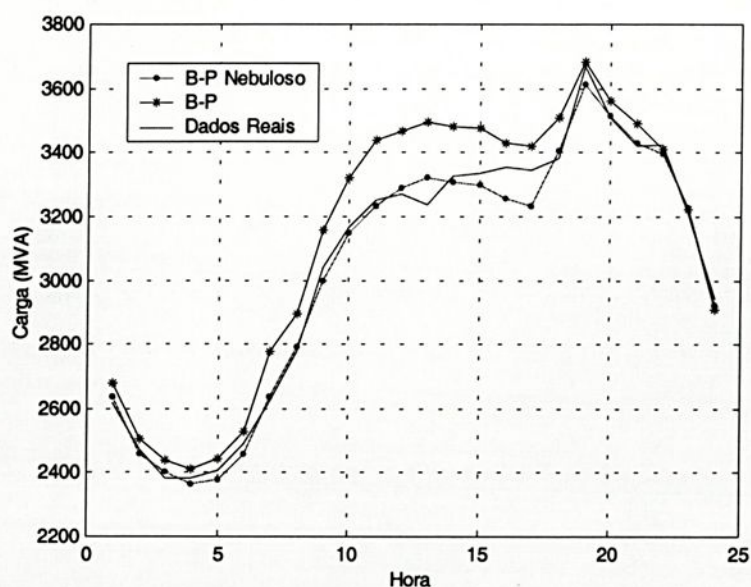


Figura 6.5: Resultado da previsão de carga para rede neural 1, aplicação 3.

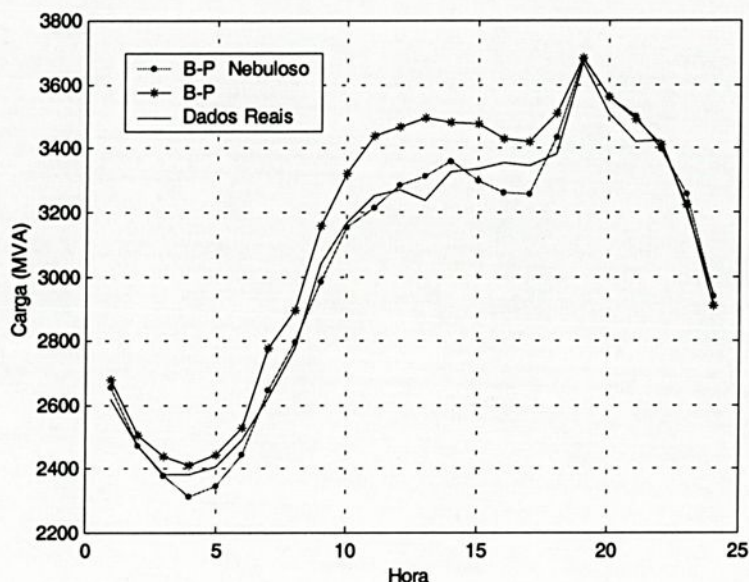


Figura 6.6: Resultado da previsão de carga para rede neural 2, aplicação 3.

Definem-se ganhos, em termos do número de ciclos e do tempo necessário para a realização do treinamento, respectivamente, da seguinte forma ([19]):

$$GC = \frac{NBP}{NBPN} \quad (6.1.5)$$

$$GT = \frac{TBP}{TBPN} \quad (6.1.6)$$

sendo:

NBP = número de ciclos do B-P;

NBPN = número de ciclos do B-P nebuloso;

TBP = tempo de execução (processamento) via B-P;

TBPN = tempo de execução via B-P nebuloso.

Para uma análise da precisão, são definidos o erro percentual absoluto médio *MAPE* ([26]) e o erro máximo da previsão diária, comparando-se os valores reais da carga com os valores estimados através da rede neural da seguinte forma:

$$MAPE = \frac{1}{N} \left\{ \sum_{i=1}^N (|L(h) - \underline{L}(h)|) / L(h) \right\} \times 100 \quad (6.1.7)$$

$$Erro\ máximo\ (\%) = \max \{ |L(h) - \underline{L}(h)| / L(h) \} \times 100 \quad (6.1.8)$$

sendo:

$L(h)$ = valor da carga real referente a hora h ;

$\underline{L}(h)$ = valor da carga estimada pela rede neural referente à hora h ;

N = número total de horas.

Tabela 6.4: Resultados comparativos para aplicação 1.

Item	B-P Convencional	B-P com controlador nebuloso	
		Rede Neural 1	Rede Neural 2
Número ciclos	24801	18394	16973
Tempo processamento (s)	1064,18	814,21	768,13
Ganho GC	–	1,35	1,46
Ganho GT	–	1,31	1,39
Erro médio (%)	2,03	1,56	1,35
Erro máximo (%)	5,62	3,09	3,31

Tabela 6.5: Resultados comparativos para aplicação 2.

Item	B-P Convencional	B-P com controlador nebuloso	
		Rede Neural 1	Rede Neural 2
Número ciclos	2452	1932	1294
Tempo processamento (s)	221,95	179,12	123,37
Ganho GC	–	1,27	1,89
Ganho GT	–	1,24	1,80
Erro médio (%)	1,97	0,91	1,43
Erro máximo (%)	6,36	2,71	3,61

Tabela 6.6: Resultados comparativos para aplicação 3.

Item	B-P Convencional	B-P com controlador nebuloso	
		Rede Neural 1	Rede Neural 2
Número ciclos	1735	1662	1338
Tempo processamento (s)	309,56	298,30	241,73
Ganho GC	–	1,04	1,30
Ganho GT	–	1,03	1,28
Erro médio (%)	3,01	1,03	1,35
Erro máximo (%)	8,01	3,45	2,80

As Tabelas 6.4, 6.5 e 6.6 apresentam os resultados comparativos em termos de ganhos de ciclo e tempo e também fornece, em porcentagem, o erro com relação ao valor real da carga. Deve-se observar que o número de ciclos de treinamento nas três aplicações foi consideravelmente diferente, tendo em vista que foram utilizados tolerâncias distintas. Procurou-se usar tolerâncias mais brandas à medida que aumentava o número de padrões. Isto se fez necessário visando evitar problemas na convergência da rede.

O programa foi elaborado em linguagem FORTRAN e processado em um microcomputador Pentium II (450 MHz / 256 MB de memória RAM). O tempo de processamento refere-se somente à execução do algoritmo B-P (convencional e com o controlador nebuloso), excluindo-se as operações de leitura / saída de dados.

6.2 Outros Testes

Também foram realizados outros testes com o programa B-P *fuzzy* controlado visando observar o seu desempenho em outros tipos de aplicações. Neste sentido, foram abordados três exemplos de aplicação: (1) função NOR exclusiva (duas-entradas) ([35]); (2) Análise de estabilidade transitória de um Sistema de Energia Elétrica composto por 8 máquinas síncronas, 23 linhas de transmissão e 17 barras ([22]) e (3) Análise de estabilidade transitória considerando-se uma versão do sistema da região sul do Brasil (45 barras, 72 linhas de transmissão e 10 máquinas síncronas) ([23]). Na análise de estabilidade transitória é adotado o modelo clássico ([02]) que corresponde à obtenção do tempo crítico para



contingências de curto-circuito com saída de linha de transmissão sob defeito. Considerando-se estes exemplos, são apresentados os resultados comparativos do treinamento realizado via algoritmo B-P convencional e B-P com controlador nebuloso.

Tabela 6.7: Especificação da rede neural.

Item	NOR Exclusivo (1)	Análise de Estabilidade Transitória	
		(2)	(3)
Número de vetores padrões	4	32	56
Número de camadas	3	3	3
Número de neurônios por camadas	2-10-1	5-10-1	5-10-1
Tolerância	0,01	0,02	0,02
Taxa de treinamento (γ^*)	0,3	5,5	5,5
Momento (η)	0,2	0,8	0,8
Inclinação da função sigmóide (λ)	0,5	0,3	0,3
Parâmetro α	0,0	0,0	0,0

Na Tabela 6.7 são apresentados os principais parâmetros referentes à rede neural utilizada no treinamento.

Tabela 6.8: Resultados comparativos.

Item	B-P convencional			B-P com controlador nebuloso		
	(1)	(2)	(3)	(1)	(2)	(3)
Ciclos	53	966	1845	51	589	686
Tempo de processamento (s)	< 1	1,32	3,07	<< 1	1,15	1,60
Ganho GC	-	-	-	1,04	1,64	2,69
Ganho GT	-	-	-	-	1,15	1,92

Na Tabela 6.8 são relacionados os resultados obtidos com o B-P convencional e com o B-P nebuloso.



Nestes exemplos, pode-se constatar que o B-P com controlador nebuloso apresentou, também, um desempenho superior se comparado ao B-P convencional, ou seja, os ganhos maiores que a unidade, sendo, portanto, mais veloz.

6.3 Conclusão

Foi desenvolvido, neste capítulo, uma metodologia para previsão de cargas elétricas via rede neural, utilizando treinamento baseado no algoritmo B-P com controlador nebuloso. Resultados foram apresentados levando-se em conta os dados históricos de uma companhia do setor elétrico brasileiro. A previsão de carga, a curto-prazo, foi realizada considerando-se 24 horas à frente. Através deste exemplo pôde-se verificar que a formulação abordada reduz o número de ciclos de treinamento, bem como o tempo de processamento, se comparado ao algoritmo B-P convencional. O *MAPE* (*Mean Absolute Percentage Error*) e erro máximo da previsão diária, observados neste experimento, foram da ordem de 1,03% e 3,45%, respectivamente, enquanto que foram de 3,01% e 8,01% quando a previsão foi realizada via B-P convencional. Isto evidencia a maior precisão e maior rapidez do método proposto.



Neste capítulo serão apresentadas as conclusões finais relacionadas ao trabalho desenvolvido que corresponde à previsão de cargas elétricas formulada através do uso de redes neurais. Na seqüência, serão relacionadas algumas sugestões para trabalhos futuros.

7.1 Conclusão

Este trabalho teve como objetivo o desenvolvimento de uma metodologia para previsão de cargas elétricas, via rede neural, utilizando treinamento baseado no algoritmo *Back-Propagation* com controlador nebuloso. Foram realizados vários testes, em especial, a previsão de cargas elétricas considerando-se uma base de dados históricos de uma companhia do setor elétrico brasileiro.

Foram realizados, também, outros testes de aplicação da rede neural correspondente à realização da função NOR exclusiva e à análise de estabilidade transitória de dois Sistemas de Energia Elétrica multi-máquinas. Estes testes serviram como parâmetros importantes para se inferir sobre a capacidade da rede neural para a resolução de alguns problemas de Engenharia, servindo, portanto, de um bom indicativo e estímulo para a continuidade da pesquisa alvo que é a previsão de carga.

A previsão de carga, a curto-prazo, foi realizada considerando-se 24 horas à frente. Através deste exemplo pôde-se verificar que a formulação abordada neste trabalho reduz o número de ciclos de treinamento, bem como o tempo de processamento, se comparado ao algoritmo B-P convencional. Pode-se observar que com o aumento do número de vetores de entrada, a rede neural nebulosa tende a ter ganhos maiores. Deve-se ressaltar que a metodologia proposta proporcionou, em todas as simulações realizadas, previsões mais



precisas quando comparadas ao B-P convencional. Isto, certamente, constitui-se num importante recurso para a resolução de problemas de previsão de carga mais complexos em que são considerados um número maior de variáveis e número maior de informações. Importante, também, para aplicação em outros problemas freqüentemente abordados na área elétrica como, por exemplo, análise de estabilidade transitória ([22]), controle de segurança dinâmica ([23]), estabilidade de tensão, etc.

7.2 Sugestões para Trabalhos Futuros

Com base nos resultados obtidos durante a pesquisa de dissertação, sugere-se como prosseguimento do trabalho, os seguintes tópicos:

- estudo e implementação de novas melhorias neste algoritmo, tornando-o mais eficiente em termos de resultados e principalmente, com relação ao tempo de processamento através do:
 - ajuste de forma autônoma da inclinação da função sigmoidal;
 - desenvolvimento de estratégias mais eficientes para o controlador nebuloso;
- desenvolvimento de uma técnica de treinamento de redes, baseada em redes neurais modulares (compostas por várias sub-redes neurais que desempenham funções específicas / mais especializadas) para a previsão de cargas elétricas realizada por barramento que é um problema relevante em abordagens em sistemas de distribuição de energia elétrica, especialmente em ambientes desregulamentados. Estes ambientes competitivos têm estimulado grandes investimentos em pesquisa e aplicação de novas tecnologias em controle e automação.



Referências Bibliográficas

- [01] ALMEIDA, C.; FISHWICH, P. A. and TANG, Z. "Time Series Forecasting Using Neural Network vs. Box-Jenkins Methodology", *Simulation Councies, Inc.*, pp. 303-310, November 1991.

- [02] ANDERSON, P. M. and FOUAD, A. "Power System Control and Stability", *IOWA State University Press, USA*, 1977.

- [03] ARABSHAHI, P.; CHOI, J. J.; MARKS II, R. J. and CAUDELL, T. P. "Fuzzy Parameter Adaptation in Optimization", *IEEE Computational Science & Engineering*, Spring 1996, pp. 57-65.

- [04] AZIZ, S. A. and PARTHIBAN, J. "Fuzzy Logic Introduction", *Fuzzy Logic and Its Uses*, 1996.

- [05] AZIZ, S. A. and PARTHIBAN, J. "Everything You've Always Wanted to Know About Designing Fuzzy Logic Machines But Were Afraid to Ask", *Fuzzy Logic and Its Uses*, 1996.

- [06] AZIZ, S. A. and PARTHIBAN, J. "You Fuzzyin' With Me?", *Fuzzy Logic and Its Uses*, 1996.

- [07] BAUER, P.; NOUAK, S. and WINKLER, R. "A Brief Course in Fuzzy Logic and Fuzzy Control", December 1996.



- [08] BEZDEK, J. C. "Fuzzy Models – What are they, and Why?" *IEEE Transactions on Fuzzy Systems*, 1: , 1993, pp. 1-6.
- [09] CICHOCKI, A. and UNBEHAUEN, R. "Neural Networks For Optimization and Signal Processing", *John Wiley & Sons*, New York, 1993.
- [10] O'DONOVAN, T. M. "Short Term Forecasting: An Introduction to the Box-Jenkins Approach", *John Wiley & Sons*, New York, 1983.
- [11] FINE, T. L. "Feedforward Neural Network Methodology", *Springer-Verlag*, 1999, USA.
- [12] HECHT-NIELSEN, R. "A Basic Introduction to Neural Networks", Computer Sciences Department at the University of Wisconsin – Madison, April 1996.
- [13] JOHN, R. I. "Fuzzy Inferencing System – Problems and Some Solutions", *CMS Working Paper no. 62, School of Computing Sciences of Montfort University*, December 1995.
- [14] KAEHLER, S. D. "Fuzzy Logic Tutorial – An Introduction", *ENCODER – The Newsletter of the Seattle Robotics Society*, March 1998.
- [15] KARTALOPOULOS, S. V. "Understanding Neural Networks and Fuzzy Logic", *IEEE Press*, New York, 1996.
- [16] KROSE, B. and SMAGT, P. V. "An Introduction to Neural Networks", University of Amsterdam, November 1996.
- [17] LAMOTHE, A. "Neural Netware", January 1998.



- [18] LIMA, C. M. and LABIDI, S. "Introdução à Inteligência Artificial", *FAPEMA* – São Luiz, Maranhão, 1999.
- [19] LOPES, M. L. M. and MINUSSI, C. R. "Treinamento de Redes Neurais via Back-Propagation com Controlador Nebuloso", XIII Congresso Brasileiro de Automática, UFSC, Setembro 2000, pp. 1616-1621.
- [20] LOPES, M. L. M.; MINUSSI, C. R. and LOTUFO, A. P. "A Fast Electric Load Forecasting Using Neural Networks", 43rd MIDWEST Symposium on Circuits and Systems, Lansing – Michigan, USA, August 2000.
- [21] MENDES FILHO, E. F. and CARVALHO, A. C. P. L. "Tutorial Introdutório sobre Redes Neurais Artificiais", Departamento de Ciências de Computação e Estatística – USP, Fevereiro 1997.
- ✓ [22] MINUSSI, C. R. and SILVEIRA, M. C. G. "Electric Power System Transient Stability By Neural Networks", 38 *Th Midwest Symposium On Circuits And System*, Rio de Janeiro-RJ, 1995, pp. 1305-1308.
- [23] MINUSSI, C. R. and FREITAS FILHO, W. "Sensitivity Analysis For Transient Stability", *IEE Proceedings – Generation, Transmission and Distribution*, vol. 145, n^o 6, 1998, pp. 669-674.
- [24] PRINS, J. – Statistical Methods Group, SEMATECH "Process on Product Monitoring and Control", June 2000.
- [25] SIMPSON, P. K. "Artificial Neural Systems: Foudations, Paradigms, Applications, and Implementations", New York, *Pergamon Press*, 1989.



- [26] SRINIVASAN, D.; TAN, S. S.; CHANG, C. S. and CHAN, E. K. "Practical Implementation of a Hybrid Fuzzy Neural Network For One-Day-Ahead Load Forecasting," *IEE Proceedings Generation, Transmission and Distribution*, vol. 145, no. 6, pp. 687 – 692, November 1998.
- [27] Statsoft R & D Department, "Electronic Text Book Statsoft – Neural Networks", October 1998.
- [28] STERGIOU, C. "Neural Networks, The Human Brain and Learning", Distributed Software Engineering Group of the Department of Computing, Imperial College of Science Technology and Medicine, University of London, England, 1996.
- [29] STORB, B. H. and WAZLAWICK, R. S. "Algumas 'Pinceladas' sobre Redes Neurais Artificiais", Departamento de Informática e Estatística – UFSC, May 1999.
- [30] TERANO, T.; ASAI, K. and SUGENO, M. "*Fuzzy Systems Theory and Its Applications*", Academic Press, 1991.
- [31] Texas Controls, Inc. "Fuzzy Applied Control Technology – The Theory of Fuzzy Logic", December 1997.
- [32] Ward Systems Group, Inc. "Neural Network Architectures (Neuro Shell 2)", February 2000.
- [33] WASSERMAN, P. D. "Neural Computing - Theory and Practice", *Van Nostrand Reinhold*, 1989.



- [34] WERBOS, P. J. "*Beyond Regression: New Tools For Prediction And Analysis in The Behavioral Sciences*," Master Thesis, Harvard University, 1974.
- ✎ [35] WIDROW, B. and LEHR, M.A. "30 Years of Adaptive Neural Networks: Perceptron, Madaline, and Backpropagation", *Proceedings of the IEEE*, pp. 1415-1442, Vol. 78, nº 9, Sep. 1990.
- [36] WOLFRAM RESEARCH, Inc. "Q & A: What is a fuzzy set?", 1999.



Apêndices A

Hora	Dias																				
	08/7	09/7	10/7	11/7	12/7	13/7	14/7	15/7	16/7	17/7	18/7	19/7	20/7	21/7	22/7	23/7	24/7	25/7	26/7	27/7	28/7
01	2588.5	2615.6	2580.1	2612.3	2442.1	2300.0	2459.6	2469.9	2554.7	2552.2	2666.6	2626.6	2371.0	2578.7	2572.2	2613.1	2603.5	2600.7	2537.5	2461.2	2615.5
02	2450.4	2477.3	2478.4	2467.2	2325.6	2194.3	2303.6	2377.2	2416.5	2419.8	2505.7	2504.2	2357.8	2421.4	2437.3	2468.9	2465.3	2466.2	2400.4	2355.1	2471.1
03	2396.6	2428.2	2423.2	2386.5	2277.4	2157.0	2273.3	2311.9	2348.4	2353.3	2434.4	2388.4	2256.4	2411.1	2388.4	2417.1	2420.4	2348.4	2329.4	2284.3	2382.8
04	2343.9	2402.0	2376.6	2325.7	2227.3	2114.7	2232.2	2279.2	2303.9	2347.0	2373.2	2337.7	2246.7	2368.5	2368.2	2380.1	2367.5	2361.1	2280.4	2237.1	2381.3
05	2364.3	2423.1	2384.1	2329.0	2195.4	2164.1	2257.4	2284.4	2336.2	2350.2	2391.2	2345.0	2254.1	2398.2	2403.1	2410.0	2398.0	2356.3	2240.8	2263.8	2406.4
06	2442.7	2508.9	2464.8	2344.4	2211.2	2227.5	2388.6	2406.0	2412.9	2446.1	2400.1	2323.3	2353.1	2468.3	2498.5	2502.2	2502.9	2379.3	2260.1	2359.8	2491.7
07	2663.8	2727.4	2746.5	2381.4	2189.7	2462.5	2550.0	2559.2	2592.6	2646.9	2395.7	2279.2	2530.4	2625.1	2639.7	2648.3	2674.3	2363.1	2177.9	2508.7	2623.7
08	2792.0	2892.1	2844.8	2413.2	2183.4	2682.2	2667.2	2702.2	2764.5	2777.2	2427.0	2255.9	2763.2	2812.2	2814.0	2831.5	3016.2	2442.3	2185.1	2705.8	2775.3
09	3027.1	3152.7	3009.1	2598.8	2265.7	2892.8	2854.6	2928.1	2950.7	3022.9	2693.0	2313.7	2990.9	3029.8	3086.8	3098.6	3145.1	2558.9	2286.8	2980.2	3041.2
10	3176.8	3250.1	3154.1	2708.8	2367.6	3047.1	2969.6	3038.5	3077.6	3143.3	2730.2	2336.1	3157.4	3211.7	3248.7	3146.3	3259.3	2677.9	2328.5	3151.7	3168.7
11	3250.9	3332.2	3216.9	2733.6	2400.8	3125.4	3034.7	3071.7	3159.6	3192.1	2758.2	2358.1	3232.1	3262.5	3264.9	3211.9	3303.6	2747.8	2367.9	3242.9	3249.9
12	3366.2	3347.4	3221.7	2716.7	2382.6	3072.7	3063.4	3170.4	3169.0	3224.8	2741.8	2415.9	3236.2	3280.3	3248.9	3289.8	3290.4	2748.7	2381.5	3269.0	3269.0
13	3351.0	3334.5	3192.6	2721.5	2394.8	3093.6	3031.3	3121.7	3143.6	3198.3	2759.3	2438.9	3165.7	3388.8	3232.8	3290.5	3259.9	2747.8	2421.9	3271.9	3237.5
14	3349.0	3359.5	3179.5	2710.2	2373.6	3110.2	3056.9	3123.0	3154.7	3246.8	2781.9	2392.0	3213.1	3412.8	3257.3	3324.9	3326.3	2726.7	2387.7	3340.5	3325.6
15	3429.4	3284.7	3210.8	2689.6	2336.6	3087.8	3085.0	3145.8	3211.6	3254.5	2745.7	2360.9	3262.1	3435.4	3262.6	3314.1	3336.1	2673.6	2367.9	3340.0	3336.7
16	3447.8	3295.4	3239.7	2719.7	2304.6	3113.4	3052.9	3178.8	3206.5	3237.8	2673.4	2339.1	3279.7	3464.0	3289.2	3311.7	3273.6	2675.9	2312.2	3292.7	3353.4
17	3400.9	3403.6	3161.7	2728.7	2111.4	3195.1	3099.0	3204.7	3270.3	3260.9	2699.0	2429.6	3365.7	3374.4	3334.2	3336.2	3301.3	2637.6	2396.5	3330.2	3347.0
18	3531.4	3569.2	3300.4	3014.4	2383.6	3345.3	3313.8	3349.5	3390.2	3380.8	3017.3	2871.9	3466.5	3489.9	3379.2	3447.2	3434.4	2979.3	2717.2	3390.5	3382.0
19	3803.1	3715.9	3594.8	3376.2	2932.4	3553.4	3567.3	3654.3	3682.3	3667.5	3451.0	3247.6	3643.4	3757.2	3626.6	3586.4	3567.5	3409.3	3218.9	3655.2	3667.4
20	3589.6	3512.6	3461.3	3316.3	3006.1	3401.8	3424.2	3457.8	3502.2	3498.2	3423.9	3182.6	3510.0	3613.3	3446.9	3448.9	3421.4	3310.8	3176.4	3490.9	3508.6
21	3503.7	3438.2	3349.8	3254.3	2964.0	3305.5	3322.1	3389.9	3415.7	3431.6	3339.0	3161.7	3472.1	3514.7	3420.4	3418.0	3338.2	3250.2	3134.5	3411.8	3421.2
22	3524.5	3449.0	3341.2	3157.4	2862.2	3304.7	3325.2	3392.1	3414.0	3374.8	3227.0	3110.8	3446.0	3427.0	3385.6	3352.8	3322.9	3216.9	3053.9	3395.3	3425.7
23	3223.4	3175.5	3149.8	2967.2	2705.5	3027.5	3045.1	3153.1	3149.3	3205.5	3095.2	2960.6	3182.9	3200.5	3181.7	3188.6	3113.6	2977.2	2939.6	3206.3	3210.5
24	2934.4	2822.2	2893.2	2713.0	2496.2	2728.5	2736.0	2830.2	2885.9	2961.9	2879.4	2754.4	2910.5	2856.1	2898.1	2910.8	2860.1	2766.9	2704.0	2882.2	2941.4



Apêndices B

Hora	Dias																				
	17/6	18/6	19/6	20/6	21/6	22/6	23/6	24/6	25/6	26/6	27/6	28/6	29/6	30/6	01/7	02/7	03/7	04/7	05/7	06/7	07/7
01	2541.6	2605.6	2620.4	2630.9	2438.8	2264.9	2440.4	2513.4	2588.8	2499.3	2581.7	2478.0	2435.4	2574.6	2515.5	2575.1	2665.7	2613.8	2582.3	2480.3	2619.2
02	2399.8	2450.1	2462.3	2474.6	2317.7	2163.3	2305.7	2366.2	2387.1	2367.6	2405.1	2357.3	2302.6	2396.7	2345.4	2460.9	2511.6	2487.7	2418.9	2354.9	2488.9
03	2366.2	2388.2	2435.1	2383.2	2259.9	2110.3	2267.7	2312.1	2348.3	2315.8	2340.9	2274.0	2284.2	2287.3	2350.6	2364.7	2440.8	2406.7	2342.3	2293.1	2402.9
04	2329.3	2396.3	2371.6	2342.9	2187.3	2094.4	2231.9	2299.9	2370.6	2300.6	2324.8	2236.9	2254.8	2241.1	2318.9	2350.6	2423.3	2368.4	2302.8	2260.5	2335.2
05	2364.7	2388.5	2394.3	2359.2	2187.7	2137.5	2271.2	2301.7	2396.9	2347.5	2290.9	2189.7	2282.5	2286.7	2334.9	2360.7	2425.9	2335.7	2262.5	2282.8	2414.7
06	2492.1	2478.5	2508.8	2386.0	2183.2	2237.1	2356.4	2408.1	2488.5	2427.4	2317.4	2169.7	2390.2	2384.3	2429.0	2440.6	2501.6	2381.4	2241.5	2410.4	2490.1
07	2729.8	2731.3	2718.5	2439.4	2166.3	2489.3	2595.7	2605.6	2722.4	2668.1	2321.5	2158.8	2626.8	2580.4	2616.4	2684.7	2639.8	2446.1	2231.7	2628.4	2714.7
08	2831.8	2835.3	2832.5	2488.8	2133.3	2689.7	2728.0	2789.7	2889.2	2772.6	2393.1	2144.0	2797.7	2720.7	2778.1	2781.7	2805.6	2455.9	2181.8	2786.3	2841.6
09	3027.0	3024.5	3082.8	2656.7	2205.1	2896.1	2948.4	3034.8	3063.8	2956.4	2581.4	2283.0	2949.1	2912.5	2969.1	3000.7	3016.2	2598.0	2293.2	2976.2	3083.9
10	3141.9	3168.4	3251.1	2788.7	2339.3	3057.1	3099.4	3122.5	3225.2	3105.4	2697.9	2313.3	3124.0	3068.9	3090.0	3146.7	3163.4	2731.0	2337.7	3153.4	3171.5
11	3194.8	3238.7	3381.0	2837.9	2410.8	3154.5	3150.5	3206.2	3280.0	3199.6	2738.4	2377.6	3194.4	3147.3	3201.8	3212.6	3259.2	2763.4	2348.7	3279.4	3215.2
12	3258.8	3247.0	3381.6	2831.1	2439.7	3201.8	3153.1	3253.4	3304.3	3151.1	2734.6	2412.3	3258.4	3190.5	3261.9	3330.7	3239.3	2803.0	2402.7	3294.9	3206.1
13	3256.6	3178.5	3363.6	2790.8	2451.0	3149.5	3148.4	3255.7	3291.9	3173.2	2717.7	2441.8	3264.7	3161.8	3240.2	3333.4	3205.6	2822.3	2392.4	3289.0	3183.6
14	3236.5	3174.7	3374.1	2809.1	2455.9	3140.3	3092.7	3285.5	3283.4	3192.1	2660.9	2432.4	3264.4	3156.0	3241.3	3345.3	3163.2	2767.3	2388.4	3314.5	3133.8
15	3265.1	3207.4	3379.2	2796.3	2433.5	3132.3	2922.4	3296.2	3243.9	3205.3	2616.2	2409.1	3256.2	3205.8	3251.9	3330.7	3010.5	2745.7	2351.6	3331.5	2968.2
16	3263.4	3211.7	3376.1	2751.1	2420.0	3129.3	2791.6	3309.6	3225.5	3190.0	2536.7	2351.9	3249.0	3202.9	3237.9	3348.6	2830.4	2737.2	2339.7	3339.7	2805.8
17	3331.3	3287.9	3364.4	2792.7	2471.1	3176.7	2479.9	3332.5	3224.5	3163.1	2307.6	2401.3	3271.8	3228.6	3301.4	3387.1	2538.7	2738.0	2412.4	3333.2	2549.6
18	3606.7	3538.4	3602.9	3181.5	2914.5	3348.6	2698.6	3598.5	3389.4	3432.1	2658.7	2858.7	3476.3	3432.2	3450.8	3552.1	2718.5	3127.4	2884.9	3476.5	2671.6
19	3718.4	3695.9	3730.3	3328.9	3136.1	3513.0	3272.8	3741.8	3556.8	3558.3	3289.2	3232.5	3673.1	3610.7	3706.5	3712.3	3298.9	3487.4	3261.2	3699.0	3062.1
20	3517.3	3519.3	3536.5	3240.2	3038.5	3383.2	3223.3	3540.6	3463.2	3400.8	3255.3	3158.5	3528.9	3484.1	3518.5	3534.1	3262.8	3376.2	3174.8	3489.7	3192.9
21	3434.1	3472.8	3488.4	3165.1	2971.4	3345.0	3194.2	3446.0	3320.6	3364.0	3152.4	3100.9	3384.9	3450.6	3444.3	3503.0	3229.2	3286.3	3152.7	3415.0	3248.9
22	3414.3	3393.9	3360.0	3028.3	2889.9	3230.9	3246.8	3434.9	3252.7	3305.8	3075.3	3006.2	3335.7	3339.9	3413.9	3425.1	3261.7	3181.8	3045.2	3519.4	3296.7
23	3222.0	3193.6	3242.5	2873.0	2726.3	3024.5	3052.5	3207.6	3024.0	3097.5	2938.9	2903.1	3147.3	3160.4	3191.3	3212.3	3160.4	2964.1	2908.2	3180.2	3160.0
24	2880.7	2917.0	2941.9	2694.1	2516.7	2671.9	2793.5	2878.8	2756.6	2852.0	2723.6	2672.9	2841.6	2839.5	2899.1	2947.6	2884.8	2765.7	2699.8	2898.7	2854.0



Hora	Dias																				
	08/7	09/7	10/7	11/7	12/7	13/7	14/7	15/7	16/7	17/7	18/7	19/7	20/7	21/7	22/7	23/7	24/7	25/7	26/7	27/7	28/7
01	2588.5	2615.6	2580.1	2612.3	2442.1	2300.0	2459.6	2469.9	2554.7	2552.2	2666.6	2626.6	2371.0	2578.7	2572.2	2613.1	2603.5	2600.7	2537.5	2461.2	2615.5
02	2450.4	2477.3	2478.4	2467.2	2325.6	2194.3	2303.6	2377.2	2416.5	2419.8	2505.7	2504.2	2357.8	2421.4	2437.3	2468.9	2465.3	2466.2	2400.4	2355.1	2471.1
03	2396.6	2428.2	2423.2	2386.5	2277.4	2157.0	2273.3	2311.9	2348.4	2353.3	2434.4	2388.4	2256.4	2411.1	2388.4	2417.1	2420.4	2348.4	2329.4	2284.3	2382.8
04	2343.9	2402.0	2376.6	2325.7	2227.3	2114.7	2232.2	2279.2	2303.9	2347.0	2373.2	2337.7	2246.7	2368.5	2368.2	2380.1	2367.5	2361.1	2280.4	2237.1	2381.3
05	2364.3	2423.1	2384.1	2329.0	2195.4	2164.1	2257.4	2284.4	2336.2	2350.2	2391.2	2345.0	2254.1	2398.2	2403.1	2410.0	2398.0	2356.3	2240.8	2263.8	2406.4
06	2442.7	2508.9	2464.8	2344.4	2211.2	2227.5	2388.6	2406.0	2412.9	2446.1	2400.1	2323.3	2353.1	2468.3	2498.5	2502.2	2502.9	2379.3	2260.1	2359.8	2491.7
07	2663.8	2727.4	2746.5	2381.4	2189.7	2462.5	2550.0	2559.2	2592.6	2646.9	2395.7	2279.2	2530.4	2625.1	2639.7	2648.3	2674.3	2363.1	2177.9	2508.7	2623.7
08	2792.0	2892.1	2844.8	2413.2	2183.4	2682.2	2667.2	2702.2	2764.5	2777.2	2427.0	2255.9	2763.2	2812.2	2814.0	2831.5	3016.2	2442.3	2185.1	2705.8	2775.3
09	3027.1	3152.7	3009.1	2598.8	2265.7	2892.8	2854.6	2928.1	2950.7	3022.9	2693.0	2313.7	2990.9	3029.8	3086.8	3098.6	3145.1	2558.9	2286.8	2980.2	3041.2
10	3176.8	3250.1	3154.1	2708.8	2367.6	3047.1	2969.6	3038.5	3077.6	3143.3	2730.2	2336.1	3157.4	3211.7	3248.7	3146.3	3259.3	2677.9	2328.5	3151.7	3168.7
11	3250.9	3332.2	3216.9	2733.6	2400.8	3125.4	3034.7	3071.7	3159.6	3192.1	2758.2	2358.1	3232.1	3262.5	3264.9	3211.9	3303.6	2747.8	2367.9	3242.9	3249.9
12	3366.2	3347.4	3221.7	2716.7	2382.6	3072.7	3063.4	3170.4	3169.0	3224.8	2741.8	2415.9	3236.2	3280.3	3248.9	3289.8	3290.4	2748.7	2381.5	3269.0	3269.0
13	3351.0	3334.5	3192.6	2721.5	2394.8	3093.6	3031.3	3121.7	3143.6	3198.3	2759.3	2438.9	3165.7	3388.8	3232.8	3290.5	3259.9	2747.8	2421.9	3271.9	3237.5
14	3349.0	3359.5	3179.5	2710.2	2373.6	3110.2	3056.9	3123.0	3154.7	3246.8	2781.9	2392.0	3213.1	3412.8	3257.3	3324.9	3326.3	2726.7	2387.7	3340.5	3325.6
15	3429.4	3284.7	3210.8	2689.6	2336.6	3087.8	3085.0	3145.8	3211.6	3254.5	2745.7	2360.9	3262.1	3435.4	3262.6	3314.1	3336.1	2673.6	2367.9	3340.0	3336.7
16	3447.8	3295.4	3239.7	2719.7	2304.6	3113.4	3052.9	3178.8	3206.5	3237.8	2673.4	2339.1	3279.7	3464.0	3289.2	3311.7	3273.6	2675.9	2312.2	3292.7	3353.4
17	3400.9	3403.6	3161.7	2728.7	2111.4	3195.1	3099.0	3204.7	3270.3	3260.9	2699.0	2429.6	3365.7	3374.4	3334.2	3336.2	3301.3	2637.6	2396.5	3330.2	3347.0
18	3531.4	3569.2	3300.4	3014.4	2383.6	3345.3	3313.8	3349.5	3390.2	3380.8	3017.3	2871.9	3466.5	3489.9	3379.2	3447.2	3434.4	2979.3	2717.2	3390.5	3382.0
19	3803.1	3715.9	3594.8	3376.2	2932.4	3553.4	3567.3	3654.3	3682.3	3667.5	3451.0	3247.6	3643.4	3757.2	3626.6	3586.4	3567.5	3409.3	3218.9	3655.2	3667.4
20	3589.6	3512.6	3461.3	3316.3	3006.1	3401.8	3424.2	3457.8	3502.2	3498.2	3423.9	3182.6	3510.0	3613.3	3446.9	3448.9	3421.4	3310.8	3176.4	3490.9	3508.6
21	3503.7	3438.2	3349.8	3254.3	2964.0	3305.5	3322.1	3389.9	3415.7	3431.6	3339.0	3161.7	3472.1	3514.7	3420.4	3418.0	3338.2	3250.2	3134.5	3411.8	3421.2
22	3524.5	3449.0	3341.2	3157.4	2862.2	3304.7	3325.2	3392.1	3414.0	3374.8	3227.0	3110.8	3446.0	3427.0	3385.6	3352.8	3322.9	3216.9	3053.9	3395.3	3425.7
23	3223.4	3175.5	3149.8	2967.2	2705.5	3027.5	3045.1	3153.1	3149.3	3205.5	3095.2	2960.6	3182.9	3200.5	3181.7	3188.6	3113.6	2977.2	2939.6	3206.3	3210.5
24	2934.4	2822.2	2893.2	2713.0	2496.2	2728.5	2736.0	2830.2	2885.9	2961.9	2879.4	2754.4	2910.5	2856.1	2898.1	2910.8	2860.1	2766.9	2704.0	2882.2	2941.4



Apêndices C

Hora	Dias																				
	04/5	05/5	06/5	07/5	08/5	09/5	10/5	11/5	12/5	13/5	14/5	15/5	16/5	17/5	18/5	19/5	20/5	21/5	22/5	23/5	24/5
01	2630.9	2683.4	2562.3	2597.4	2588.5	2588.0	2547.1	2374.9	2604.9	2724.3	2728.1	2833.2	3160.0	2880.5	2498.9	2658.7	2603.6	2662.4	2764.2	2912.0	2816.5
02	2577.2	2527.8	2460.4	2424.8	2449.5	2427.6	2396.1	2282.6	2485.9	2601.2	2572.9	2708.7	2995.3	2746.3	2398.0	2514.5	2512.8	2536.4	2623.2	2760.0	2672.5
03	2483.4	2436.1	2403.7	2392.8	2368.9	2350.0	2305.7	2226.1	2433.1	2525.5	2563.9	2616.0	2891.6	2641.8	2325.3	2449.9	2446.2	2505.9	2574.8	2628.4	2599.9
04	2503.5	2420.3	2354.8	2374.2	2368.9	2332.1	2236.9	2210.8	2415.4	2495.1	2537.5	2627.1	2872.9	2559.5	2316.0	2422.1	2408.0	2457.5	2527.8	2615.3	2510.8
05	2432.7	2377.0	2373.4	2389.7	2373.7	2328.3	2217.6	2225.2	2430.5	2518.9	2500.7	2626.1	2802.2	2532.6	2355.9	2432.0	2452.0	2461.1	2564.4	2582.6	2487.4
06	2559.8	2518.4	2484.9	2513.9	2459.0	2361.4	2227.1	2349.1	2547.0	2592.5	2625.2	2701.9	2787.9	2506.1	2442.7	2540.7	2541.3	2566.8	2651.5	2601.0	2496.1
07	2771.7	2719.0	2664.8	2684.4	2617.5	2316.3	2161.1	2531.1	2694.8	2735.0	2774.6	2889.5	2764.8	2498.2	2649.9	2773.2	2752.9	2747.9	2829.6	2603.5	2444.1
08	2959.6	2913.3	2830.1	2761.2	2761.8	2391.5	2178.7	2706.6	2827.2	2925.6	2964.5	3027.8	2782.9	2489.3	2832.4	2896.6	2872.0	2941.1	3000.0	2651.9	2381.9
09	3185.8	3154.8	3080.1	3047.0	2990.0	2566.5	2269.7	2948.2	3035.9	3143.8	3177.2	3327.2	2897.3	2536.4	3090.2	3118.3	3140.5	3155.4	3234.8	2763.0	2474.0
10	3469.2	3291.6	3199.2	3192.0	3163.5	2665.3	2311.9	3111.9	3224.7	3299.3	3340.0	3464.4	2986.0	2603.4	3258.7	3237.5	3283.8	3329.2	3373.0	2880.5	2522.6
11	3555.7	3386.8	3293.4	3259.9	3278.7	2724.1	2359.5	3218.5	3312.9	3405.6	3532.7	3588.7	3041.8	2644.9	3359.8	3292.0	3361.2	3417.2	3490.3	2957.5	2493.2
12	3552.6	3416.0	3366.8	3250.4	3280.8	2701.5	2425.3	3272.2	3346.5	3433.9	3461.6	3616.7	3008.2	2638.8	3351.6	3314.8	3409.9	3450.3	3505.9	2949.8	2540.1
13	3450.4	3335.5	3326.8	3226.1	3245.8	2740.3	2405.8	3254.5	3343.3	3433.8	3470.1	3630.4	3012.8	2640.9	3325.9	3279.7	3353.7	3445.8	3499.4	2963.1	2542.6
14	3434.4	3338.3	3309.5	3245.7	3249.3	2690.7	2376.9	3273.9	3392.4	3510.6	3497.7	3733.4	3011.9	2558.8	3357.6	3292.1	3317.4	3451.1	3497.9	2958.8	2565.5
15	3447.7	3325.5	3303.2	3207.9	3241.0	2698.1	2287.4	3295.3	3451.6	3534.9	3521.1	3764.0	2993.3	2495.4	3342.3	3265.1	3405.9	3447.2	3463.2	2897.8	2553.0
16	3477.7	3286.4	3259.4	3234.3	3198.8	2671.3	2313.3	3284.3	3482.1	3542.2	3546.9	3732.6	2987.2	2496.5	3363.1	3244.8	3417.2	3420.8	3487.5	2866.1	2557.5
17	3559.3	3322.4	3328.1	3266.0	3201.3	2684.8	2341.2	3300.9	3483.0	3500.4	3538.0	3715.4	3088.8	2565.8	3448.9	3352.6	3424.4	3495.5	3470.3	2915.5	2625.9
18	3694.7	3541.7	3523.8	3440.7	3348.2	3065.3	2712.3	3499.3	3631.7	3659.5	3779.1	3863.7	3488.8	2993.5	3680.9	3625.2	3631.2	3701.3	3683.3	3326.2	3063.5
19	3622.3	3632.0	3604.7	3587.6	3571.7	3425.9	3154.4	3675.9	3787.0	3815.0	3822.6	4015.9	3593.3	3224.6	3650.8	3645.3	3762.4	3781.7	3777.9	3675.4	3394.1
20	3497.6	3429.3	3460.6	3438.3	3405.8	3319.3	3060.3	3535.0	3697.7	3669.7	3726.1	3780.7	3507.5	3137.6	3489.3	3487.0	3583.2	3594.0	3621.5	3593.9	3378.5
21	3435.9	3342.7	3375.4	3374.4	3332.3	3264.7	3068.4	3431.9	3541.2	3527.2	3605.8	3710.9	3429.7	3120.5	3397.7	3418.0	3497.1	3549.0	3526.5	3483.4	3316.0
22	3322.8	3294.7	3306.2	3325.4	3288.1	3126.0	3003.9	3362.6	3493.6	3467.1	3509.8	3727.6	3325.1	3039.8	3335.5	3372.5	3429.2	3443.2	3458.0	3356.2	3224.0
23	3151.1	3155.3	3138.5	3107.7	3103.6	2976.6	2884.1	3184.8	3312.6	3327.1	3429.7	3571.9	3237.1	2957.6	3164.9	3224.5	3312.8	3349.5	3378.4	3251.4	3140.4
24	2879.2	2858.7	2862.0	2853.2	2821.3	2769.2	2664.2	2897.9	3007.9	3038.5	3134.5	3365.3	3054.9	2740.8	2906.3	2907.8	2961.0	3023.9	3127.1	3025.5	2879.2



Hora	Dias																				
	25/5	26/5	27/5	28/5	29/5	30/5	31/5	01/6	02/6	03/6	04/6	05/6	06/6	07/6	08/6	09/6	10/6	12/6	13/6	14/6	15/6
01	2666.3	2837.3	2882.5	2725.2	2758.3	2635.2	2568.2	2401.6	2573.8	2613.3	2571.6	2568.8	2622.9	2521.5	2374.7	2512.6	2523.0	2597.1	2587.5	2506.5	2411.8
02	2538.6	2741.6	2777.1	2607.7	2593.7	2508.8	2404.1	2303.4	2459.5	2494.6	2468.1	2468.6	2467.7	2384.3	2288.1	2426.4	2411.0	2460.4	2463.6	2406.8	2331.4
03	2500.1	2669.3	2766.8	2552.9	2564.7	2436.0	2340.5	2281.9	2405.6	2424.8	2385.4	2399.0	2403.1	2305.7	2213.3	2372.4	2363.8	2383.8	2413.5	2299.7	2248.4
04	2468.2	2635.9	2722.7	2534.8	2507.7	2375.9	2285.6	2264.4	2383.7	2398.5	2395.7	2398.5	2366.4	2252.3	2218.5	2350.6	2362.4	2354.7	2370.0	2254.6	2233.4
05	2498.9	2611.7	2690.1	2546.9	2520.7	2384.5	2286.8	2278.6	2396.3	2427.0	2434.9	2398.8	2358.7	2234.7	2261.8	2389.1	2365.0	2312.4	2346.7	2242.1	2268.2
06	2599.0	2716.3	2786.9	2651.5	2641.7	2380.5	2267.2	2403.9	2501.1	2547.6	2511.4	2495.9	2388.5	2243.9	2354.6	2485.7	2450.6	2429.8	2364.9	2276.3	2348.3
07	2772.8	2877.3	2984.4	2873.8	2813.2	2474.3	2204.6	2606.3	2681.3	2769.2	2792.0	2703.4	2370.4	2159.3	2548.8	2675.4	2610.7	2591.1	2388.6	2237.2	2591.4
08	2921.0	2965.9	3101.7	2993.5	2956.4	2509.2	2174.9	2766.6	2762.3	2917.3	2941.6	2852.6	2421.0	2074.4	2694.5	2792.9	2698.9	2753.6	2439.6	2191.1	2757.8
09	3173.2	3212.9	3385.0	3288.6	3218.9	2662.0	2316.6	2957.1	2983.0	3130.2	3115.6	3044.2	2559.0	2189.8	2923.3	2953.7	2937.1	3007.8	2613.0	2315.1	2973.6
10	3414.3	3419.8	3503.2	3442.9	3364.6	2796.8	2357.0	3128.9	3096.1	3274.5	3242.3	3169.4	2677.6	2319.8	3058.5	3074.1	3015.7	3154.7	2702.2	2328.5	3077.1
11	3486.5	3502.6	3569.9	3489.0	3455.6	2844.5	2422.6	3256.6	3178.4	3324.3	3318.2	3254.4	2731.7	2370.9	3145.2	3171.3	3100.3	3217.4	2776.9	2412.1	3201.6
12	3540.3	3534.7	3563.9	3505.1	3437.5	2837.6	2448.2	3276.1	3202.4	3375.0	3339.2	3263.1	2742.4	2391.3	3213.2	3176.1	3051.0	3234.1	2739.7	2427.7	3210.6
13	3527.8	3510.9	3514.3	3467.1	3442.7	2814.5	2427.0	3250.4	3220.8	3354.1	3326.5	3255.9	2737.5	2410.8	3193.2	3150.2	2788.7	3146.0	2757.0	2446.9	3211.9
14	3550.9	3558.3	3513.5	3436.4	3507.7	2807.1	2394.8	3276.9	3226.2	3310.7	3320.1	3255.3	2747.7	2414.0	3217.5	3174.8	2635.2	3202.7	2703.8	2419.5	3218.8
15	3558.8	3638.9	3568.0	3437.2	3531.7	2742.7	2393.2	3262.9	3237.3	3321.2	3328.6	3270.2	2679.8	2368.0	3229.3	3168.3	2707.1	3209.6	2697.9	2386.7	3219.4
16	3523.6	3639.5	3628.7	3409.7	3466.0	2694.5	2418.8	3260.3	3182.4	3312.2	3333.6	3244.5	2663.4	2367.3	3201.3	3182.9	2922.1	3161.1	2667.2	2308.1	3214.1
17	3586.2	3652.9	3606.5	3426.1	3470.2	2791.0	2483.8	3264.2	3227.9	3398.9	3379.7	3259.9	2700.7	2463.4	3244.4	3240.1	3004.2	3201.1	2770.6	2389.9	3239.0
18	3808.3	3441.1	3833.5	3765.8	3689.3	3243.9	2958.5	3552.2	3499.1	3637.5	3589.4	3519.5	3130.5	2890.3	3580.8	3504.4	3290.3	3402.8	3211.1	2875.2	3576.8
19	3891.9	3628.8	3809.4	3805.0	3665.6	3519.0	3271.7	3640.2	3631.7	3692.5	3653.2	3618.7	3444.6	3249.5	3616.8	3669.4	3491.4	3519.2	3439.0	3275.7	3669.0
20	3711.0	3528.2	3619.8	3574.8	3474.4	3408.2	3223.9	3494.1	3447.2	3525.1	3487.7	3462.6	3349.3	3131.5	3470.8	3493.7	3368.6	3375.2	3380.7	3163.5	3508.6
21	3610.7	3514.9	3567.5	3540.4	3437.7	3333.6	3125.8	3420.8	3361.5	3401.5	3403.2	3373.0	3327.6	3051.6	3428.2	3432.6	3248.3	3297.3	3241.9	3096.9	3433.5
22	3600.7	3530.2	3443.1	3479.4	3327.1	3165.4	3029.1	3350.6	3361.5	3370.6	3322.1	3331.1	3161.0	2998.6	3310.6	3361.0	3206.2	3243.8	3106.1	3019.0	3343.1
23	3404.0	3446.1	3237.1	3320.8	3138.3	2987.5	2863.5	3101.5	3193.9	3167.9	3144.1	3168.2	2959.2	2864.5	3089.1	3117.9	3051.6	3056.0	2937.4	2894.6	3128.8
24	3111.3	3189.7	3005.0	3047.6	2899.8	2779.5	2659.9	2839.5	2859.9	2839.0	2863.0	2865.3	2756.6	2580.7	2792.4	2816.5	2815.9	2807.7	2718.5	2639.1	2832.3



Hora	Dias																				
	16/6	17/6	18/6	19/6	20/6	21/6	22/6	23/6	24/6	25/6	26/6	27/6	28/6	29/6	30/6	01/7	02/7	03/7	04/7	05/7	06/7
01	2583.2	2541.6	2605.6	2620.4	2630.9	2438.8	2264.9	2440.4	2513.4	2588.8	2499.3	2581.7	2478.0	2435.4	2574.6	2515.5	2575.1	2665.7	2613.8	2582.3	2480.3
02	2426.5	2399.8	2450.1	2462.3	2474.6	2317.7	2163.3	2305.7	2366.2	2387.1	2367.6	2405.1	2357.3	2302.6	2396.7	2345.4	2460.9	2511.6	2487.7	2418.9	2354.9
03	2370.9	2366.2	2388.2	2435.1	2383.2	2259.9	2110.3	2267.7	2312.1	2348.3	2315.8	2340.9	2274.0	2284.2	2287.3	2350.6	2364.7	2440.8	2406.7	2342.3	2293.1
04	2339.6	2329.3	2396.3	2371.6	2342.9	2187.3	2094.4	2231.9	2299.9	2370.6	2300.6	2324.8	2236.9	2254.8	2241.1	2318.9	2350.6	2423.3	2368.4	2302.8	2260.5
05	2351.0	2364.7	2388.5	2394.3	2359.2	2187.7	2137.5	2271.2	2301.7	2396.9	2347.5	2290.9	2189.7	2282.5	2286.7	2334.9	2360.7	2425.9	2335.7	2262.5	2282.8
06	2463.0	2492.1	2478.5	2508.8	2386.0	2183.2	2237.1	2356.4	2408.1	2488.5	2427.4	2317.4	2169.7	2390.2	2384.3	2429.0	2440.6	2501.6	2381.4	2241.5	2410.4
07	2654.0	2729.8	2731.3	2718.5	2439.4	2166.3	2489.3	2595.7	2605.6	2722.4	2668.1	2321.5	2158.8	2626.8	2580.4	2616.4	2684.7	2639.8	2446.1	2231.7	2628.4
08	2763.9	2831.8	2835.3	2832.5	2488.8	2133.3	2689.7	2728.0	2789.7	2889.2	2772.6	2393.1	2144.0	2797.7	2720.7	2778.1	2781.7	2805.6	2455.9	2181.8	2786.3
09	2947.2	3027.0	3024.5	3082.8	2656.7	2205.1	2896.1	2948.4	3034.8	3063.8	2956.4	2581.4	2283.0	2949.1	2912.5	2969.1	3000.7	3016.2	2598.0	2293.2	2976.2
10	3071.9	3141.9	3168.4	3251.1	2788.7	2339.3	3057.1	3099.4	3122.5	3225.2	3105.4	2697.9	2313.3	3124.0	3068.9	3090.0	3146.7	3163.4	2731.0	2337.7	3153.4
11	3152.9	3194.8	3238.7	3381.0	2837.9	2410.8	3154.5	3150.5	3206.2	3280.0	3199.6	2738.4	2377.6	3194.4	3147.3	3201.8	3212.6	3259.2	2763.4	2348.7	3279.4
12	3142.5	3258.8	3247.0	3381.6	2831.1	2439.7	3201.8	3153.1	3253.4	3304.3	3151.1	2734.6	2412.3	3258.4	3190.5	3261.9	3330.7	3239.3	2803.0	2402.7	3294.9
13	3106.5	3256.6	3178.5	3363.6	2790.8	2451.0	3149.5	3148.4	3255.7	3291.9	3173.2	2717.7	2441.8	3264.7	3161.8	3240.2	3333.4	3205.6	2822.3	2392.4	3289.0
14	3095.7	3236.5	3174.7	3374.1	2809.1	2455.9	3140.3	3092.7	3285.5	3283.4	3192.1	2660.9	2432.4	3264.4	3156.0	3241.3	3345.3	3163.2	2767.3	2388.4	3314.5
15	2970.2	3265.1	3207.4	3379.2	2796.3	2433.5	3132.3	2922.4	3296.2	3243.9	3205.3	2616.2	2409.1	3256.2	3205.8	3251.9	3330.7	3010.5	2745.7	2351.6	3331.5
16	2789.0	3263.4	3211.7	3376.1	2751.1	2420.0	3129.3	2791.6	3309.6	3225.5	3190.0	2536.7	2351.9	3249.0	3202.9	3237.9	3348.6	2830.4	2737.2	2339.7	3339.7
17	2490.1	3331.3	3287.9	3364.4	2792.7	2471.1	3176.7	2479.9	3332.5	3224.5	3163.1	2307.6	2401.3	3271.8	3228.6	3301.4	3387.1	2538.7	2738.0	2412.4	3333.2
18	2783.1	3606.7	3538.4	3602.9	3181.5	2914.5	3348.6	2698.6	3598.5	3389.4	3432.1	2658.7	2858.7	3476.3	3432.2	3450.8	3552.1	2718.5	3127.4	2884.9	3476.5
19	3320.5	3718.4	3695.9	3730.3	3328.9	3136.1	3513.0	3272.8	3741.8	3556.8	3558.3	3289.2	3232.5	3673.1	3610.7	3706.5	3712.3	3298.9	3487.4	3261.2	3699.0
20	3279.4	3517.3	3519.3	3536.5	3240.2	3038.5	3383.2	3223.3	3540.6	3463.2	3400.8	3255.3	3158.5	3528.9	3484.1	3518.5	3534.1	3262.8	3376.2	3174.8	3489.7
21	3261.0	3434.1	3472.8	3488.4	3165.1	2971.4	3345.0	3194.2	3446.0	3320.6	3364.0	3152.4	3100.9	3384.9	3450.6	3444.3	3503.0	3229.2	3286.3	3152.7	3415.0
22	3244.4	3414.3	3393.9	3360.0	3028.3	2889.9	3230.9	3246.8	3434.9	3252.7	3305.8	3075.3	3006.2	3335.7	3339.9	3413.9	3425.1	3261.7	3181.8	3045.2	3519.4
23	3079.1	3222.0	3193.6	3242.5	2873.0	2726.3	3024.5	3052.5	3207.6	3024.0	3097.5	2938.9	2903.1	3147.3	3160.4	3191.3	3212.3	3160.4	2964.1	2908.2	3180.2
24	2781.1	2880.7	2917.0	2941.9	2694.1	2516.7	2671.9	2793.5	2878.8	2756.6	2852.0	2723.6	2672.9	2841.6	2839.5	2899.1	2947.6	2884.8	2765.7	2699.8	2898.7



Hora	Dias																				
	07/7	08/7	09/7	10/7	11/7	12/7	13/7	14/7	15/7	16/7	17/7	18/7	19/7	20/7	21/7	22/7	23/7	24/7	25/7	26/7	27/7
01	2619.2	2588.5	2615.6	2580.1	2612.3	2442.1	2300.0	2459.6	2469.9	2554.7	2552.2	2666.6	2626.6	2371.0	2578.7	2572.2	2613.1	2603.5	2600.7	2537.5	2461.2
02	2488.9	2450.4	2477.3	2478.4	2467.2	2325.6	2194.3	2303.6	2377.2	2416.5	2419.8	2505.7	2504.2	2357.8	2421.4	2437.3	2468.9	2465.3	2466.2	2400.4	2355.1
03	2402.9	2396.6	2428.2	2423.2	2386.5	2277.4	2157.0	2273.3	2311.9	2348.4	2353.3	2434.4	2388.4	2256.4	2411.1	2388.4	2417.1	2420.4	2348.4	2329.4	2284.3
04	2335.2	2343.9	2402.0	2376.6	2325.7	2227.3	2114.7	2232.2	2279.2	2303.9	2347.0	2373.2	2337.7	2246.7	2368.5	2368.2	2380.1	2367.5	2361.1	2280.4	2237.1
05	2414.7	2364.3	2423.1	2384.1	2329.0	2195.4	2164.1	2257.4	2284.4	2336.2	2350.2	2391.2	2345.0	2254.1	2398.2	2403.1	2410.0	2398.0	2356.3	2240.8	2263.8
06	2490.1	2442.7	2508.9	2464.8	2344.4	2211.2	2227.5	2388.6	2406.0	2412.9	2446.1	2400.1	2323.3	2353.1	2468.3	2498.5	2502.2	2502.9	2379.3	2260.1	2359.8
07	2714.7	2663.8	2727.4	2746.5	2381.4	2189.7	2462.5	2550.0	2559.2	2592.6	2646.9	2395.7	2279.2	2530.4	2625.1	2639.7	2648.3	2674.3	2363.1	2177.9	2508.7
08	2841.6	2792.0	2892.1	2844.8	2413.2	2183.4	2682.2	2667.2	2702.2	2764.5	2777.2	2427.0	2255.9	2763.2	2812.2	2814.0	2831.5	3016.2	2442.3	2185.1	2705.8
09	3083.9	3027.1	3152.7	3009.1	2598.8	2265.7	2892.8	2854.6	2928.1	2950.7	3022.9	2693.0	2313.7	2990.9	3029.8	3086.8	3098.6	3145.1	2558.9	2286.8	2980.2
10	3171.5	3176.8	3250.1	3154.1	2708.8	2367.6	3047.1	2969.6	3038.5	3077.6	3143.3	2730.2	2336.1	3157.4	3211.7	3248.7	3146.3	3259.3	2677.9	2328.5	3151.7
11	3215.2	3250.9	3332.2	3216.9	2733.6	2400.8	3125.4	3034.7	3071.7	3159.6	3192.1	2758.2	2358.1	3232.1	3262.5	3264.9	3211.9	3303.6	2747.8	2367.9	3242.9
12	3206.1	3366.2	3347.4	3221.7	2716.7	2382.6	3072.7	3063.4	3170.4	3169.0	3224.8	2741.8	2415.9	3236.2	3280.3	3248.9	3289.8	3290.4	2748.7	2381.5	3269.0
13	3183.6	3351.0	3334.5	3192.6	2721.5	2394.8	3093.6	3031.3	3121.7	3143.6	3198.3	2759.3	2438.9	3165.7	3388.8	3232.8	3290.5	3259.9	2747.8	2421.9	3271.9
14	3133.8	3349.0	3359.5	3179.5	2710.2	2373.6	3110.2	3056.9	3123.0	3154.7	3246.8	2781.9	2392.0	3213.1	3412.8	3257.3	3324.9	3326.3	2726.7	2387.7	3340.5
15	2968.2	3429.4	3284.7	3210.8	2689.6	2336.6	3087.8	3085.0	3145.8	3211.6	3254.5	2745.7	2360.9	3262.1	3435.4	3262.6	3314.1	3336.1	2673.6	2367.9	3340.0
16	2805.8	3447.8	3295.4	3239.7	2719.7	2304.6	3113.4	3052.9	3178.8	3206.5	3237.8	2673.4	2339.1	3279.7	3464.0	3289.2	3311.7	3273.6	2675.9	2312.2	3292.7
17	2549.6	3400.9	3403.6	3161.7	2728.7	2111.4	3195.1	3099.0	3204.7	3270.3	3260.9	2699.0	2429.6	3365.7	3374.4	3334.2	3336.2	3301.3	2637.6	2396.5	3330.2
18	2671.6	3531.4	3569.2	3300.4	3014.4	2383.6	3345.3	3313.8	3349.5	3390.2	3380.8	3017.3	2871.9	3466.5	3489.9	3379.2	3447.2	3434.4	2979.3	2717.2	3390.5
19	3062.1	3803.1	3715.9	3594.8	3376.2	2932.4	3553.4	3567.3	3654.3	3682.3	3667.5	3451.0	3247.6	3643.4	3757.2	3626.6	3586.4	3567.5	3409.3	3218.9	3655.2
20	3192.9	3589.6	3512.6	3461.3	3316.3	3006.1	3401.8	3424.2	3457.8	3502.2	3498.2	3423.9	3182.6	3510.0	3613.3	3446.9	3448.9	3421.4	3310.8	3176.4	3490.9
21	3248.9	3503.7	3438.2	3349.8	3254.3	2964.0	3305.5	3322.1	3389.9	3415.7	3431.6	3339.0	3161.7	3472.1	3514.7	3420.4	3418.0	3338.2	3250.2	3134.5	3411.8
22	3296.7	3524.5	3449.0	3341.2	3157.4	2862.2	3304.7	3325.2	3392.1	3414.0	3374.8	3227.0	3110.8	3446.0	3427.0	3385.6	3352.8	3322.9	3216.9	3053.9	3395.3
23	3160.0	3223.4	3175.5	3149.8	2967.2	2705.5	3027.5	3045.1	3153.1	3149.3	3205.5	3095.2	2960.6	3182.9	3200.5	3181.7	3188.6	3113.6	2977.2	2939.6	3206.3
24	2854.0	2934.4	2822.2	2893.2	2713.0	2496.2	2728.5	2736.0	2830.2	2885.9	2961.9	2879.4	2754.4	2910.5	2856.1	2898.1	2910.8	2860.1	2766.9	2704.0	2882.2

**DEPARTAMENTO DE ENGENHARIA ELÉTRICA
AV. BRASIL, 56 - CAIXA POSTAL 31
15378-000 ILHA SOLTEIRA - SP**

