



Universidade Federal de Pernambuco
Centro de Informática

Pós-graduação em Ciência da Computação

Proposta de uma Classe de Perceptrons Híbridos com Aprendizagem baseada em Gradiente Descendente

Ricardo de Andrade Araújo

Dissertação de Mestrado

Recife
Fevereiro de 2012

Universidade Federal de Pernambuco
Centro de Informática

Ricardo de Andrade Araújo

Proposta de uma Classe de Perceptrons Híbridos com Aprendizagem baseada em Gradiente Descendente

*Trabalho apresentado ao Programa de Pós-graduação em
Ciência da Computação do Centro de Informática da Univer-
sidade Federal de Pernambuco como requisito parcial para
obtenção do grau de Mestre em Ciência da Computação.*

Orientador: *Silvio Romero de Lemos Meira*
Co-orientador: *Adriano Lorena Inácio de Oliveira*

Recife
Fevereiro de 2012

Catálogo na fonte
Bibliotecária Jane Souto Maior, CRB4-571

Araújo, Ricardo de Andrade

**Proposta de uma classe de perceptrons híbridos com
aprendizagem baseada em gradiente descendente /
Ricardo de Andrade Araújo - Recife: O Autor, 2012.
xxv, 142 folhas: il., fig., tab., gráf.**

**Orientador: Silvio Romero de Lemos Meira.
Dissertação (mestrado) - Universidade Federal de
Pernambuco. CIn, Ciência da Computação, 2012.**

Inclui bibliografia.

**1. Inteligência computacional. 2. Modelos híbridos. 3.
Morfologia matemática. 4. Teoria de reticulados. I. Meira, Silvio
Romero de Lemos (orientador). II. Título.**

006.3

CDD (23. ed.)

MEI2012 – 023

*A Jesus, Deus, ao Divino Espírito Santo
À minha esposa, Alany
À minha Mãe, Rubenilde*

Agradecimentos

Primeiramente gostaria de mencionar meus agradecimentos a Jesus, Deus, e ao Divino Espírito Santo, que deram-me saúde, força, luz e sabedoria para caminhar nesta jornada repleta de momentos difíceis e caminhos longínquos. Também, gostaria de agradecer a Mãe Rainha, a Nossa Senhora do Trabalho, a Nossa Senhora Desatadora dos Nós, a Nossa Senhora da Conceição, a Nossa Senhora Aparecida, Nossa Senhora das Graças e a Nossa Senhora de Fátima que iluminaram constantemente meu caminho desde o início deste trabalho.

Agradeço à minha querida e amada Alany, minha esposa, companheira e fonte eterna de inspiração, pelos momentos de amor, amizade, cumplicidade, incentivo, apoio, força e coragem nas situações decisivas e críticas para a conclusão deste trabalho, sendo o meu eterno porto seguro, sendo a calma do meu mar.

Agradeço à minha Mãe Rubenilde, pelo seu amor incondicional, pela sua força e coragem que me garantiram uma criação exemplar, por todas as dificuldades e privações que enfrentou para que eu pudesse estudar e ser alguém na vida, por todo seu empenho e dedicação em ser meu norte quando eu navegava por águas turvas e turbulentas, por ter desempenhado com sucesso a função de Pai e Mãe simultaneamente, e pelo seu incansável incentivo para a conclusão deste trabalho. Meu sentimento de agradecimento por minha Mãe converge para três palavras: “Eu te amo!”.

Agradeço ao meu querido, amado e saudoso avô Guilherme e minha avó Benilde, que sempre sonharam com esta conquista. Um agradecimento especial para toda minha família materna. Também, um agradecimento especial para Diana e para minha querida Vó, que constantemente estão orando e pedindo graças para mim.

Gostaria de agradecer aos meus orientadores, amigos e conselheiros, Prof. Adriano Oliveira e Prof. Silvio Meira, por acreditarem neste trabalho, e principalmente pelo empenho, apoio, sugestões e contribuições que viabilizaram o desenvolvimento e enriquecimento desta dissertação. Também um agradecimento em particular ao amigo Prof. Sergio Soares pelo apoio e força antes e durante o desenvolvimento deste trabalho.

Um agradecimento especial ao Prof. Aluizio Araújo, responsável por minha formação teórica em redes neurais artificiais e sistemas dinâmicos, fundamentais para o desenvolvimento desta dissertação. Também, agradeço a todos os professores da pós-graduação do CIn/UFPE que contribuíram de forma direta ou indireta em minha formação acadêmica.

Agradeço a todos os professores da graduação do DEI/UNICAP que contribuíram expressivamente em meus primeiros passos da vida acadêmica. Um agradecimento particular ao Prof. Francisco Madeiro, ao Prof. Tiago Ferreira, ao Prof. Robson Pequeno, ao Prof. Sergio Murilo Fernandes, ao Prof. Robson Lins, ao Prof. Silvio Bandeira e ao Prof. Bertino Figueiredo. Também um agradecimento particular a minha antiga “equipe” de iniciação científica: Mariana, Wellington, Glaucio, Euclides, Saulo, Sarmanho, Paulo, Gustavo, Iandé, Aranildo, Renato, Carol, Harry, Zé Henrique, Jefferson e Marcelo.

Gostaria de agradecer ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pelo apoio financeiro (Processo 132721/2011-2), permitindo o desenvolvimento e conclusão desta dissertação.

Por fim, um “muito obrigado” a todos aqueles que de alguma forma contribuíram com o desenvolvimento deste trabalho.

A melhor maneira que o homem dispõe para se aperfeiçoar é aproximar-se de Deus
—PITÁGORAS (570 a.C–496 a.C.)

Resumo

Este trabalho apresenta uma classe de *perceptrons* híbridos baseado nos princípios da morfologia matemática (*mathematical morphology*, MM) no contexto de teoria de reticulados (*lattice theory*). O modelo proposto, chamado de *perceptron* de dilatação-erosão-linear (*dilation-erosion-linear perceptron*, DELP), consiste de uma combinação linear entre operadores não-lineares (do tipo morfológicos no contexto de teoria de reticulados) e um operador linear (do tipo resposta finita ao impulso), sendo desenvolvido na tentativa de superar o dilema do passeio aleatório (*random walk dilemma*, RWD) no problema de previsão de séries temporais financeiras. Para projetar o DELP (processo de aprendizagem), foi apresentado um método de gradiente descendente utilizando ideias do algoritmo de retropropagação do erro (*back propagation*, BP) e uma abordagem sistemática para superar o problema da não-diferenciabilidade das operações morfológicas de dilatação e erosão. Também, no processo de aprendizagem do DELP, foi incluída uma etapa adicional para ajustar distorções de fase temporais que ocorrem na reconstrução do espaço de fase de fenômenos temporais provenientes do mercado financeiro.

Uma análise experimental foi conduzida utilizando um conjunto de séries temporais financeiras: Índice da Bolsa de Valores de São Paulo, Índice *Dow Jones Industrial Average*, Índice *National Association of Securities Dealers Automated Quotation*, Índice *Financial Times and London Stock Exchange 100*, Preço das ações do Bradesco PN, Preço das ações da Gol PN, Preço das ações do Itaú Unibanco PN, Preço das ações da Petrobras PN, Preço das ações da Usiminas PNA e Preço das ações da Vale PNA. Nestes experimentos, foram utilizadas cinco métricas e uma função de avaliação para mensurar o desempenho preditivo do modelo proposto, e os resultados alcançados superaram aqueles obtidos utilizando técnicas consolidadas na literatura.

Palavras-chave: Perceptrons Híbridos, Morfologia Matemática, Teoria de Reticulados, Método de Gradiente Descendente, Previsão de Séries Temporais Financeiras, Dilema do Passeio Aleatório, Ajuste Automático de Fase Temporal.

Abstract

This work presents a class of hybrid perceptrons based on principles of mathematical morphology (MM) under context of lattice theory. The proposed model, called the dilation-erosion-linear perceptron (DELP), consists of a linear combination of nonlinear operators (morphological like under context of lattice theory) and a linear operator (finite impulse response like), being developed to overcome the random walk dilemma (RWD) in the financial time series prediction problem. In order to design the proposed DELP (learning process), we present a gradient steepest descent method using the back propagation (BP) algorithm ideas and a systematic approach to overcome the problem of nondifferentiability of morphological operations. We have included an additional step in the DELP learning process to adjust time phase distortions that occur in the phase space reconstruction of temporal phenomena from financial market.

An experimental analysis was conducted using a relevant set of financial time series: Bolsa de Valores de São Paulo Index, Dow Jones Industrial Average Index, National Association of Securities Dealers Automated Quotation Index, Financial Times and London Stock Exchange 100 Index, Bradesco PN stock prices, Gol PN stock prices, Itaú Unibanco PN stock prices, Petrobras PN stock prices, Usiminas PNA stock prices and Vale PNA stock prices. In these experiments, we used five relevant metrics and an evaluation function to assess the prediction performance of the proposed model, and the achieved results overcame those obtained by established techniques in the literature.

Keywords: Hybrid Perceptrons, Mathematical Morphology, Lattice Theory, Gradient Steepest Descent Method, Financial Time Series Prediction, Random Walk Dilemma, Time Phase Adjustment.

Sumário

Lista de Figuras	xix
Lista de Tabelas	xxi
Lista de Símbolos	xxiii
1 Introdução	1
1.1 Introdução	1
1.2 Motivação e Justificativa	2
1.3 Objetivos	4
1.4 Estrutura da Dissertação	6
2 O Problema de Previsão de Séries Temporais	9
2.1 Introdução	9
2.2 Definição do Problema de Previsão	10
2.3 Definição do Dilema do Passeio Aleatório	13
2.4 Resumo do Capítulo	15
3 Modelos para Previsão de Séries Temporais	17
3.1 Introdução	17
3.2 Modelos Estatísticos Lineares	18
3.2.1 Modelos de Box e Jenkins	19
3.3 Modelos Estatísticos Não-Lineares	20
3.3.1 Modelos Heterocedásticos Condicionais Auto-Regressivos	20
3.3.2 Modelos Bilineares	21
3.3.3 Modelos de Limiar Auto-Regressivo	21
3.3.4 Modelos de Estados Gerais Dependentes	22
3.3.5 Modelos Auto-Regressivos Exponenciais	22
3.3.6 Modelos de Limiar Auto-Regressivo Auto-Excitado	22
3.3.7 Considerações	23
3.4 Redes Neurais Artificiais	23
3.4.1 Perceptron Multicamadas	24
3.5 Algoritmos Evolucionários	27
3.5.1 Algoritmos Genéticos	28
3.5.1.1 Algoritmo Genético Modificado	30
3.5.2 Otimizador por Enxame de Partículas	32

3.6	Sistemas Híbridos	34
3.6.1	Algoritmos Genéticos e Redes Neurais Multicamadas	35
3.6.2	Otimizador por Exame de Partículas e Redes Neurais Multicamadas	39
3.7	Sistemas Híbridos Otimizados	41
3.7.1	O método <i>time-delay added evolutionary forecasting</i> (TAEF)	41
3.7.2	O método <i>swarm-based hybrid intelligent forecasting</i> (SHIF)	44
3.8	Considerações Finais	48
3.9	Resumo do Capítulo	48
4	Caracterização das Séries Temporais Utilizadas	49
4.1	Séries Temporais Financeiras	49
4.1.1	Índice da Bolsa de Valores de São Paulo (BVSP)	49
4.1.2	Índice <i>Dow Jones Industrial Average</i> (DJI)	49
4.1.3	Índice <i>National Association of Securities Dealers Automated Quotation</i> (IXIC)	52
4.1.4	Índice <i>Financial Times and London Stock Exchange 100</i> (FTSE)	53
4.1.5	Preço das Ações do Bradesco PN (BBDC4)	54
4.1.6	Preço das Ações da GOL PN (GOLL4)	55
4.1.7	Preço das Ações do Itaú Unibanco PN (ITUB4)	57
4.1.8	Preço das Ações da Petrobras PN (PETR4)	58
4.1.9	Preço das Ações da Usiminas PNA (USIM5)	59
4.1.10	Preço das Ações da Vale PNA (VALE5)	61
4.1.11	Considerações	62
4.2	Métricas para Desempenho de Previsão	63
4.2.1	Erro Médio Quadrático	63
4.2.2	Erro Médio Percentual Absoluto	64
4.2.3	Estatística <i>u</i> de Theil	64
4.2.4	Previsão de Mudança na Direção	64
4.2.5	Variância Relativa Média	64
4.2.6	Função de Avaliação	65
4.3	Resumo do Capítulo	65
5	O Modelo Proposto: <i>Dilation-Erosion-Linear Perceptron</i> (DELP)	67
5.1	Introdução	67
5.2	Fundamentos e Teorias Relevantes	68
5.3	Definição do DELP	77
5.3.1	Fundamentos para o Processo de Aprendizagem	77
5.3.2	Processo de Aprendizagem com Ajuste Automático de Fase Temporal	80
5.4	Resumo do Capítulo	84
6	Simulações e Resultados Experimentais	85
6.1	Metodologia	85
6.2	Resultados	89
6.2.1	Índice da Bolsa de Valores de São Paulo (BVSP)	89

6.2.2	Índice <i>Dow Jones Industrial Average</i> (DJI)	91
6.2.3	Índice <i>National Association of Securities Dealers Automated Quotation</i> (IXIC)	94
6.2.4	Índice <i>Financial Times and London Stock Exchange 100</i> (FTSE)	97
6.2.5	Preço das Ações do Bradesco PN (BBDC4)	101
6.2.6	Preço das Ações da GOL PN (GOLL4)	104
6.2.7	Preço das Ações do Itaú Unibanco PN (ITUB4)	107
6.2.8	Preço das Ações da Petrobras PN (PETR4)	110
6.2.9	Preço das Ações da Usiminas PNA (USIM5)	115
6.2.10	Preço das Ações da Vale PNA (VALE5)	117
6.2.11	Considerações	120
6.3	Resumo do Capítulo	122
7	Conclusões	123
7.1	Resumo da Dissertação	123
7.2	Principais Contribuições	127
7.3	Principais Limitações da Dissertação	128
7.4	Trabalhos Futuros	129

Lista de Figuras

3.1	Pseudo-código do algoritmo genético padrão.	29
3.2	Pseudo-código do algoritmo genético modificado.	31
3.3	Pseudo-código do otimizador de enxame de partículas.	34
3.4	Esquema do sistema híbrido MGA–MLP.	37
3.5	Esquema do sistema híbrido PSO–MLP.	40
3.6	Esquema do sistema híbrido otimizado TAEF.	42
3.7	Esquema do sistema híbrido otimizado SHIF.	45
4.1	Gráfico da série BVSP.	50
4.2	<i>Lagplot</i> da série BVSP.	51
4.3	Gráfico da série DJI.	51
4.4	<i>Lagplot</i> da série DJI.	52
4.5	Gráfico da série IXIC.	53
4.6	<i>Lagplot</i> da série IXIC.	53
4.7	Gráfico da série FTSE.	54
4.8	<i>Lagplot</i> da série FTSE.	55
4.9	Gráfico da série BBDC4.	55
4.10	<i>Lagplot</i> da série BBDC4.	56
4.11	Gráfico da série GOLL4.	57
4.12	<i>Lagplot</i> da série GOLL4.	57
4.13	Gráfico da série ITUB4.	58
4.14	<i>Lagplot</i> da série ITUB4.	59
4.15	Gráfico da série PETR4.	59
4.16	<i>Lagplot</i> da série PETR4.	60
4.17	Gráfico da série USIM5.	61
4.18	<i>Lagplot</i> da série USIM5.	61
4.19	Gráfico da série VALE5.	62
4.20	<i>Lagplot</i> da série VALE5.	63
5.1	Arquitetura do DELP.	78
5.2	Procedimento de correção de fase.	83
5.3	Passos do algoritmo de treinamento do modelo proposto.	84
6.1	Previsão da série BVSP gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).	91

6.2	Previsão da série BVSP gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).	92
6.3	Previsão da série DJI gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).	95
6.4	Previsão da série DJI gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).	95
6.5	Previsão da série IXIC gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).	98
6.6	Previsão da série IXIC gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).	98
6.7	Previsão da série FTSE gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).	101
6.8	Previsão da série FTSE gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).	102
6.9	Previsão da série BBDC4 gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).	104
6.10	Previsão da série BBDC4 gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).	105
6.11	Previsão da série GOLL4 gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).	108
6.12	Previsão da série GOLL4 gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).	108
6.13	Previsão da série ITUB4 gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).	111
6.14	Previsão da série ITUB4 gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).	111
6.15	Previsão da série PETR4 gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).	114
6.16	Previsão da série PETR4 gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).	114
6.17	Previsão da série USIM5 gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).	117
6.18	Previsão da série USIM5 gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).	118
6.19	Previsão da série VALE5 gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).	120
6.20	Previsão da série VALE5 gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).	121

Lista de Tabelas

4.1	Estatísticas da série BVSP.	50
4.2	Estatísticas da série DJI.	50
4.3	Estatísticas da série IXIC.	52
4.4	Estatísticas da série FTSE.	54
4.5	Estatísticas da série BBDC4.	56
4.6	Estatísticas da série GOLL4.	56
4.7	Estatísticas da série ITUB4.	58
4.8	Estatísticas da série PETR4.	60
4.9	Estatísticas da série USIM5.	60
4.10	Estatísticas da série VALE5.	62
6.1	Resultados de todos os experimentos utilizando o modelo DELP para a série BVSP (conjunto de treinamento)	90
6.2	Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série BVSP.	90
6.3	Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série BVSP.	91
6.4	Resultados de todos os experimentos utilizando o modelo proposto para a série DJI (conjunto de treinamento)	93
6.5	Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série DJI.	93
6.6	Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série DJI.	94
6.7	Resultados de todos os experimentos utilizando o modelo proposto para a série IXIC (conjunto de treinamento)	96
6.8	Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série IXIC.	97
6.9	Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série IXIC.	97
6.10	Resultados de todos os experimentos utilizando o modelo proposto para a série FTSE (conjunto de treinamento)	99
6.11	Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série FTSE.	100
6.12	Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série FTSE.	100

6.13	Resultados de todos os experimentos utilizando o modelo proposto para a série BBDC4 (conjunto de treinamento)	103
6.14	Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série BBDC4.	103
6.15	Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série BBDC4.	104
6.16	Resultados de todos os experimentos utilizando o modelo proposto para a série GOLL4 (conjunto de treinamento)	106
6.17	Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série GOLL4.	106
6.18	Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série GOLL4.	107
6.19	Resultados de todos os experimentos utilizando o modelo proposto para a série ITUB4 (conjunto de treinamento)	109
6.20	Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série ITUB4.	110
6.21	Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série ITUB4.	110
6.22	Resultados de todos os experimentos utilizando o modelo proposto para a série PETR4 (conjunto de treinamento)	112
6.23	Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série PETR4.	113
6.24	Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série PETR4.	113
6.25	Resultados de todos os experimentos utilizando o modelo proposto para a série USIM5 (conjunto de treinamento)	115
6.26	Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série USIM5.	116
6.27	Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série USIM5.	116
6.28	Resultados de todos os experimentos utilizando o modelo proposto para a série VALE5 (conjunto de treinamento)	119
6.29	Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série VALE5.	119
6.30	Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série VALE5.	120

Lista de Símbolos

\leq	relação binária em um conjunto não vazio
(A, \leq)	conjunto parcialmente ordenado
\bigvee	operador de supremo
\bigwedge	operador de ínfimo
$+$	operação de soma entre grupos
$+'$	operação adicional de soma entre grupos
α	saída do módulo não-linear do DELP
\mathbf{a}	vetor de coeficientes do operador de dilatação do DELP
a^*	conjugado de um elemento
ACF	<i>autocorrelation function</i> – função de auto-correlação
ANN	<i>artificial neural network</i> – rede neural artificial
AR	<i>autoregressive</i> – modelo auto-regressivo
ARCH	<i>autoregressive conditional heteroskedasticity</i> – modelo heterocedástico condicional auto-regressivo
ARMA	<i>autoregressive moving average</i> – modelo auto-regressivo de médias móveis
ARIMA	<i>autoregressive integrated moving average</i> – modelos autoregressivos integrados de médias móveis
ARV	<i>average relative variance</i> – variância relativa média
\mathbf{b}	vetor de coeficientes do operador de erosão do DELP
β	saída do módulo linear do DELP
BBDC4	preço de fechamento das ações do bradesco PN
BLOG	<i>bounded lattice ordered group</i> – grupo ordenado de reticulado limitado
BP	<i>back propagation</i> – algoritmo de retropropagação do erro
BVSP	índice da bolsa de valores de são paulo
$\mathbf{c}(\mathbf{x}, r)$	r -ézimo vetor indicador de posto do vetor \mathbf{x}
$\mathbf{c}_\sigma(\mathbf{x}, r)$	r -ézimo vetor indicador de posto suave do vetor \mathbf{x}
CI	<i>computational intelligence</i> – inteligência computacional
CLODUM	<i>complete lattice ordered double monoid</i> – monóide duplo ordenado de reticulado completo
d	dimensionalidade do vetor de entrada (retardos temporais) do DELP
δ	operador de dilatação
DELP	<i>dilation-erosion-linear perceptron</i> – <i>perceptron</i> de dilatação-erosão-linear
DJI	índice <i>dow jones industrial average</i>
$e(m)$	erro instantâneo para o m -ézimo padrão de treinamento

EA	<i>evolutionary algorithm</i> – algoritmo evolucionário
EF	<i>evaluation function</i> – função de avaliação
ε	operador de erosão
φ	saída do operador de dilatação do DELP
FTSE	índice <i>financial times and london stock exchange 100</i>
$g(t)$	componente não-linear sub-dominante de uma série temporal no tempo t
GA	<i>genetic algorithm</i> – algoritmo genético
<i>Ganho</i>	ganho percentual
GARCH	<i>generalized autoregressive conditional heteroskedasticity</i> – modelo heterocedástico condicional auto-regressivo generalizado
GI	critério de parada <i>generalization loss</i>
GOLL4	preço de fechamento das ações da gol PN
HS	<i>hybrid system</i> – sistema híbrido
ITUB4	preço de fechamento das ações do itaú unibanco PN
IXIC	índice <i>national association of securities dealers automated quotation</i>
$J(\mathbf{w})$	função objetivo do processo de aprendizagem do DELP
lag	parâmetro para definição dos retardos temporais utilizados para caracterização da série temporal
<i>lags</i>	retardos temporais utilizados no modelo DELP
λ	termo de combinação da saída do DELP
LM	algoritmo <i>levenberg-marquardt</i>
MA	<i>moving average</i> – modelo de médias móveis
MAPE	<i>mean absolute percentage error</i> – erro médio percentual absoluto
$\max(\cdot)$	operação de máximo
<i>MaxHidden</i>	quantidade máxima de unidades de processamento na camada escondida da MLP
<i>MaxLags</i>	quantidade máxima de retardos temporais
$\text{mean}(\cdot)$	operação de média
MGA	<i>modified genetic algorithm</i> – algoritmo genético modificado
MGA-MLP	sistema híbrido composto de MGA e MLP
<i>MgaMaxGer</i>	quantidade máxima de gerações do MGA
$\min(\cdot)$	operação de mínimo
<i>minFit</i>	valor mínimo aceitável da função de avaliação
MANN	<i>minimax algebra neural network</i> – rede neural da álgebra minimax
MLP	<i>multilayer perceptron neural network</i> – rede neural <i>perceptron</i> multicamadas
<i>MlpMod</i>	parâmetro para definição do modelo da MLP
<i>MlpHiddenUnit</i>	parâmetro para definição da quantidade de unidades de processamento na camada escondida da MLP
<i>MlpTrain</i>	parâmetro para definição do algoritmo de treinamento da MLP
MNN	<i>morphological neural network</i> – rede neural morfológica
MM	<i>mathematical morphology</i> – morfologia matemática
MSE	<i>mean squared error</i> – erro médio quadrático
μ	taxa de aprendizagem do processo de aprendizagem do DELP
$\nabla J(\mathbf{w})$	gradiente de J em relação a \mathbf{w} utilizado no processo de aprendizagem do DELP

ω	saída do operador de erosão do DELP
OSSCG	algoritmo <i>one step secant conjugate gradient</i>
p	vetor de coeficientes do operador linear do DELP
PACF	<i>partial autocorrelation function</i> – função de auto-correlação parcial
PETR4	preço de fechamento das ações da petrobras PN
PFP	<i>phase fix procedure</i> – procedimento de correção de fase
POCID	<i>prediction of change in direction</i> – previsão de mudança na direção
PSO	<i>particle swarm optimizer</i> – otimizador de exame de partículas
PSO–MLP	sistema híbrido composto de PSO e MLP
<i>PsoMaxIter</i>	quantidade máxima de iterações do PSO
P_t	critério de parada <i>process training</i>
$Q(\mathbf{x})$	vetor de funções de impulso unitário
$Q_\sigma(\mathbf{x})$	vetor de funções de impulso suave
$q(x)$	função de impulso unitário
$q_\sigma(x)$	função de impulso suave
QUICKPROP	algoritmo <i>quick-propagation</i>
$\mathcal{R}_r(\mathbf{x})$	<i>r</i> -ésima função de posto do vetor \mathbf{x}
r_t	termo de ruído de uma série temporal no tempo t
RPROP	algoritmo <i>resilient back propagation</i>
RWD	<i>random walk dilemma</i> – dilema do passeio aleatório
RWM	<i>random walk model</i> – modelo de passeio aleatório
SCG	algoritmo <i>scaled conjugate gradient</i>
SETAR	<i>selfexcited threshold autoregressive</i> – modelo de limiar auto-regressivo auto-excitado
SGA	<i>standard genetic algorithm</i> – algoritmo genético padrão
SHIF	método <i>swarm-based hybrid intelligent forecasting</i>
σ	fator de escala da função de impulso suave
$std(\cdot)$	operação de desvio padrão
TAEF	método <i>time-delay added evolutionary forecasting</i>
TAR	<i>threshold autoregressive</i> – modelo de limiar auto-regressivo
THEIL	<i>u of theil statistics</i> – estatística u de theil
θ	termo de combinação do módulo não linear do DELP
USIM5	preço de fechamento das ações da usiminas PNA
VALE5	preço de fechamento das ações da vale PNA
X	série temporal
x	vetor de entrada do DELP
x_t	observação de uma série temporal no tempo t
$X \vee Y$	produto máximo entre os grupos ordenados de reticulado completo X e Y
$X \wedge Y$	produto mínimo entre os grupos ordenados de reticulado completo X e Y
y	saída do modelo DELP
w	vetor de pesos do DELP

CAPÍTULO 1

Introdução

Este capítulo apresenta uma breve introdução ao problema abordado, a previsão de séries temporais. Também são discutidos a motivação, a justificativa e os objetivos desta dissertação. Posteriormente, é apresentada a estrutura dos capítulos subsequentes desta dissertação.

1.1 Introdução

Atualmente a previsão do comportamento futuro de fenômenos temporais utilizando observações passadas tem despertado o interesse de diversos pesquisadores em diversas áreas de pesquisa, uma vez que a previsão permite a prevenção de possíveis adventos ou novidades que possam vir a acontecer no fenômeno temporal de interesse, e representa um elemento chave para o sucesso ou fracasso de um determinado processo de tomada de decisão [37, 40, 43, 46, 60]. Diversas aplicações necessitam de sistemas de previsão [25, 37, 40, 41, 43, 46, 47, 49, 60, 61], tais como planejamento de produção e financeiro, gerenciamento de estoque, escalonamento de pessoal, dentre outros. Neste contexto, é possível identificar que o objetivo central nestes sistemas é definido pela estimativa de eventos futuros associado à minimização do fator risco no processo de tomada de decisão.

Tipicamente, sistemas de previsão são responsáveis pela construção de mapeamentos capazes de estimar (aproximar) o futuro a partir de informações do passado e presente. Independente do sistema a ser utilizado, este possui algum tipo de imprecisão, seja por falhas na construção do sistema ou pela complexidade do fenômeno temporal. Note que a magnitude destas imprecisões depende da escolha propriamente dita do tipo de sistema de previsão e do seu custo, em termos de recursos, para geração das previsões. Assim, devido ao processo de previsão nunca poder eliminar totalmente as imprecisões, é necessário sua inclusão no processo de tomada de decisão, sendo definida por [25, 60, 61]

$$DA = DP + E, \quad (1.1)$$

onde DA é a decisão atual, DP é a decisão assumindo que a previsão está correta e E é a compensação devido a imprecisão no sistema de previsão. Este fato implica que um sistema de previsão poderia prover uma descrição para o erro de previsão tão bem quanto para a própria previsão [25, 60, 61].

O principal problema do uso dos sistemas de previsão em aplicações práticas está associado ao custo (recursos) para geração de previsões. Entretanto, a maximização dos recursos utilizados não implica, necessariamente, na minimização das imprecisões do sistema de previsão, e consequentemente na minimização do risco no processo de tomada de decisão [60, 61]. Portanto, o ganho obtido, em termos de desempenho de previsão, não cresce necessariamente na

mesma escala dos custos, isto é, a minimização das imprecisões alcançada por sistemas de previsão clássicos da literatura não justificam o alto custo necessário para geração das previsões.

Na literatura, é possível verificar que os procedimentos para a realização do processo de previsão podem ser divididos em duas categorias [60, 61]:

1. Automáticos: procedimentos que utilizam os dados históricos, referidos como séries temporais, de um determinado fenômeno temporal para a geração da previsão, através de modelos que podem ser desenvolvidos e implementados em computadores digitais;
2. Não Automáticos: procedimentos que exigem intervenção humana especializada, para serem aplicados, mantidos e analisados.

O foco desta dissertação está relacionado com o desenvolvimento de modelos automáticos para previsão de séries temporais, na tentativa prever valores futuros de um determinado fenômeno temporal baseado em seus valores presentes e passados.

1.2 Motivação e Justificativa

Nos tempos modernos, o maior bem que existe é a informação. Em particular, no mundo da economia e finanças, um volume incomensurável de informações tem sido armazenadas com relação às mais diversas transações realizadas. Estas informações dizem respeito à transações bancárias efetuadas pelas empresas e seus clientes, dados de compras e vendas de produtos e serviços, dados de acompanhamento médico sobre pacientes, pesquisas de opiniões, etc [25, 28, 32, 33, 35, 36, 60, 61].

Nestes dados, escondem-se informações relevantes de importância estratégica para o processo de tomada de decisão [25, 28, 32, 33, 35, 36, 60, 61]. Além disso, a descoberta deste tipo particular de informações pode permitir, dentre outras coisas, a previsão do comportamento de ações e índices financeiros em bolsas de valores e a previsão de variações em taxas de câmbio. Desta forma, a previsão pode ser considerada um elemento chave na tomada de decisão [25–27, 29, 30, 38, 39, 60, 61], uma vez que qualquer decisão depende dos eventos que a antecederam.

Neste contexto, o desenvolvimento de soluções para previsão de séries temporais é considerado um problema bastante difícil de ser solucionado, uma vez que em tais séries, frequentemente, estão incluídas características bastante complexas como irregularidades, volatilidade, tendências e ruído [25–30, 32, 33, 35–41, 43, 46, 47, 60, 61]. Diversas abordagens, provenientes de várias áreas do conhecimento humano, têm sido estudadas para o desenvolvimento de modelos de previsão na tentativa de mapear o comportamento futuro de um dado fenômeno temporal [6, 74, 75, 81, 84, 98, 101, 115, 136, 151]. Em particular, modelos lineares e não-lineares provenientes da matemática e estatística foram propostos na tentativa de solucionar o problema [10, 99, 111, 116, 128].

Uma classe particular dos modelos lineares bastante popular na literatura, os modelos autoregressivos integrados de médias móveis (*autoregressive integrated moving average*, ARIMA) [10], são utilizados na prática em previsão de séries temporais. No entanto, como os modelos ARIMA são lineares e uma grande maioria das aplicações de séries temporais são governadas

por fenômenos não-lineares, tal fato introduz uma limitação nas previsões geradas por modelos puramente lineares [25–27, 29, 30, 38, 39, 49, 60, 61].

Na tentativa de superar as limitações dos modelos lineares, diversas abordagens estatísticas não-lineares têm sido desenvolvidas para previsão de séries temporais, como os modelos heterocedásticos condicionais auto-regressivos (*autoregressive conditional heteroskedasticity*) [57], os modelos bilineares (*bilinear models*) [116], os modelos de limiar auto-regressivo (*threshold autoregressive models*) [99], e os modelos de estados gerais dependentes (*general state dependent models*) [128], os modelos auto-regressivos exponenciais (*exponential autoregressive models*) [111], dentre outros. No entanto, de acordo com o trabalho de Clements *et al.* [14], a não-linearidade de tais modelos não implica, na prática, em um ganho significativo em termos de desempenho preditivo do modelo quando comparado ao desempenho de modelos estatísticos lineares, uma vez que o alto custo para geração dos modelos não-lineares não é justificado para se obter um desempenho similar aos modelos estatísticos lineares. Além disso, a sua alta complexidade matemática impede o desenvolvimento de um sistema automático de previsão, uma vez que existe a necessidade de atuação de um especialista no problema para validar as previsões geradas pelo modelo [14].

Alternativamente, as redes neurais artificiais (*artificial neural networks*, ANNs) [74, 75] têm sido aplicadas com sucesso para a modelagem não-linear de alguns tipos de séries temporais [16, 25, 28, 33, 35, 36, 41, 47, 60, 61, 136, 165, 166]. No entanto, as ANNs necessitam da definição de um conjunto de parâmetros que é bastante difícil de se otimizar. A topologia, a quantidade de unidades de processamento na camada escondida, e o algoritmo de treinamento (e suas variáveis correspondentes) são apenas exemplos destes parâmetros que necessitam ser otimizados [25, 28, 33, 35, 36, 41, 47, 60, 61]. Além disso, no caso particular de séries temporais, outro elemento crucial que requer definição são os retardos temporais relevantes para uma correta caracterização da série temporal [25–30, 32, 33, 35–41, 43, 46, 47, 49, 60, 61].

Um caso particular de ANNs que tem obtido relativo sucesso como solução para diversos problemas são as redes neurais morfológicas (*morphological neural networks*, MNNs) [37, 42–44, 104, 106, 120, 122, 139, 140, 147, 148, 150, 151, 153, 160], sendo baseadas em conceitos fundamentais da morfologia matemática (*mathematical morphology*, MM) [89, 132] sob o contexto de teoria de reticulados (*lattice theory*) [4, 76, 127, 133]. As MNNs diferem das ANNs clássicas no sentido em que a computação realizada por cada neurônio é determinada por operadores básicos da MM, as quais têm apresentado desempenho superior para problemas de previsão de séries temporais [26, 27, 29, 30, 37–40, 43, 46].

Neste contexto, diversas metodologias híbridas, compostas por algoritmos evolucionários (*evolutionary algorithms*, EAs) [19, 72, 85, 162, 164] combinados com ANNs [74, 75] ou MNNs [104, 106, 120, 122, 139, 140, 147] têm sido propostas na literatura. Tais métodos utilizam EAs para selecionar os melhores retardos temporais (*time lags*) para uma correta caracterização do fenômeno de uma série temporal, bem como para otimizar o processo de escolha dos parâmetros, estrutura e arquitetura dos modelos de previsão, combinando técnicas evolucionárias e técnicas baseadas em gradiente descendente [25–30, 32, 33, 35–41, 43, 46, 47, 49, 60, 61]. No entanto, é possível verificar que tais metodologias híbridas possuem um custo bastante significativo, caracterizado em termos de tempo computacional, uma vez que utilizam técnicas que necessitam de quantidades significativas de recursos computacionais para geração de um

modelo de previsão confiável e preciso, gerando assim uma limitação para a utilização prática destas metodologias, uma vez que o aumento do custo não implica no aumento significativo da precisão da previsão [26, 27, 29, 30, 38, 39].

Mesmo com as técnicas sofisticadas e metodologias híbridas que têm sido recentemente propostas na literatura para solucionar o problema de previsão de séries temporais, uma limitação surge de todos estes modelos de previsão no caso particular de previsão de séries temporais financeiras, conhecido como dilema do passeio aleatório (*random walk dilemma*, RWD), sendo recentemente reportado na literatura [25–30, 32, 33, 35–41, 43, 46, 47, 49, 60, 61, 88, 136]. Este dilema ocorre quando se tenta prever qualquer série temporal financeira utilizando um modelo arbitrário de previsão, onde conseqüentemente as previsões geradas apresentam um característico atraso de um passo a frente em relação aos valores reais da série temporal, ou seja, possuem uma distorção de fase temporal na reconstrução do espaço de fase de um determinado fenômeno temporal proveniente do mercado financeiro [25–27, 29, 30, 38, 39, 60, 61]. A definição formal deste problema será apresentada no decorrer dos capítulos desta dissertação.

Devido a este fato, alguns pesquisadores argumentaram que séries temporais financeiras são geradas por processos puramente aleatórios e não podem ser previstas [88, 136]. Entretanto, trabalhos recentes [25–27, 29, 30, 38, 39, 60, 61] mostraram que séries temporais financeiras são geradas por componentes não-lineares complexas subdominantes combinadas a uma componente linear dominante (comportamento encontrado quando a ordem dos retardos temporais analisados é significativamente pequena). Mas, quando esta ordem é significativamente incrementada, a componente não-linear complexa se torna dominante sobre a componente linear, podendo ser facilmente identificada. Este argumento sustenta o fato que estas séries não são geradas por processos puramente aleatórios e podem ser eficientemente previstas [25–27, 29, 30, 38, 39, 60, 61].

Também, outra observação interessante foi encontrada por Araújo [26, 27, 29, 30, 38, 39], o qual demonstrou que tais componentes podem ser eficientemente aproximadas em termos de uma combinação balanceada entre operadores lineares de resposta finita ao impulso (componente linear) e operadores não-lineares crescentes (componente não-linear). Desta forma, grandes esforços ainda necessitam ser aplicados para o desenvolvimento de modelos híbridos lineares e não-lineares para previsão de séries temporais financeiras, uma vez que estes novos resultados apresentados por [26, 27, 29, 30, 38, 39] mostram novos caminhos, com o uso da MM sob o contexto de teoria de reticulados, a serem traçados nesta área de pesquisa.

1.3 Objetivos

Este trabalho se concentrou no desenvolvimento de uma classe particular de *perceptrons* híbridos baseada nos princípios da morfologia matemática [89, 132] no contexto de teoria de reticulados [4, 76, 127, 133] para superar o dilema do passeio aleatório no problema de previsão de séries temporais financeiras. O modelo proposto, chamado de *perceptron* de dilatação-erosão-linear (*dilation-erosion-linear perceptron*, DELP), consiste de uma combinação linear entre operadores não-lineares (do tipo morfológicos) e um operador linear (do tipo resposta finita ao impulso), tendo a habilidade de determinar automaticamente o percentual balanceado de utilização das componentes linear e não-linear do modelo.

Para o projeto do DELP (processo de aprendizagem), foi apresentado um método de gra-

diente descendente (*gradient steepest descent method*) [74, 75] utilizando ideias do algoritmo de retropropagação do erro (*back propagation*, BP) [74, 75] e uma abordagem sistemática para superar o problema da não-diferenciabilidade das operações morfológicas, baseado no trabalho de Pessoa e Maragos [105, 106] e Sousa [139, 140]. Também, no processo de aprendizagem do DELP, foi incluída uma etapa adicional, definida como procedimento de correção de fase (*phase fix procedure*) [25–27, 29, 30, 38, 39, 60, 61], que tem o intuito de ajustar distorções de fase temporais que ocorrem na reconstrução do espaço de fase de fenômenos temporais financeiros.

Uma análise experimental foi conduzida com o modelo proposto utilizando um conjunto de séries temporais provenientes do mercado financeiro nacional e internacional (Índice da Bolsa de Valores de São Paulo, Índice *Dow Jones Industrial Average*, Índice *National Association of Securities Dealers Automated Quotation*, Índice *Financial Times and London Stock Exchange 100*, Preço das ações do Bradesco PN, da Gol PN, do Itaú Unibanco PN, da Petrobras PN, da Usiminas PNA e da Vale PNA). Foram empregadas cinco métricas (erro médio quadrático, erro médio percentual absoluto, estatística *u* de theil, previsão de mudança na direção e variância relativa média) e uma função de avaliação para mensurar o desempenho preditivo do modelo proposto na tentativa de consolidar o DELP como um modelo robusto e eficaz.

Portanto, o objetivo geral desta dissertação é propor um modelo de previsão híbrido para superar o dilema do passeio aleatório em séries temporais financeiras. Para alcançar este objetivo geral foi estabelecido o seguinte conjunto de objetivos específicos:

1. Estudo e análise dos modelos de previsão clássicos lineares e não-lineares, e modelos de previsão híbridos clássicos e otimizados, baseados em redes neurais artificiais combinadas com algoritmos evolucionários, para geração de um referencial teórico;
2. Determinação de um conjunto diversificado de séries temporais financeiras para estudo, análise e teste:
 - Índice da Bolsa de Valores de São Paulo;
 - Índice *Dow Jones Industrial Average*;
 - Índice *National Association of Securities Dealers Automated Quotation*;
 - Índice *Financial Times and London Stock Exchange 100*;
 - Preço das ações do Bradesco PN;
 - Preço das ações da Gol PN;
 - Preço das ações do Itaú Unibanco PN;
 - Preço das ações da Petrobras PN;
 - Preço das ações da Usiminas PNA;
 - Preço das ações da Vale PNA.
3. Investigação teórica de morfologia matemática e teoria de reticulados para o desenvolvimento do *perceptron* de dilatação-erosão-linear (*dilation-erosion-linear perceptron*, DELP);

4. Desenvolvimento de uma metodologia para superar a não-diferenciabilidade dos operadores morfológicos de dilatação e erosão que serão utilizados no algoritmo de treinamento do DELP;
5. Desenvolvimento do algoritmo de treinamento com ajuste de fase temporal para projetar o DELP;
6. Realização de uma análise experimental com o modelo DELP a partir de gráficos, de cinco métricas consolidadas na literatura e de uma função de avaliação para medição do desempenho dos modelos de previsão investigados.

1.4 Estrutura da Dissertação

A estrutura desta dissertação é composta por sete capítulos descritos a seguir:

Capítulo 1 - Introdução: neste capítulo foi elaborada uma breve introdução do problema de previsão de séries temporais, definindo alguns termos básicos e características peculiares do problema abordado. Também, foram apresentados as motivações e objetivos desta dissertação;

Capítulo 2 - O Problema de Previsão de Séries Temporais: neste capítulo é apresentada a definição formal de séries temporais, do principal problema de previsão e do dilema do passeio aleatório para previsão financeira;

Capítulo 3 - Modelos para Previsão de Séries Temporais: neste capítulo são conceituados os modelos clássicos lineares, não-lineares e híbridos utilizados na literatura para solucionar o problema de previsão de séries temporais;

Capítulo 4 - Caracterização das Séries Temporais Utilizadas: neste capítulo são definidas as séries temporais financeiras utilizadas nesta dissertação a fim de testar os modelos estudados em diferentes situações práticas. Para cada série estudada, é apresentada uma breve descrição do conjunto de dados. Ao final do capítulo, é definido um conjunto de cinco métricas clássicas da literatura de previsão de séries temporais e uma função de avaliação, que serão posteriormente utilizadas nesta dissertação.

Capítulo 5 - O Modelo Proposto: neste capítulo são apresentados os conceitos básicos de morfologia matemática e teoria de reticulados, os fundamentos para superar o problema da não-diferenciabilidade das operações morfológicas, a definição formal do DELP proposto e do seu algoritmo de treinamento com ajuste de fase temporal.

Capítulo 6 - Simulações e Resultados Experimentais: neste capítulo é descrito o procedimento empregado para realização das simulações. Também, são apresentados os resultados obtidos de forma a permitir uma comparação do modelo proposto com os modelos estudados neste trabalho. Estes resultados foram analisados graficamente, bem como através das métricas e da função de avaliação definidas no Capítulo 4.

Capítulo 7 - Conclusões: neste capítulo final são apresentadas as considerações finais e as conclusões desta dissertação. Também, é feita uma discussão sobre o modelo DELP desenvolvido, bem como suas limitações e sugestões para trabalhos futuros.

O Problema de Previsão de Séries Temporais

Neste capítulo, são apresentadas as definições formais de séries temporais, do problema de previsão de séries temporais e do dilema do passeio aleatório para previsão de séries financeiras.

2.1 Introdução

Uma série temporal [10] é definida como uma sequência de amostras de um dado fenômeno temporal de interesse. Este conjunto de dados pode ser obtido a partir de observações de tal evento, que representam pontos temporais discretos ou contínuos, usualmente equidistantes. Considerando uma determinada série temporal discreta no tempo, sendo obtida continuamente durante um intervalo de tempo de amplitude T e, sendo Δt o intervalo de amostragem, a quantidade de pontos da série é definida por

$$N = \frac{T}{\Delta t}. \quad (2.1)$$

Assim, uma série temporal pode ser representada por

$$\mathbf{X} = \{x_t \in \mathbb{R} \mid t = 1, 2, 3 \dots N\}, \quad (2.2)$$

onde t é um índice temporal e N é o número de observações ou pontos da série.

Note que \mathbf{X} representa uma sequência de observações que obedecem uma ordem temporal definida por um índice cronológico t , denominado tempo, que definirá a granularidade das observações [60, 61]. O intervalo de amostragem, definido pelo tempo entre duas observações consecutivas, é comumente determinado pela disponibilidade dos dados, onde espera-se que exista uma quantidade mínima de intervalos possíveis de forma a representar mais precisamente o fenômeno temporal em questão.

Séries temporais são compostas por vários componentes ou características, que podem ser identificadas isoladas ou em conjunto, tais como:

1. Ciclos: representada por movimentações cíclicas na observação do fenômeno temporal sem uma definição clara de sua frequência;
2. Sazonalidade: similar à componente cíclica, embora as flutuações possuam caráter sazonal, isto é, com sua frequência bem definida;
3. Tendência: representada pelo incremento ou decremento gradual sistemático das observações do fenômeno temporal;

4. Flutuações Aleatórias: representados pelas imprecisões ou erros na observação do fenômeno temporal (ruído inerente ao processo de observação).

Portanto, ao se analisar uma série temporal procura-se identificar cada uma destas componentes ou características para que se tenha uma boa estimativa da contribuição de cada uma em particular em relação ao todo. Assim, o principal objetivo do estudo e análise de séries temporais é como modelar o fenômeno sob observação e posteriormente avaliar o desempenho preditivo em termos estatísticos.

No entanto, vários cuidados devem ser tomados no processo de análise de séries temporais [49, 60, 61]:

Planejamento: a coleta de dados deve ser planejada, onde o intervalo de amostragem e a quantidade de observações do fenômeno devem ser previamente determinadas segundo o foco e metas do trabalho em questão.

Estacionariedade: a primeira suposição sobre uma série temporal é em relação a sua estacionariedade, ou seja, se ela se desenvolve aleatoriamente no tempo em torno de uma média e desvio padrão constantes. No entanto, na prática, séries temporais apresentam algum tipo de não estacionariedade, e portanto devem ser cuidadosamente analisadas sob a amostra em questão.

Transformações: uma grande maioria dos processos estatísticos para previsão de séries temporais supõe que as séries são estacionárias, mas como na prática acontece o contrário, é necessário que sejam feitas transformações nos dados, na tentativa de tornar a série mais estacionária possível.

Irregularidades: séries temporais são normalmente observadas em instantes de tempo igualmente espaçados. No entanto, mesmo definindo o intervalo de amostragem, o conjunto de dados pode ter certas irregularidades, ou perda de informação, seja por quebra do instrumento de observação ou por falha operacional. Portanto, no caso da quantidade de irregularidades ser insignificante, este problema pode ser resolvido através de interpolação. Caso contrário, técnicas estatísticas devem ser empregadas e estão fora do escopo deste trabalho.

Outliers: uma vez que o processo de previsão depende de observações, que dependem de pessoas e equipamentos, é possível identificar no conjunto de valores observados, dados bastante diferentes da maioria dos outros valores, conhecidos como *outliers*. Tais valores podem ser erros no processo de observação ou pode ser um valor real, que pode vir a identificar uma nova característica ou padrão da série temporal, sendo muito importante verificar a natureza de tais acontecimentos e tratá-los de forma devida.

2.2 Definição do Problema de Previsão

O objetivo de se aplicar técnicas de previsão a uma dada série temporal X é construir um mapeamento que seja capaz de prever o futuro [25, 28, 33, 36, 37, 41, 43, 47, 49, 60, 61]. Em

outras palavras, o objetivo das técnicas de previsão é criar um mecanismo que permita, com uma certa precisão, a previsão dos valores futuros da série, dados por x_{t+h} , onde h representa o horizonte de previsão de h passos a frente.

Uma classe de decisões importantes, no sentido temporal, para o desenvolvimento de modelos preditivos envolve os seguintes elementos:

1. Período da previsão;
2. Horizonte da previsão;
3. Intervalo da previsão.

O período da previsão é a unidade básica de tempo na qual as previsões são realizadas. O horizonte da previsão corresponde ao número de períodos cobertos, no futuro, pela previsão. Desta forma, é possível requerer uma previsão para as próximas dez semanas, com uma análise semanal, assim o horizonte seria dez semanas e o período de uma semana. Finalmente, o intervalo da previsão é a frequência com a qual novas previsões são preparadas. Vale mencionar que, de acordo com Mattos [49], é importante considerar a disponibilidade de dados históricos sobre o problema, a precisão e a representatividade estatística de tais dados, sendo estas características essenciais para o sucesso no processo de previsão.

Em relação ao horizonte de previsão, dependendo do valor assumido para h , é possível classificar as previsões dos valores futuros de uma série temporal como de curto, médio ou longo prazo. Portanto, é possível caracterizar os modelos de previsão, em relação ao seu horizonte de previsão, como:

1. Multi-passo: são adotadas valores de $h > 1$, onde o objetivo é se identificar tendências gerais e pontos de inflexão relevantes na série temporal.
2. Um-passo: a previsão é feita apenas para o período de tempo imediatamente posterior ao atual, ou seja, o parâmetro $h = 1$. Neste trabalho considera-se apenas previsões utilizando esta abordagem.

Neste contexto, a ideia básica do problema de previsão é a determinação de uma janela temporal, que contém pontos do passado e do presente da série, e assim utilizar um modelo de previsão com o objetivo de determinar os seus pontos futuros. Esta janela temporal é formada por um conjunto de observações de um determinado fenômeno temporal, trazendo consigo informações para a determinação das observações futuras do fenômeno gerador da série, onde tal conjunto de observações ou pontos contidos na janela temporal são referidos como retardos temporais (*time lags*) [25, 28, 33, 36, 41, 49, 60, 61].

Portanto, um elemento chave para garantir um bom desempenho no processo de previsão de um determinado fenômeno temporal é a escolha correta dos retardos temporais relevantes para a caracterização de tal fenômeno. Desta forma, quando um determinado retardo temporal é relevante para a série, este irá gerar algum tipo de relacionamento entre os dados históricos, de tal maneira a possibilitar uma correta caracterização das leis que governam o fenômeno gerador da série [25, 28, 33, 36, 41, 49, 60, 61].

O problema de previsão de séries temporais pode ser definido de uma forma alternativa [155]. Seja M um espaço de fase. Seja $\varphi(x)$ uma trajetória temporal definida pelo fenômeno de interesse sobre o espaço de fase M . Seja $f(\varphi(x))$ a observação de cada ponto relativo a evolução temporal deste fenômeno, isto é, a própria série temporal. Assim, pode-se verificar que o problema de previsão é a determinação correta do mapeamento $\Phi_{(\varphi,f)} : M \rightarrow \mathbb{R}^d$ entre as observações dos pontos de um dado fenômeno e o espaço de fase M , sendo este mapeamento definido por [49, 60, 61]

$$\Phi_{(\varphi,f)} = [f(x), f(\varphi(x)), \dots, f(\varphi^d(x))], \quad (2.3)$$

em que d é a dimensão do espaço de fase M .

Note que os relacionamentos entre as observações de um determinado fenômeno temporal constituem um espaço de fase d -dimensional, onde d representa a dimensão mínima capaz de representar, com uma certa precisão, tais relacionamentos. Assim, tal espaço de fase d -dimensional pode ser reconstruído de tal forma a conter informação necessária para uma correta representação de uma determinada série temporal, fato que pode ser comprovado com o trabalho de Takens [155], que provou, matematicamente, que se d é suficientemente grande, tal espaço de fase reconstruído é topologicamente equivalente ao espaço de fase original que gerou a série temporal [25, 28, 33, 36, 37, 41, 43, 47, 49, 60, 61], isto é, a dinâmica deste espaço é topologicamente idêntica à dinâmica do espaço de fase que originalmente gerou o sistema.

O trabalho de Takens [155] é, então, a justificativa teórica da possibilidade de reconstrução de um espaço de fase utilizando os retardos temporais relevantes para a caracterização de uma determinada série. Levando em consideração tais aspectos, é possível concluir que o principal problema na reconstrução do espaço de fase que gera um fenômeno temporal é naturalmente a escolha da dimensionalidade do espaço de fase d , ou mais precisamente, a escolha dos retardos temporais relevantes necessários para uma correta caracterização da dinâmica do sistema [25–27, 29, 30, 38, 39, 49, 60, 61].

De acordo com Box e Jenkins [10], quando existe apenas relacionamentos lineares entre as observações do fenômeno temporal, a função de auto-correlação (*autocorrelation function*, ACF) [10] e a função de auto-correlação parcial (*partial autocorrelation function*, PACF) [10] são capazes de identificar as dependências relevantes entre tais observações, ou seja, têm a habilidade de definir os retardos temporais relevantes para representar corretamente a série temporal. Entretanto, quando se analisa séries temporais reais, ou mais especificamente, séries temporais complexas com todas as suas dependências e variáveis incontrolláveis, o relacionamento que envolve os dados históricos da série é geralmente não-linear, fato que torna a análise e definição automática dos retardos temporais relevantes para uma correta caracterização do fenômeno temporal gerador da série temporal um processo bastante complexo [25–27, 29, 30, 38, 39, 49, 60, 61].

Assim, vários métodos foram propostos na literatura para a definição de retardos temporais relevantes em séries temporais [109, 131, 156]. Tais métodos se baseiam no cálculo de probabilidades condicionais, considerando que séries temporais sejam representadas por

$$x_t = f(x_{t-1}, x_{t-2}, \dots, x_{t-d}) + r_t, \quad (2.4)$$

onde $f : \mathbb{R}^d \rightarrow \mathbb{R}$ representa uma função que realiza o mapeamento entre os pontos do passado $(x_{t-1}, x_{t-2}, \dots, x_{t-d})$ e do futuro (x_t) . O termo r_t representa um termo de ruído. O termo r_t

decrece com o incremento de d , bem como se o sistema for completamente determinístico, o termo r_t tende a zero quando d excede a dimensão embutida mínima necessária para uma correta descrição do sistema gerador da série temporal [25–27, 29, 30, 38, 39, 49, 60, 61].

Vale mencionar a existência de uma metodologia, conhecida como *lagplot* ou *phase portrait* [81, 102], que é aplicada para determinação e análise das relações entre os retardos temporais relevantes de uma determinada série temporal. O *lagplot* pode ser definido como gráficos de dispersão relacionados com os diferentes retardos temporais da série ($x_t \times x_{t-1}, x_t \times x_{t-2}, \dots, x_t \times x_{t-d}$). Desta forma, a partir deste tipo de análise, é possível caracterizar relevância em um determinado retardo temporal em questão quando se verifica a existência de alguma estrutura bem definida em seu *lagplot*. No entanto, tal técnica é bastante limitada, haja vista que ela depende de interpretação humana dos gráficos, e nem sempre as relações não-lineares contidas nestes gráficos são humanamente simples de se identificar (a medida que a dimensionalidade d aumenta). Entretanto, sua simplicidade é um forte argumento para sua utilização [25–30, 33, 36, 38, 39, 41, 49, 60, 61].

2.3 Definição do Dilema do Passeio Aleatório

Uma estratégia simples de previsão é definir a última observação da série temporal associado a um termo de ruído como a melhor previsão do valor futuro. Este tipo de modelo é conhecido como modelo de passeio aleatório (*random walk model*, RWM) [94, 136], que é formalmente definido por

$$x_t = x_{t-1} + r_t, \quad (2.5)$$

onde x_t é a observação atual, x_{t-1} é a observação imediatamente anterior a x_t , e r_t é um termo de ruído com distribuição Gaussiana com média zero e desvio padrão dada por σ ($\mathcal{N}(0, \sigma)$). Em outras palavras, de acordo com o RWM, o termo de ruído da série temporal é definida em termos de uma componente aleatória, isto é, um ruído branco (*white noise*) [25, 28, 33, 36, 41, 49, 60, 61].

Este modelo descrito acima claramente implica que, como o conjunto de informações consiste nos dados passados da série temporal e o termo de ruído é um ruído branco, os dados futuros da série não serão passíveis de previsão. Em média, o valor x_t é o melhor valor para x_{t-1} , uma vez que x_{t-1} representa a previsão com menor custo e um desempenho aceitável. Este comportamento é característico em mercados financeiros e na economia mundial, sendo formalmente conhecido como dilema do passeio aleatório (*random walk dilemma*, RWD) ou hipótese do passeio aleatório (*random walk hypothesis*, RWH) [94, 136].

Como o custo computacional para previsão de séries temporais financeiras usando o RWM é extremamente baixo, é de se esperar que qualquer modelo de previsão mais custoso que um RWM tenha, obrigatoriamente, desempenho superior. Entretanto, o que se verifica, no caso particular das séries temporais financeiras, é que às leis que governam seu fenômeno gerador são bastante similares às leis que governam o fenômeno gerador do RWM. Isto implica que qualquer modelo aplicado a este tipo de série tenderá a ter um desempenho similar ao RWM, isto é, a previsão gerada por qualquer modelo terá, obrigatoriamente, um atraso de um passo em relação aos valores reais da série. Tal afirmação está de acordo com os resultados obtidos por [94, 136] e a prova desta afirmação é desenvolvida a seguir [25–30, 33, 36, 38, 39, 41, 49, 60, 61].

Assumindo que um modelo eficaz de previsão seja utilizado para construir um valor estimado de x_t , dado por \hat{x}_t , o valor esperado ($E[\cdot]$) da diferença entre \hat{x}_t e x_t deve tender a zero

$$E[\hat{x}_t - x_t] \rightarrow 0. \quad (2.6)$$

Se o fenômeno gerador da série temporal for composto por componentes lineares dominantes e componentes não-lineares sub-dominantes (definida por $g(t)$), e assumindo que $E[r_t] = 0$, o valor esperado da diferença entre \hat{x}_t e x_t (assumindo que $x_t = x_{t-1} + g(t) + r_t$, ou seja, que a série não é gerada por um processo puramente aleatório) será

$$E[\hat{x}_t - (x_{t-1} + g(t) + r_t)] \rightarrow 0, \quad (2.7)$$

$$E[\hat{x}_t] - E[x_{t-1}] - E[g(t)] - E[r_t] \rightarrow 0, \quad (2.8)$$

$$E[\hat{x}_t] - E[x_{t-1}] - E[g(t)] \rightarrow 0, \quad (2.9)$$

$$E[\hat{x}_t] \rightarrow E[x_{t-1}] + E[g(t)]. \quad (2.10)$$

Como $E[g(t)] \rightarrow 0$, devido a magnitude da componente não-linear sub-dominante em relação a magnitude da componente linear dominante, temos que

$$E[x_{t-1}] + E[g(t)] \rightarrow E[x_{t-1}]. \quad (2.11)$$

Portanto, a Equação (2.10) pode ser redefinida, e dada por

$$E[\hat{x}_t] \rightarrow E[x_{t-1}]. \quad (2.12)$$

Desta forma, a Equação (2.12) garante que qualquer modelo de previsão arbitrário aplicado ao problema de previsão de séries temporais financeiras, terá em suas previsões um característico atraso de um passo a frente em relação aos valores reais da série. Portanto, nestas condições, superar o RWD é uma tarefa bastante complexa, pois como o fenômeno gerador de séries temporais financeiras é bastante similar ao fenômeno gerador do RWM, qualquer modelo de previsão tenderá a ser guiado por esse comportamento [136].

Indicações destas características em séries temporais financeiras foram recentemente reportadas na literatura [25, 28, 33, 36, 41, 49, 60, 61] e podem ser observadas nas séries temporais estudadas neste trabalho a partir dos gráficos *lagplots* que serão posteriormente apresentados no Capítulo 4. Nestes trabalhos, foi mostrado que séries temporais financeiras são geradas por componentes não-lineares complexas sub-dominantes e uma componente linear dominante (comportamento encontrado quando a ordem dos retardos temporais analisados é significativamente pequena). Mas, quando esta ordem é significativamente incrementada, a componente não-linear complexa se torna dominante sobre a componente linear, podendo ser facilmente identificada [25, 28, 33, 36, 41, 49, 60, 61].

Note que, caso as séries financeiras fossem geradas puramente por RWMs, quando se aumentasse a ordem dos retardos temporais analisados via *lagplot* não seria possível identificar qualquer estrutura não-linear bem definida. Fato que não está de acordo com a análise do *lagplot* de séries temporais financeiras no Capítulo 4, onde claramente pode-se identificar relacionamentos não-lineares em retardos de alta ordem. Portanto, este trabalho assume que séries temporais financeiras são séries complexas geradas a partir de $x_t = x_{t-1} + g(t) + r_t$, sendo compostas por componentes lineares dominantes e componentes não-lineares sub-dominantes. Portanto, este argumento sustenta o fato que estas séries não são geradas por RWM e podem ser eficientemente previstas [25, 28, 33, 36, 41, 49, 60, 61].

2.4 Resumo do Capítulo

Neste capítulo foram apresentados os conceitos básicos relacionados a séries temporais. Foi apresentado o problema de previsão de séries temporais, tomando como base o trabalho de Takens [155], que provou matematicamente a possibilidade de reconstrução de um espaço de fase utilizando os retardos temporais relevantes para uma correta caracterização de uma determinada série temporal. Também, foram apresentados argumentos que sustentam o fato de séries temporais financeiras não serem geradas por processos puramente aleatórios, uma vez que estas possuem componentes lineares dominantes e componentes não-lineares sub-dominantes, bem como foram apresentadas demonstrações matemáticas que indicam o motivo pelos quais os modelos de previsão clássicos apresentarem em suas previsões um característico atraso de um passo a frente em relação aos valores reais da série.

Modelos para Previsão de Séries Temporais

Neste capítulo serão definidos os modelos clássicos para previsão de séries temporais apresentados na literatura, como modelos estatísticos lineares e não-lineares, redes neurais *perceptron* multicamadas, modelos de previsão híbridos com redes neurais e algoritmos evolucionários, e modelos híbridos otimizados.

3.1 Introdução

O processo de previsão de séries temporais consiste em representar as características da série temporal em um modelo que possa generalizar tais características temporais, sendo necessário que este modelo alcance uma boa representação das observações em qualquer instante de tempo da série, sendo tal instante pertencente ao passado, presente ou futuro [28, 33, 36, 41].

Apesar da maioria dos modelos de previsão considerarem apenas a análise das observações de uma dada série temporal, existem alguns modelos que consideram a inclusão de outras séries temporais para análise de uma série de interesse. Desse modo, é possível encontrar na literatura três classes de modelos para previsão de séries temporais [49]:

- **Univariados:** são os mais difundidos para solucionar o problema de previsão de séries temporais. Nesta classe, os modelos consideram a utilização de uma única série para geração das previsões, que estão fortemente relacionadas com as informações contidas na série histórica de interesse (modelos clássicos), ou que podem incorporar outras informações supostamente relevantes e que não estão diretamente contidas na série histórica (modelos bayesianos);
- **Função de Transferência:** são modelos que consideram outras séries para a geração das previsões, onde as outras séries incluídas no processo de geração devem, obrigatoriamente, ser não correlatas entre si;
- **Multivariados:** são modelos que também consideram outras séries para geração das previsões, com a diferença que tais séries não estão necessariamente relacionadas, ou seja, sem qualquer imposição em relação à causalidade entre essas séries.

Independente da classificação quanto ao tipo de modelo, existem diversos modelos de previsão para descrever o comportamento de uma série temporal. A escolha do modelo mais adequado é dependente de vários fatores, tais como o comportamento do fenômeno ou o conhecimento *a priori* da sua natureza e do objetivo da análise temporal [25–30, 33, 36, 38, 39, 41,

49,60,61]. Neste sentido, os modelos clássicos de previsão são baseados em funções temporais algébricas ou transcendentais, podendo ser representados de forma geral por [49, 60]

$$x_t = b_1 f_1(t) + b_2 f_2(t) + \cdots + b_i f_i(t) + r_t \quad (3.1)$$

onde b_i e $f_i(t)$ ($i = 1, 2, \dots, I$) representam, respectivamente, os parâmetros constantes e funções matemáticas de t . O termo r_t representa uma componente aleatória ou ruído.

Ocasionalmente, existem outras formas de modelagem de uma série temporal, onde \mathbf{X} é modelada como uma função das componentes aleatórias ordenadas temporalmente, do presente para o passado ($r_t, r_{t-1}, r_{t-2}, \dots$). Esta forma de representação é conhecida como “modelos de filtros lineares de resposta finita”, os quais são amplamente aplicados quando as observações são altamente correlacionadas [10]. Desta forma, x_t pode ser definido por [49, 60]

$$x_t = m + y_0 r_t + y_1 r_{t-1} + y_2 r_{t-2} + \cdots + y_i r_{t-i}, \quad (3.2)$$

onde m e y_i ($i = 1, 2, \dots, I$) são termos constantes.

Em resumo, o processo de previsão de séries temporais consiste basicamente no projeto do comportamento futuro de um determinado fenômeno temporal [25–30, 33, 36, 38, 39, 41, 49, 60, 61].

3.2 Modelos Estatísticos Lineares

Na literatura, vários modelos foram propostos com o objetivo de resolver o problema de previsão de séries temporais, onde pode-se verificar que uma grande maioria deles é linear, dentre os quais os modelos de Box e Jenkins [10] recebem um destaque especial, haja vista que são os mais populares e comumente utilizados para a resolução de situações reais de previsão de séries temporais financeiras. Antes de apresentar formalmente o modelo de Box e Jenkins [10], é necessário definir o que são modelos estatísticos lineares auto-regressivos e de médias móveis.

Um modelo auto-regressivo (*autoregressive*, AR) é definido por [49, 60]

$$\tilde{x}_t = \phi_1 \tilde{x}_{t-1} + \phi_2 \tilde{x}_{t-2} + \cdots + \phi_p \tilde{x}_{t-p} + r_t, \quad (3.3)$$

onde $\tilde{x}_i = x_i - \mu$, sendo μ definido como o nível médio da série. Os termos ϕ_i ($i = 1, 2, \dots, p$) representam os coeficientes auto-regressivos.

Um modelo de médias móveis (*moving average*, MA) é definido por [49, 60]

$$x_t = \mu + r_t - \theta_1 r_{t-1} - \theta_2 r_{t-2} - \cdots - \theta_q r_{t-q}. \quad (3.4)$$

Considerando o fato que $\tilde{x}_i = x_i - \mu$, temos [49, 60]

$$\tilde{x} = (1 - \theta_1 B - \theta_2 B^2 - \cdots - \theta_q B^q) r_t = \Theta(B) a_t \quad (3.5)$$

onde $\Theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \cdots - \theta_q B^q$ representa o operador de médias móveis.

A união de ambos os modelos, AR e MA, geram o modelo auto-regressivo de médias móveis (*autoregressive moving average*, ARMA) de ordem (p, q) , onde p representa o número de termos auto-regressivos e q representa o número de termos da média móvel. O Modelo ARMA foi proposto na tentativa de se gerar um modelo de previsão bastante compacto. Desta forma, o modelo ARMA é definido por [49, 60]

$$\tilde{x}_t = \phi_1 \tilde{x}_{t-1} + \cdots + \phi_p \tilde{x}_{t-p} + r_t - \theta_1 r_{t-1} - \cdots - \theta_q r_{t-q}. \quad (3.6)$$

3.2.1 Modelos de Box e Jenkins

Box e Jenkins [10] propuseram uma família de modelos algébricos, baseado no modelo ARMA, referidos como modelos auto-regressivos integrados de médias móveis (*autoregressive integrated moving average*, ARIMA). O modelo ARIMA consiste basicamente na aplicação dos dados a um filtro passa-alta (*high-pass filter*) [10], o qual é sensível apenas às altas frequências da função, sendo estes aplicados a um dado modelo ARMA. Este procedimento, também chamado de diferenças entre dados, é realizado no intuito de se remover as tendências dos dados, tornando a série um processo estacionário, ou seja, um modelo ARIMA(p, q, d) é um estudo algébrico que mostra como uma variável de uma série temporal (\mathbf{X}) é relacionada com seus próprios valores passados ($x_{t-1}, x_{t-2}, \dots, x_{t-p}$) e os valores passados dos termos de ruído ($r_{t-1}, r_{t-2}, \dots, r_{t-p}$), diferenciados d vezes, uma vez que algumas séries temporais quando não estacionárias se tornam estacionárias quando aplicadas a um processo de diferenciação [49, 60]. Por exemplo, na equação a seguir, x_t é tida como uma série temporal não estacionária e w_t uma série estacionária.

$$w_t = x_t - x_{t-1} = (1 - B)x_t = \Delta x_t. \quad (3.7)$$

Os modelos com este tipo de filtro de diferenças são apropriados para séries temporais com comportamento não explosivo, ou seja, séries com alguma forma de homogeneidade em seu comportamento não estacionário [10], uma vez que, tomando-se uma quantidade finita de diferenças d , tornam-se estacionárias, e são chamadas séries não estacionárias homogêneas [49, 60].

Considerando que

$$w_t = \Delta x_t \quad (3.8)$$

é estacionária, w_t pode ser representado por um modelo ARMA(p, q), dado por [49, 60]

$$\Phi(B)w_t = \Theta(B)r_t. \quad (3.9)$$

Como w_t é uma diferença de x_t , esta é uma integral de w_t , representado pela letra “I” (*integrated*) na notação ARIMA, sendo formalmente definido por [49, 60]

$$\Phi(B)\Delta^d x_t = \Theta(B)r_t. \quad (3.10)$$

No modelo ARIMA(p, q, d), os termos p e q representam as ordens de $\Phi(B)$ e $\Theta(B)$, respectivamente, e o termo d representa a ordem das diferenças. Em situações práticas, os modelos ARIMA são construídos utilizando $d = 1$ ou $d = 2$, que correspondem a casos comuns de não estacionariedade homogênea [49, 60]:

- **Séries temporais não estacionárias em relação ao nível:** oscilam em torno de um nível médio durante algum tempo e saltam para outro nível. Para torná-las estacionárias basta utilizar a primeira diferença;
- **Séries temporais não estacionárias em relação a inclinação:** oscilam em uma direção por algum tempo e saltam para outra direção. Para torná-las estacionárias basta utilizar a segunda diferença.

Desta forma, Box e Jenkins [10] propuseram um procedimento capaz de encontrar um modelo adequado para solucionar uma determinada situação. Este procedimento é dividido em três passos: *i*) no Passo 1 é utilizado algum mecanismo para se medir a correlação entre as observações dentro de um conjunto de dados da série temporal. Este mecanismo é representado pela função de auto-correlação (*autocorrelation function*, ACF) e a função de auto-correlação parcial (*partial autocorrelation function*, PACF), ambas formalmente definidas em [10]; *ii*) no Passo 2 estima-se os coeficientes do modelo escolhido no passo 1; e *iii*) no Passo 3, Box e Jenkins [10] propuseram alguns procedimentos de validação a fim de determinar a adequação do modelo candidato como solução para o problema em questão. Ao final, o modelo de previsão estará pronto para ser utilizado na prática quando não falhar em nenhum destes processos.

3.3 Modelos Estatísticos Não-Lineares

Como mencionado na seção anterior, o modelo ARIMA [10] é uma das escolhas mais comuns dentre as técnicas utilizadas em situações reais de previsão de séries temporais. No entanto, os modelos ARIMA são lineares e a maioria das aplicações do mundo real envolvem problemas não-lineares. Isto introduz uma limitação na precisão das previsões geradas, pois tais modelos assumem que as séries são estacionárias (ou que podem ser transformadas em séries estacionárias a partir de transformações simples), ou que tais séries também são processos lineares, de forma que podem ser representadas por modelos lineares. No entanto, não existem motivos pelo qual deve-se generalizar a suposição de linearidade de uma determinada série temporal, haja vista que devido à natureza não-linear que forma o cenário de um fenômeno temporal real, espera-se a existência de relacionamentos não-lineares nos dados históricos da série [25–30, 33, 36, 38, 39, 41, 49, 60, 61].

Uma série temporal pode ser modelada não-linearmente, por [49, 60]

$$x_t = h(x_{t-1}, \dots, x_{t-p}, r_{t-1}, \dots, r_{t-q}) + r_t, \quad (3.11)$$

onde r_t representa uma componente aleatória, ou termo de ruído. Os termos p e q representam índices que determinam as janelas de dependência temporal nos termos passados da série e ruído, respectivamente. O termo $h(\cdot)$ representa uma função de transferência não-linear, que realiza o mapeamento entre os valores do passado e do futuro da série.

A fim de superar as limitações dos modelos estatísticos lineares, vários modelos estatísticos não-lineares tem sido propostos na literatura, como os modelos heterocedásticos condicionais auto-regressivos (*autoregressive conditional heteroskedasticity*) [57], os modelos bilineares (*bilinear models*) [116], os modelos de limiar auto-regressivo (*threshold autoregressive models*) [99], e os modelos de estados gerais dependentes (*general state dependent models*) [128], os modelos auto-regressivos exponenciais (*exponential autoregressive models*) [111], dentre outros.

3.3.1 Modelos Heterocedásticos Condicionais Auto-Regressivos

Esta classe de modelos, referida como heterocedásticos condicionais auto-regressivos (*autoregressive conditional heteroskedasticity*, ARCH), representa uma família paramétrica específica

de heteroscedasticidade em um modelo AR [57]. O modelo ARCH de ordem q é definido como um modelo AR(q) onde r_t é condicionado à informação avaliada no tempo $t - 1$, com variância [57]

$$var_t = \alpha_0 + \sum_{i=1}^q \alpha_i r_{t-i}^2, \quad (3.12)$$

com $\alpha_0 > 0$ e $\alpha_i \geq 0$.

Desta forma o modelo ARCH é formalmente definido por [57]

$$x_t = r_t \sqrt{\alpha_0 + \sum_{i=1}^q \alpha_i r_{t-i}^2}. \quad (3.13)$$

Note que a não-linearidade do modelo ARCH se dá pela variância condicional de r_t [57]. Vários modelos foram propostos baseados no modelo ARCH. No entanto, a mais importante extensão do modelo ARCH é conhecida como modelo heterocedástico condicional auto-regressivo generalizado (*generalized autoregressive conditional heteroskedasticity*, GARCH) [9], onde a variância condicional var_t também é função do tipo $var_t, var_{t-1}, \dots, var_{t-p}$, gerando desta forma um modelo GARCH(p, q).

3.3.2 Modelos Bilineares

Uma classe particular de modelos não-lineares, conhecida como bilinear [116] pode ser vista como uma extensão não-linear natural de um modelo ARMA, fazendo $\mu(x)$ e $\phi_i(x)$ constantes e $\theta_j(y_{t-1}) = \theta_j + \sum_{v=1}^Q c_{jv} x_{t-v}$ (c_{jv} são os coeficientes a serem determinados). A forma geral dos modelos bilineares de ordem (p, q, P, Q) é definida por [116]

$$x_t + \sum_{i=1}^p \phi_i x_{t-i} = \mu + r_t + \sum_{j=1}^q \theta_j r_{t-j} + \sum_{u=1}^P \sum_{v=1}^Q c_{uv} x_{t-v} r_{t-u}. \quad (3.14)$$

Assim, é possível verificar que a Equação (3.14) é linear nos termos x_t e r_t , e não-linear em relação ao termo cruzado de x e r . Desta forma, um modelo bilinear de primeira ordem pode ser montado a partir da Equação (3.14) e dado por [116]

$$x_t = \alpha x_{t-1} + \beta r_t + \gamma x_{t-1} r_{t-1}, \quad (3.15)$$

onde α , β e γ são os parâmetros a serem determinados.

3.3.3 Modelos de Limiar Auto-Regressivo

Outra classe de modelos não-lineares que são utilizados para a previsão de séries temporais são os de limiar auto-regressivo (*threshold autoregressive*, TAR) [99], onde seus parâmetros dependem apenas dos valores passados do seu próprio processo, bem como representam um conjunto finito de possíveis modelos AR que um determinado processo pode vir a obedecer em qualquer ponto no tempo. Um modelo TAR de primeira ordem pode ser definido por [99]

$$x_t = \begin{cases} \alpha_1 x_{t-1} + r_t, & \text{se } x_{t-1} < c \\ \alpha_2 x_{t-1} + r_t, & \text{se } x_{t-1} \geq c \end{cases}, \quad (3.16)$$

onde a constante c é definida como limiar.

3.3.4 Modelos de Estados Gerais Dependentes

O modelo de estados gerais dependentes (*general state dependent models*) [128] de ordem (p, q) é definido como uma expansão de uma linearização local da Equação (3.11) em uma série de Taylor, em torno de um ponto fixo no tempo, que pode ser definida por [128]

$$x_t + \sum_{i=1}^p \phi_i(y_{t-1})x_{t-1} = \mu(y_{t-1}) + r_t + \sum_{j=1}^q \theta_j(y_{t-1})r_{t-1}, \quad (3.17)$$

onde $y_t = (r_{t-q+1}, \dots, r_t, x_{t-q+1}, \dots, x_t)$ é definido como vetor de estado.

3.3.5 Modelos Auto-Regressivos Exponenciais

Outra classe particular de tais modelos não-lineares, conhecida como auto-regressivos exponenciais (*exponential autoregressive*) [111], de ordem p , é determinada utilizando $\mu(x)$ um termo constante, $\theta_j(x) = 0(\forall x)$ e $\phi_i(y_{t-1}) = \phi_i + \pi_i \exp(-\gamma x_{t-1}^2)$ na Equação (3.17), sendo formalmente definido por [111]

$$x_t + \sum_{i=1}^p \{\phi_i + \pi_i \exp(-\gamma x_{t-1}^2)\} x_{t-1} = \mu + r_t, \quad (3.18)$$

onde γ representa um fator de escala para a série temporal.

3.3.6 Modelos de Limiar Auto-Regressivo Auto-Excitado

Caso a passagem relativa de tais modelos for determinada pela localização dos valores dos dados em relação aos limiares, o modelo TAR é então conhecido como modelo de limiar auto-regressivo autoexcitado (*selfexcited threshold autoregressive, SETAR*) [14], que pode ser definido por

$$\mu(x) = \phi_0(j), \quad (3.19)$$

$$\theta_j(x) = 0(\forall x), \quad (3.20)$$

$$\phi_i(y_{t-1}) = \phi_i(j), \quad (3.21)$$

se $x_{t-d} \in R^{(j)}$ ($i = 1, 2, \dots, p; j = 1, 2, \dots, l$), sendo d um inteiro positivo e $R^{(j)}$ um subconjunto dos números reais (os limiares). A Equação (3.13) pode, assim, ser reescrita nestes termos, definindo um modelo SETAR por [14]

$$x_t + \phi_0^{(j)} + \sum_{i=1}^p \phi_i^{(j)} x_{t-i} = r_t^{(j)}, \text{ se } x_{t-d} \in R^{(j)}, j = 1, 2, \dots, l, \quad (3.22)$$

onde tal equação representa um modelo SETAR(l, p, \dots, p). O termo $r_t^{(j)}$ representa um ruído branco, sendo $r_t^{(j)}$ independente de $r_t^{(j')}$, com $j \neq j'$.

3.3.7 Considerações

Vale mencionar que, de acordo com Clements *et al.* [14], mesmo com uma grande quantidade de modelos não-lineares propostos na literatura, não se encontram evidências claras, em termos de desempenho de previsão, a favor dos modelos não-lineares quando comparados aos modelos lineares. Também, Ferreira [60] e Mattos [49] argumentaram que o desempenho de previsão de tais modelos não-lineares está bem abaixo do esperado, e este problema ainda permanece em aberto até os dias atuais.

3.4 Redes Neurais Artificiais

De acordo com [49,60], temos duas classes bem definidas de modelos estatísticos para previsão de séries temporais:

1. lineares: incapazes de solucionar eficientemente o problema devido a não-linearidade das séries temporais financeiras;
2. não-lineares: possuem desempenho equivalente aos modelos lineares e têm limitações práticas devido ao seu alto grau de complexidade computacional e matemática.

Neste contexto, uma alternativa promissora que vem sendo utilizada na literatura para superar as deficiências dos modelos estatísticos lineares e não-lineares são as redes neurais artificiais (*artificial neural networks*, ANN) [74, 75], que são modelos não-lineares com baixo grau de complexidade computacional e matemática quando comparadas aos modelos estatísticos. As ANNs são modelos que simulam o comportamento dos sistemas neurais biológicos [103]. Desta forma, ANNs podem então ser vistas como um conjunto de unidades de processamento paralelas e distribuídas, referidas como neurônios, que são capazes de realizar mapeamentos não-lineares [74, 75].

Geralmente, estes neurônios compõem uma estrutura em camadas que são interligadas através de conexões. Na maioria dos modelos, estas conexões são referenciadas como pesos sinápticos, sendo estes responsáveis pelo armazenamento do conhecimento adquirido do fenômeno em análise e atuam no sistema como ponderadores de sinais [49, 60]. Cada unidade de processamento da ANN recebe um sinal, que é ponderado pelos pesos sinápticos, processado por uma função matemática, referida como função de ativação ou função de transferência, produzindo assim um novo sinal de saída que é propagado pela rede. Portanto, os neurônios representam a unidade fundamental de processamento da informação [74, 75]. Tipicamente, a saída de um dado neurônio de uma ANN é dada por

$$y = f(u), \quad (3.23)$$

com

$$u = \sum_{j=1}^J w_j x_j + b, \quad (3.24)$$

onde J é a dimensionalidade do sinal de entrada, x_j com $j = 1, 2, \dots, J$ é o sinal de entrada, w_j são os pesos sinápticos, u é o nível de ativação interna ou potencial de ativação, $f(\cdot)$ é a função

de ativação, b é referido como *bias*, que tem o efeito de transladar a função de ativação em torno da origem, e y representa a ativação de saída do neurônio.

O neurônio também pode ser definido utilizando notação vetorial. Seja $\mathbf{x} = (x_1, x_2, \dots, x_J)$ o vetor que representa o sinal de entrada, $\mathbf{w} = (w_1, w_2, \dots, w_J)$ o vetor que representa os pesos sinápticos do neurônio e b um escalar. Então a saída do neurônio é definida por [74, 75]

$$y(\mathbf{w}, \mathbf{x}, b) = f(\mathbf{w}^T \mathbf{x} + b), \quad (3.25)$$

em que \cdot^T representa uma operação de transposição.

A função de ativação $f(\cdot)$ é responsável pela definição da ativação de saída do neurônio em termos do seu nível de ativação interna. Existem diversas funções de ativação propostas na literatura, mas existem três que são comumente utilizadas [74, 75]:

- Linear: a ativação de saída do neurônio é dada pelo seu próprio nível de ativação interna, sendo definida por

$$f(u) = u; \quad (3.26)$$

- Sigmóide: é a função mais utilizada em ANNs, sendo definida como uma função monotônica crescente que apresenta propriedades assintóticas e de suavidade, contida no intervalo $[0, 1]$. A ativação de saída do neurônio é dada por

$$f(u) = \frac{1}{1 + e^{-au}}, \quad (3.27)$$

em que a representa o parâmetro de inclinação da suavização da função;

- Tangente hiperbólica: é uma função sigmóide contida no intervalo $[-1, 1]$. A ativação de saída do neurônio é dada por

$$f(u) = \frac{1 - e^{-au}}{1 + e^{-au}}. \quad (3.28)$$

Portanto, em analogia a cérebro humano, as ANNs são compostas por neurônios artificiais e são dotadas da capacidade de aprendizado através de exemplos, bem como de realizar interpolações e extrapolações de tais informações aprendidas [49, 60]. Entretanto, para que a ANN produza resultados satisfatórios, esta deve ser condicionada a um processo de aprendizagem, isto é, a determinação dos valores dos pesos sinápticos de cada neurônio da rede. Note que o processo de aprendizagem realiza o ajuste fino dos pesos das conexões na tentativa de melhor generalizar o fenômeno em questão.

Dentre as diversas ANNs propostas na literatura, de acordo com [49, 60], a mais difundida para previsão de séries temporais e abordada neste trabalho é a *perceptron* multicamadas (*multilayer perceptron*, MLP) [74, 75].

3.4.1 Perceptron Multicamadas

As redes neurais *perceptron* multicamadas (*multilayer perceptron*, MLP) [74, 75], são redes dotadas de arquiteturas neurais onde os neurônios são organizados em duas ou mais camadas

de processamento, sendo a ANN mais frequentemente encontrada na literatura de previsão de séries temporais [49, 60]. Redes MLP são compostas por uma camada de entrada (*input layer*), cuja função é receber a informação e distribuí-la (sem modificá-la) a todos os neurônios da camada seguinte, e uma camada de saída (*output layer*), que tem a função de apresentar a resposta final para a informação apresentada à camada de entrada da rede. Entre as camadas de entrada e saída, existe uma ou mais camadas interligadas, conhecidas como camadas escondidas (*hidden layers*). No entanto, redes MLP com apenas uma camada escondida são aproximadores universais [74, 75]. Note que os neurônios de uma determinada camada são conectados a outros neurônios da camada posterior. Se cada neurônio de uma determinada camada estiver conectado a todos os neurônios da camada posterior, tem-se uma rede MLP totalmente conectada.

No caso particular do problema de previsão de séries temporais, as entradas da rede MLP são as variáveis de predição. Desta forma, a relação funcional pode ser escrita como [49, 60]:

$$y = g(\mathbf{x}), \quad (3.29)$$

onde $\mathbf{x} = (x_1, x_2, \dots, x_J)$ é um vetor com J variáveis independentes e y é uma variável dependente. Portanto a rede MLP funciona de forma equivalente a um modelo de regressão não-linear.

Por outro lado, encarando o problema como uma extrapolação, as entradas da rede são tipicamente as observações passadas da série temporal e a saída é o seu respectivo valor futuro. Então, a rede MLP dependerá de uma função de mapeamento dada por [49, 60]

$$y_t = g(\mathbf{y}), \quad (3.30)$$

onde $\mathbf{y} = (y_{t-1}, y_{t-2}, \dots, y_{t-J})$ é um vetor com J observações da série temporal, sendo y_t a observação no tempo t . Portanto, a rede MLP funciona de forma equivalente a um modelo auto-regressivo não-linear. Vale mencionar que uma rede MLP pode incorporar tanto variáveis de previsão, quanto retardos temporais de uma série temporal, equivalendo-se a um modelo geral de função de transferência [49, 60, 61].

A mais importante propriedade de uma rede MLP é sua capacidade de aprendizagem. Uma rede MLP aprende através de um processo iterativo de ajustes aplicados aos seus pesos sinápticos e *bias*, o qual pode ser expresso na forma de um algoritmo computacional. Segundo [74, 75], o termo aprendizagem é definido como um processo pelo qual os parâmetros livres de uma rede neural são adaptados através de um processo de estímulo pelo ambiente no qual a rede está inserida. O processo de aprendizagem de uma rede MLP é do tipo supervisionado. Este tipo de aprendizado é caracterizado pela presença de um agente externo que induz a rede MLP a uma resposta desejada a um determinado estímulo apresentado pelo ambiente, de forma a conseguir realizar o mapeamento entre a entrada e saída desejada, através da minimização de um sinal de erro, definido como o erro médio quadrático entre resposta desejada e a resposta observada na saída da rede MLP.

O objetivo do processo de aprendizagem por correção de erro é minimizar uma função de custo baseada no sinal de erro, de modo que a resposta observada se aproxime da resposta desejada a cada iteração, definida aqui como época, no processo de aprendizagem. Uma vez definida a função de custo, o problema de aprendizagem se torna um problema de otimização não-linear. Portanto, a rede MLP aprende através da minimização da função de custo em relação aos pesos sinápticos e *bias* da rede.

Note que uma função de custo f define uma superfície de erro sobre o espaço de pesos. Se P representa a dimensionalidade dos vetor de pesos ajustáveis na rede neural e N representa a dimensionalidade do padrão de saída do problema, então $f : \mathbb{R}^P \rightarrow \mathbb{R}^N$. Nesta superfície, tipicamente tem-se a presença de mínimos locais e globais. Os métodos de otimização tipicamente utilizados para minimizar a função f utilizam informações do gradiente descendente do erro para ajustar os parâmetros da rede. Teoricamente tais métodos sempre encontram pontos de mínimo na superfície de erro, no entanto nada se pode observar sobre a natureza do mínimo (local ou global) encontrado a partir de uma condição inicial arbitrária.

O método clássico que é utilizado no processo de aprendizagem de redes neurais MLP, que utiliza informações do gradiente descendente do erro, é o algoritmo de retro-propagação do erro (*back-propagation*, BP) [74, 75]. O algoritmo BP consiste em duas fases [74, 75]:

1. Um dado padrão de entrada é fornecido e seu efeito é propagado pela rede, de camada a camada. Nesta etapa, os pesos sinápticos da rede permanecem fixos. A ideia é que o sinal entrada é ponderado e somado em cada neurônio da primeira camada escondida. O resultado desta soma é transformado pela função de ativação do neurônio, gerando seu sinal de saída, que é propagado para a próxima camada da rede. Este processo se repete até que a ativação dos neurônios da camada de saída sejam calculados.
2. O sinal de erro é calculado na camada de saída da rede e este erro é retro-propagado, de camada a camada, e ao final deste processo os pesos sinápticos são ajustados de acordo com uma regra de correção de erro.

Como mencionado por [74, 75], o algoritmo de treinamento mais popular para redes MLP é o BP. No entanto, outros algoritmos de aprendizagem têm sido utilizados com sucesso para treinamento de redes MLP. Tais algoritmos são modificações ou variações do BP tradicional. Como exemplos temos os métodos adaptativos [79, 100], *quick-propagation* (QUICKPROP) [58, 59], o *resilient back-propagation* (RPROP) [117], e outros métodos de segunda ordem, como levenberg-marquardt (LM) [71], gradiente conjugado escalar (*scaled conjugate gradient*, SCG) [97], gradiente conjugado de um passo secante (*one step secant conjugate gradient*, OSSCG) [5], dentre outros. A escolha do algoritmo de treinamento também é uma tarefa de difícil solução, dependendo fortemente do problema a ser abordado.

Os dados utilizados no processo de aprendizagem da rede MLP são referidos como padrões de treinamento (no caso particular de séries temporais esses padrões assumem os valores dos retardos temporais da série ou valores de variáveis independentes), correspondendo a cada um dos elementos destes padrões a um neurônio na camada de entrada da rede. No caso particular da aplicação de redes MLP para previsão de séries temporais, os dados são usualmente dividido em dois ou três conjuntos [49, 60]:

- Na primeira configuração, quando os dados são divididos em dois conjuntos, o primeiro conjunto, chamado de treinamento, é utilizado no processo de aprendizagem do modelo de previsão, e o segundo conjunto, chamado de teste, é utilizado para realizar os testes de desempenho do modelo. *A priori*, não existe nenhuma regra bem estabelecida para determinar a cardinalidade de cada um destes conjuntos, entretanto os trabalhos apresentados

na literatura [49,60] sugerem um percentual de 70% a 90% dos dados para o conjunto de treinamento, e o restante para o conjunto de teste.

- Na segunda configuração, quando os dados são divididos em três conjuntos, o primeiro conjunto é utilizado para o processo de aprendizagem, o segundo conjunto é utilizado para validar o processo de aprendizagem (determinando o momento mais adequado para o seu término), e o terceiro conjunto constitui o teste do modelo. Também não existe nenhuma regra definitiva para determinar a cardinalidade desta configuração. Entretanto, existe uma tentativa de padronização de experimentos com redes neurais que sugere a divisão destes dados nos seguintes percentuais [110]: 50% dos dados para o conjunto de treinamento, 25% dos dados para o conjunto de validação e 25% dos dados para o conjunto de teste. Esta configuração será a utilizada neste trabalho

Note que, independente da configuração dos percentuais de divisão dos dados, o conjunto de teste é bastante inferior ao conjunto de treinamento, uma vez que como redes MLP são semi-paramétricas, estas precisam de uma grande quantidade de dados para construção precisa do mapeamento entrada / saída dos dados [74,75].

3.5 Algoritmos Evolucionários

Os algoritmos evolucionários (*evolutionary algorithms*, EAs) [56] são uma poderosa classe de algoritmos de otimização estocásticos que tem sido largamente utilizados para solucionar problemas complexos que não possuem solução analítica, como por exemplo problemas de otimização com características de descontinuidade e multimodalidade. Tais algoritmos foram motivados pela teoria da evolução [20], onde o principal objetivo é encontrar meios pelos quais os mecanismos do processo de evolução natural possam ser integrados em sistemas computacionais.

Os EAs têm sido utilizados com sucesso em diversos tipos de aplicações práticas devido ao seu alto poder de busca em espaços de estados, sendo largamente aplicados a problemas de aprendizagem de máquina e otimização numérica [56]. Este tipo de técnica trabalha com um conjunto de soluções candidatas ao problema (estados iniciais). Este conjunto, referido como população (composta por indivíduos), é evoluído utilizando operadores evolucionários de seleção, cruzamento e mutação, de forma a convergir, a partir dos estados iniciais, para pontos sub-ótimos ou ótimos no espaço de estados do problema, realizando uma busca simultânea por múltiplas trajetórias [56].

A metáfora dos EAs é bastante simples: o meio ambiente é definido por um grupo, ou população, de indivíduos, que disputam entre si pela sobrevivência no meio ambiente. A aptidão mostra o quão adaptado é o indivíduo ao meio ambiente, de forma a representar a capacidade de se reproduzir e gerar descendentes (cruzamento). Como em qualquer meio, possíveis alterações ou aberrações podem surgir nos indivíduos (mutação). Assim, é possível notar que a escolha da função de aptidão é um passo crucial para o sucesso para resolução de um problema de otimização utilizando EAs.

Na literatura, é possível encontrar uma quantidade expressiva de EAs [1,2,12,54,72,85–87,144,145,158,159,162,164]. O principal desafio enfrentado por estes EAs é encontrar soluções

ótimas ou sub-ótimas de um dado problema utilizando uma quantidade pequena de indivíduos na população associado a um alto desempenho em termos de velocidade de convergência do algoritmo. Nestes trabalhos são propostos diversos operadores de cruzamento e mutação, bem como a utilização de outras técnicas para evoluir as soluções iniciais na população.

Embora se tenha uma grande quantidade de algoritmos de otimização disponíveis na literatura, os mais difundidos e utilizados neste trabalho são: i) algoritmos genéticos (*genetic algorithms*, GAs) [64, 66, 77, 78, 85, 96, 108], e ii) otimizador de enxame de partículas (*particle swarm optimizer*, PSO) [54, 55, 134, 135, 161, 162, 164].

3.5.1 Algoritmos Genéticos

O conceito de algoritmos genéticos (*genetic algorithms*, GAs) foi desenvolvido por Holland [77, 78] e posteriormente desenvolvido por Goldberg [66], aplicando GAs em aplicações práticas industriais. Os GAs representam uma poderosa técnica de aprendizagem de máquina, busca e otimização inspirada na teoria da evolução [20].

O desenvolvimento dos GAs partiu da ideia que as espécies se evoluem pelo princípio de seleção natural e sobrevivência dos mais aptos. A seleção privilegia os indivíduos mais aptos com maior longevidade, e por isso estes possuem maior probabilidade de sobrevivência e reprodução, repassando seu código genético para as próximas gerações. Estes códigos genéticos constituem as características hereditárias que estão representadas nos cromossomos. Portanto, indivíduos que possuírem características hereditárias que são mais aptas ao meio ambiente terão grandes chances de sobreviver, enquanto os que não forem dotados de tais características serão extintos ao longo das gerações.

O procedimento do GA é bastante simples. Este realiza uma busca pela melhor solução do problema, partindo de um conjunto inicial de soluções do espaço de estados. Cada solução é referida como indivíduo. Um grupo de indivíduos é chamado de população. Os operadores de seleção, cruzamento e mutação são iterativamente aplicados para permitir a evolução da população para pontos de ótimo no espaço de estados. Portanto, como um típico EA, o GA realiza busca através de múltiplas trajetórias simultaneamente, podendo vir a ser estendidos para execução em computadores paralelos [62].

De forma geral, os GAs necessitam da definição do espaço de busca do problema e da função de aptidão (*fitness function*). Note que o GA realiza uma busca em um espaço de estados, sendo o melhor estado aquela que tiver a maior função de aptidão. Existe diversas modificações propostas para GAs, mas todos seguem duas etapas básicas: i) Na primeira geração um conjunto de soluções é gerado e avaliado através de uma função de aptidão, ii) Nas gerações restantes, novas populações são construídas através de operadores evolucionários, e posteriormente avaliados. A segunda etapa é iterativa até que a convergência seja alcançada. O pseudo-código descrito na Figura 3.1, descreve um algoritmo genético padrão (*standard genetic algorithm*, SGA) [66, 77, 78, 96].

O SGA é iniciado através da criação de uma população de indivíduos (soluções candidatas ao problema). Cada indivíduo é formado por um cromossomo, que contém toda a informação para representar o indivíduo [66, 77, 78, 96]. O cromossomo pode ser implementado por várias estruturas de dados. No entanto, na maioria dos casos este é implementado por um vetor, onde cada elemento do vetor é definido como um gene que codifica uma determinada característica

```

AlgoritmoGenéticoPadrão() {
    geração = 0;
    inicializa população;
    avalia  $f(população)$ ; //  $f(\cdot)$ : função de aptidão
    enquanto ( não condição de parada ) {
        geração = geração + 1;
        seleciona pares de indivíduos da população;
        executa o operador de cruzamento com os pares de indivíduos selecionados;
        gera novapopulação;
        executa o operador de mutação com novapopulação;
        avalia  $f(novapopulação)$ ;
        população = novapopulação;
    }
}

```

Figura 3.1 Pseudo-código do algoritmo genético padrão.

do indivíduo. Assim, dado um problema de otimização, idealiza-se um indivíduo como sendo formado por um cromossomo composto de vários genes (parâmetros de uma possível solução do problema) [66, 77, 78, 96].

A avaliação da população no SGA é feita para todos os indivíduos, através da função de aptidão. O indivíduo com maior aptidão será aquele que obtiver a melhor avaliação. A função de aptidão pode ser uma função matemática ou um processo que gera um índice de ordenamento para todos os indivíduos da população. Esta função é totalmente dependente do problema [66, 77, 78, 96].

Para criar a nova população, é necessário selecionar pares de indivíduos. Existem diversos meios de realizar esta seleção, como seleção baseada em meta [85, 96], método de amostragem estocástica (método da roleta) [85, 96], estratégias elitistas [85, 96], eleição de estados ligados [85, 96] e seleção por torneio [85, 96]. Dentre estes, a seleção por torneio e o método da roleta merecem um destaque especial por serem os mais comumente utilizados na literatura.

O método da seleção por torneio [85, 96] escolhe aleatoriamente uma certa quantidade de indivíduos da população, e a partir destes escolhe-se o mais apto (maior valor da função de aptidão). Tal procedimento é repetido até que sejam selecionados a quantidade necessária de pares de indivíduos para execução dos operadores genéticos. O método da roleta utiliza um procedimento de seleção por amostragem estocástica, onde o indivíduo mais apto terá uma chance de ser selecionado diretamente proporcional ao valor da sua função de aptidão e inversamente proporcional ao somatório do valor da função de aptidão de todos os indivíduos da população. Tal procedimento é representado graficamente por um diagrama circular, onde os valores de cada categoria estatística representam a probabilidade de um determinado indivíduo da população ser selecionado. Quanto maior a área da categoria, maior será a chance do indivíduo associado a categoria ser selecionado.

Após o processo de seleção, são utilizados os operadores genéticos, que são responsáveis pela modificação dos valores dos genes de cada indivíduo, e correspondem as versões abstra-

tas dos operadores encontrados na evolução biológica, que são os operadores de cruzamento (*crossover*) e mutação.

O operador de cruzamento usualmente combina os genes de um par de indivíduos para extrair informação genética de ambos, obtendo os indivíduos descendentes. Existem diversos procedimentos utilizados para executar o operador de cruzamento, como cruzamento de um ponto, dois pontos ou multiponto [85, 96], cruzamento aritmético [85], cruzamento heurístico [85], dentre outros. Os indivíduos gerados pelo operador de cruzamento são definidos como a nova população.

Um dos fatores que gera diversificação de uma população de indivíduos em um sistema biológico é o aparecimento de indivíduos mutacionados, que tem uma probabilidade baixa gerada por fatores causais. Analogamente, o operador de mutação é empregado na nova população, que contém características relevantes de todos os pares de indivíduos obtidos no processo de seleção. O operador de mutação é responsável pela modificação aleatória dos genes dos indivíduos, induzindo a diversidade na população e permitindo ao SGA escapar de ótimos locais da superfície da função de aptidão.

Após a aplicação do operador de cruzamento e mutação, a nova população é avaliada pela função de aptidão e esta é definida como a população atual para a nova geração do SGA. Todo este procedimento é repetido até que uma condição de parada seja alcançada. De forma geral, apenas estes operadores são necessários para se encontrar boas soluções em uma grande quantidade de aplicações práticas de GAs.

Vale mencionar que os GAs não podem ser diretamente utilizados para solucionar o problema de previsão de séries temporais, no entanto servem como ferramenta auxiliar a modelos de previsão na tentativa de se encontrar uma modelagem mais precisa de previsão, gerando desta forma sistemas híbridos [25–30, 33, 36, 38, 39, 41, 49, 60, 61].

A seção a seguir apresenta uma versão modificada do SGA, definida como algoritmo genético modificado (*modified genetic algorithm*, MGA) [85], que emprega operadores genéticos modificados com o intuito de aceleração de sua convergência.

3.5.1.1 Algoritmo Genético Modificado

O algoritmo genético modificado (*modified genetic algorithm*, MGA), proposto por Leung [85], utiliza operadores de cruzamento e mutação modificados, de forma a cobrir mais eficientemente a superfície da função de aptidão do problema, e conseqüentemente acelerando a velocidade de convergência para pontos de ótimo no espaço de estados.

O pseudo-código do MGA é apresentado em detalhes na Figura 3.2.

De acordo com a Figura 3.2, o procedimento do MGA consiste na seleção de um par de indivíduos e então executar os operadores de cruzamento e mutação modificados (gerando a descendência) até que um critério de parada seja alcançado. Ao final do procedimento, o indivíduo com maior aptidão é selecionado como solução do problema em questão.

O operador de cruzamento modificado é utilizado para extrair informação de dois indivíduos da população (vetores \mathbf{P}^1 e \mathbf{P}^2) obtidos no processo de seleção utilizando o método da roleta [85]. O operador de cruzamento gera quatro novos indivíduos descendentes (vetores \mathbf{C}^1 ,

```

AlgoritmoGenéticoModificado() {
  geração = 0;
  inicializa população;
  avalia f(população); // f(·): função de aptidão
  enquanto ( não condição de parada ) {
    geração = geração + 1;
    seleciona um par de indivíduos da população (vetores P1 e P2);
    executa o operador de cruzamento com o par de indivíduos selecionados
    gerando quatro novos descendentes (vetores C1, C2, C3 e C4);
    o descendente mais apto dentre os quatro gerados é definido como Cbest;
    avalia f(Cbest);
    insere Cbest na população na posição do indivíduo menos apto;
    executa o operador de mutação com Cbest gerando três novos descendentes
    mutacionados (vetores CM1, CM2 e CM3);
    se (número aleatório < pmut) {
      o indivíduo mutacionado com maior aptidão é inserido na população
      na posição do indivíduo menos apto;
    } senão {
      se ( f(CM1) > aptidão do pior indivíduo da população ) {
        insere o indivíduo CM1 na posição do indivíduo menos apto;
      }
      se ( f(CM2) > aptidão do pior indivíduo da população ) {
        insere o indivíduo CM2 na posição do indivíduo menos apto;
      }
      se ( f(CM3) > aptidão do pior indivíduo da população ) {
        insere o indivíduo CM3 na posição do indivíduo menos apto;
      }
    }
    avalia f(população);
  }
}

```

Figura 3.2 Pseudo-código do algoritmo genético modificado.

C², C³ e C⁴), que são formalmente definidos por

$$\mathbf{C}^1 = \frac{\mathbf{P}^1 + \mathbf{P}^2}{2}, \quad (3.31)$$

$$\mathbf{C}^2 = w[\max(\mathbf{P}^1, \mathbf{P}^2)] + (1 - w)\mathbf{P}^{\max}, \quad (3.32)$$

$$\mathbf{C}^3 = w[\min(\mathbf{P}^1, \mathbf{P}^2)] + (1 - w)\mathbf{P}^{\min}, \quad (3.33)$$

$$\mathbf{C}^4 = \frac{w(\mathbf{P}^1 + \mathbf{P}^2) + (1 - w)(\mathbf{P}^{\max} + \mathbf{P}^{\min})}{2}. \quad (3.34)$$

O símbolo $w \in [0, 1]$ representa o peso do cruzamento (o quão próximo w estiver do limite superior do intervalo, maior é a contribuição direta dos indivíduos \mathbf{P}^1 e \mathbf{P}^2). Os termos $\max(\mathbf{P}^1, \mathbf{P}^2)$ e $\min(\mathbf{P}^1, \mathbf{P}^2)$ representam os vetores cujos elementos são, respectivamente, dados pelos valores máximo e mínimo de \mathbf{P}^1 e \mathbf{P}^2 . Os termos \mathbf{P}^{max} e \mathbf{P}^{min} denotam os vetores com os valores máximo e mínimo, respectivamente, que os genes podem assumir. Note que os indivíduos \mathbf{C}^1 e \mathbf{C}^4 tendem a cobrir a região central do domínio do espaço de busca, e os indivíduos \mathbf{C}^2 e \mathbf{C}^3 tendem a cobrir a região em torno dos limites inferiores e superiores do domínio do espaço de busca. O indivíduo dentre os quatro descendentes gerados com maior aptidão é denotado pelo vetor \mathbf{C}^{best} , e é desta forma o descendente gerado pelo operador de cruzamento. O indivíduo \mathbf{C}^{best} é inserido na população na posição do indivíduo com menor aptidão.

Após o término do operador de cruzamento, é aplicado o operador de mutação, onde três novos indivíduos mutacionados são gerados (vetores \mathbf{CM}^1 , \mathbf{CM}^2 e \mathbf{CM}^3) a partir de \mathbf{C}^{best} e formalmente definidos por

$$\mathbf{CM}^j = \mathbf{C}^{best} + \Gamma^j \Delta \mathbf{CM}^j, \quad j = 1, 2, 3. \quad (3.35)$$

Na Equação (3.35), os vetores $\Delta \mathbf{CM}^j$ satisfazem o critério $\mathbf{P}^{min} \leq \mathbf{C}^{best} + \Delta \mathbf{CM}^j \leq \mathbf{P}^{max}$ para $j = 1, 2, 3$. O vetor Γ^j é um vetor binário que satisfaz as seguintes condições: i) O vetor Γ^1 tem apenas um valor zero escolhido aleatoriamente, ii) o vetor Γ^2 é um vetor binário aleatório, e iii) o vetor Γ^3 é um vetor constante $\mathbf{1}$ (consistindo apenas do valor um).

Os indivíduos mutacionados descendentes do operador de mutação são incluídos na população de acordo com o esquema a seguir: Um número aleatório r é gerado no intervalo unitário $[0, 1]$. Se $r < pmut$, onde $pmut$ representa a probabilidade de mutação, então o indivíduo mutacionado de maior aptidão é incluído na população na posição do indivíduo menos apto. Caso contrário, o MGA executa o seguinte passo para $j = 1, 2, 3$: se o valor de aptidão do indivíduo \mathbf{CM}^j é maior do que o valor de aptidão do indivíduo menos apto da população, então \mathbf{CM}^j é incluído na população na posição do indivíduo menos apto.

3.5.2 Otimizador por Enxame de Partículas

O otimizador de enxame de partículas (*particle swarm optimizer*, PSO) [54, 55] é uma técnica de otimização estocástica, que como um típico EA, também realiza o processo de busca através de múltiplas trajetórias simultaneamente, sendo facilmente estendidos para execução em computadores paralelos [62]. O PSO, diferentemente dos GAs, foi idealizado a partir da observação do comportamento auto-organizado exercido entre o indivíduo (referenciado como partícula) e a população ou grupo de indivíduos (enxame) que se adaptam ao meio ambiente.

A idealização do PSO foi desenvolvida a partir de observações de agrupamentos de animais como um bando de pássaros ou um cardume de peixes em busca de alimento. Foi percebido, a partir destas observações, que as ações do grupo são influenciadas pela experiência individual acumulada e pela experiência coletiva [54, 55].

De forma prática, esta análise comportamental é traduzida numa população de partículas (indivíduos), que são soluções candidatas ao problema. Como num bando de pássaros, cada pássaro (partícula) é influenciada pela melhor posição encontrada pelo próprio indivíduo (termo cognitivo), quanto pela melhor posição encontrada pelo enxame como um todo (termo social)

[54, 55]. Desta forma, cada partícula se direciona a partir de sua posição atual com influência dos termos cognitivos, social, e um termo de velocidade. O termo de velocidade ou fator de aceleração é um valor que varia iterativamente tomando como base os termos cognitivo e social. Adicionalmente, o fator de aceleração é ponderado através de uma componente aleatória, na tentativa do algoritmo não ficar preso em pontos de ótimos locais e garantindo a diversidade das partículas [54, 55]. Quando uma partícula descobre novas regiões promissoras no espaço de busca, o enxame é guiado progressivamente devido ao termo social, de forma a explorar mais precisamente a região encontrada [54, 55].

O PSO inicia o seu procedimento a partir da criação do enxame, ou seja, pela inicialização de cada partícula (posições e velocidades aleatórias dentro do domínio do espaço de busca). Todo o enxame é avaliado através de uma função de aptidão. Para cada partícula do enxame, é verificado se a mesma possui uma posição atual mais apta que a melhor posição encontrada individualmente e a melhor posição encontrada pelo enxame, atualizando-as se necessário. Posteriormente a velocidade e a posição atual de todas as partículas do enxame são atualizadas. Todos estes passos são repetidos até que uma condição de parada seja encontrada.

No pseudo-código apresentado na Figura 3.3, o tamanho do enxame é definido por S . A i -ésima partícula do enxame ($1 \leq i \leq S$) tem uma posição atual no espaço de busca, definida por

$$\mathbf{pos}_i(t) = (pos_{i,1}(t), pos_{i,2}(t), \dots, pos_{i,j}(t), \dots, pos_{i,p}(t)), \quad (3.36)$$

uma velocidade atual, definida por

$$\mathbf{vel}_i(t) = (vel_{i,1}(t), vel_{i,2}(t), \dots, vel_{i,j}(t), \dots, vel_{i,p}(t)), \quad (3.37)$$

e a melhor posição individual armazenada, definida por

$$\mathbf{pb}_i(t) = (pb_{i,1}(t), pb_{i,2}(t), \dots, pb_{i,j}(t), \dots, pb_{i,p}(t)). \quad (3.38)$$

Assumindo um problema de maximização n -dimensional, em cada iteração do PSO, a velocidade de cada partícula do enxame é atualizada por

$$\mathbf{vel}_i(t+1) = w\mathbf{vel}_i(t) + c_1 r_1 [\mathbf{pb}_i(t) - \mathbf{pos}_i(t)] + c_2 r_2 [\mathbf{pb}(t) - \mathbf{pos}_i(t)], \quad (3.39)$$

onde $j \in 1, 2, \dots, n$, $\mathbf{pb}(t) = [\hat{pb}_1(t), \hat{pb}_2(t), \dots, \hat{pb}_n(t)]$ denota um vetor n -dimensional representando a melhor posição encontrada no espaço de busca pelo enxame (encontrada por qualquer partícula), c_1 e c_2 representam os coeficientes de aceleração, que controlam como a partícula deve se mover em uma determinada iteração, e $r_1 \sim U(0, 1)$ e $r_2 \sim U(0, 1)$ são elementos de duas sequências aleatórias uniformes no intervalo $[0, 1]$. O termo w é definido como peso de inércia, onde este valor é tipicamente definido para variar linearmente no intervalo $[0, 1]$, tendo seu valor decrementado durante o processo de busca.

Portanto, a nova posição atual da i -ésima partícula é atualizada por

$$\mathbf{pos}_i(t+1) = \mathbf{pos}_i(t) + \mathbf{vel}_i(t+1). \quad (3.40)$$

A melhor posição individual da partícula e a melhor posição encontrada pelo enxame são atualizadas, respectivamente, por

$$\mathbf{pb}_i(t+1) = \begin{cases} \mathbf{pb}_i(t) & \text{se } f(\mathbf{pos}_i(t+1)) \leq f(\mathbf{pb}_i(t)), \\ \mathbf{pos}_i(t+1) & \text{caso contrário.} \end{cases} ; \quad (3.41)$$

e

$$\hat{\mathbf{pb}}(t+1) = \arg \max [f(\mathbf{pb}_i(t+1))], \quad 1 \leq i \leq S, \quad (3.42)$$

onde $\arg \max(\cdot)$ denota o argumento máximo e $f(\cdot)$ é uma função de custo.

Cada componente do vetor \mathbf{vel}_i é normalizado no intervalo $[-vel_{max}, vel_{max}]$, de forma o enxame atue sob o domínio restrito do espaço de busca do problema, fazendo com que a busca se torne uma exploração. Vale mencionar que este mecanismo não restringe os valores de \mathbf{pos}_i no intervalo de \mathbf{vel}_i , pois este apenas limita a distância máxima que cada partícula poderá se mover durante cada iteração do PSO [162].

```

OtimizadorPorEnxameDePartículas() {
  inicializa enxame;
  avalia  $f(\mathit{enxame})$ ; //  $f(\cdot)$ : função de aptidão
  repita
    para  $i = 1$  to  $S$  faça
      se  $f(\mathbf{pos}_i(t)) > f(\mathbf{pb}_i(t))$  {
         $\mathbf{pb}_i(t) = \mathbf{pos}_i(t)$ 
      }
      se  $f(\mathbf{pb}_i(t)) > f(\hat{\mathbf{pb}}(t))$  {
         $\hat{\mathbf{pb}}(t) = \mathbf{pb}_i(t)$ 
      }
    }
  atualiza a velocidade, a posição atual e a melhor posição
  de cada partícula do enxame;
  atualiza a melhor posição do enxame;
  avalia  $f(\mathit{enxame})$ ;
  até ( condição de parada );
}

```

Figura 3.3 Pseudo-código do otimizador de enxame de partículas.

Também, note que o PSO não pode ser diretamente utilizado para solucionar o problema de previsão de séries temporais, uma vez que o mesmo apenas pode ser utilizado como ferramenta auxiliar a modelos de previsão na tentativa de se encontrar uma representação mais precisa da série, gerando desta forma sistemas híbridos para previsão.

3.6 Sistemas Híbridos

Os seres humanos são exemplos de máquinas que possuem informações híbridas [60]. As atitudes e ações dos seres humanos refletem a combinação entre informação genética e informação adquirida através de aprendizagem [67]. A combinação destes diferentes tipos de processamento de informação é indispensável ao processo de sobrevivência dos seres humanos em meios complexos e dinâmicos [67].

O processo de hibridização de sistemas computacionais tem produzido resultados bastante interessantes em alguns tipos de tarefas, em particular em certos problemas complexos que não podem ser solucionados por uma única técnica inteligente [67]. Considerando que uma única técnica tem propriedades particulares que não são suficientes para a resolução completa de um determinado problema, é necessário combinar técnicas com propriedades complementares de forma a resolver completamente o problema, e este tem sido a grande motivação para o estudo de sistemas híbridos (*hybrid systems*, HSs) [67], onde duas ou mais técnicas são combinadas com o intuito de superar as limitações de técnicas utilizadas individualmente [67].

Os HSs se tornam necessários quando consideramos o largo domínio das aplicações, uma vez que estas possuem componentes complexas de problemas distintos, cada qual necessitando de um tipo particular de processamento. Além disso, os HSs não somente representam a combinação de diferentes técnicas, mas a integração delas com sistemas computacionais convencionais [60]. É possível sumarizar três principais motivos para a criação de sistemas híbridos [67]: i) técnicas de intensificação: representa uma integração de diferentes técnicas na tentativa de superação das limitações da utilização de técnicas individualmente, ii) multiplicidade de aplicações em tarefas: representa a integração de diversas técnicas para diversos sub-problemas que uma determinada aplicação venha a possuir, e iii) realização de múltipla funcionalidade: representa a integração de técnicas com capacidade de processamento múltiplo de informações dentro de uma arquitetura.

Segundo Goonatilake and Khebbal [67], os HSs têm um papel importante no desenvolvimento de sistemas inteligentes futuros, sendo atualmente solicitados por uma demanda crescente de aplicações práticas, especialmente nas áreas de percepção de máquina, processamento inteligente de sinais, sistemas de apoio à decisão, previsão de séries temporais, análise de risco de apólices de seguros, processamento de linguagem natural, dentre outras. Existem uma variedade de combinações entre técnicas distintas que podem ser utilizadas para se definir uma solução híbrida para previsão de séries temporais. No entanto, a discussão aqui exposta se limita a combinação de um modelo conexionista (ANN do tipo MLP) com dois modelos evolucionários (MGA e PSO), uma vez que, de acordo com [49, 60], estas integrações apresentam os resultados mais expressivos e comumente encontrados na literatura.

3.6.1 Algoritmos Genéticos e Redes Neurais Multicamadas

A fim de se definir uma solução a um dado problema de previsão de séries temporais, as ANNs do tipo MLP requerem a definição de um conjunto de parâmetros que são bastante difíceis de determinar, sendo esta escolha de fundamental importância para o desempenho de previsão do modelo. Assim, a grande questão para otimizar o desempenho de rede MLP para problemas de previsão é como determinar os valores ótimos destes parâmetros da rede, bem como determinar os retardos temporais relevantes para caracterizar o fenômeno gerador da série temporal, uma vez que estes se encontram num espaço que contém infinitas possibilidades. Uma possível solução é a aplicação de uma técnica de busca que tenha a capacidade de realizar a escolha automática destes parâmetros, alcançando pontos de ótimo dentro deste espaço de parâmetros [60, 61].

Apesar de redes MLP serem aproximadores universais [74, 75], estas possuem um custo computacional bastante elevado para determinar um conjunto de parâmetros que solucione o

problema de previsão com desempenho aceitável, como a quantidade de neurônios na camada escondida, os pesos sinápticos, a arquitetura, o algoritmo de treinamento e seus respectivos parâmetros, dentre outros [60, 61]. A definição incorreta destes parâmetros afeta diretamente a eficiência do processo de aprendizagem da rede e conseqüentemente implica na incapacidade de generalização da rede.

Os GAs representam uma classe de algoritmos que realizam buscas evolucionárias em espaços de estados. Então, a combinação MLP e GAs, gerando uma solução híbrida, otimiza o desempenho do modelo de previsão (no caso particular uma rede MLP), uma vez que o GA busca, de forma evolucionária, por uma configuração sub-ótima (senão ótima) da rede MLP para o problema em questão. Neste tipo de solução híbrida, cada indivíduo da população do GA representa uma rede candidata à solução final do problema, onde os parâmetros de cada rede MLP são evoluídos a cada geração do GA.

Desta forma, o HS definido como MGA–MLP, representado pela combinação de redes MLP e o MGA apresentado na Seção 3.5.1.1 (pois o mesmo apresenta convergência mais rápida do que o GA padrão [85]) segue o paradigma de intensificação, uma vez que onde o ponto forte do MGA, determinado pela busca de parâmetros em espaços de estados, supre o ponto fraco das redes MLP, determinado pela escolha de um conjunto de parâmetros que por definição está contido em um espaço tão imenso que se torna inviável a sua escolha manual.

O HS MGA–MLP consiste em: três modelos distintos são utilizados para descrever a arquitetura da rede MLP, onde cada modelo é treinado e ajustado pelo MGA, que determina [60,61]:

1. A quantidade mínima de (e seus específicos) retardos temporais relevantes para representar a série temporal – inicialmente a quantidade máxima de retardos temporais (*MaxLags*) é previamente definida e o MGA pode escolher qualquer valor no intervalo $[1, MaxLags]$;
2. A quantidade de unidades de processamento na camada intermediária da rede MLP – inicialmente a quantidade máxima de unidades de processamento é previamente definida (*MaxHidden*) e o MGA pode escolher qualquer valor no intervalo $[1, MaxHidden]$;
3. O modelo, a configuração dos parâmetros, pesos e *bias* da rede MLP.

A Figura 3.4 ilustra o esquema do HS MGA–MLP [60, 61].

De acordo com a Figura 3.4, os parâmetros do MGA e MLP são inicializados. Após este procedimento, a população do MGA é gerada, isto é, um conjunto de redes MLP candidatas a solução do problema. Posteriormente, cada indivíduo da população do MGA é avaliado. Um par de indivíduos pais são escolhidos e os operadores de cruzamento e mutação são aplicados, gerando uma nova população. A condição de parada é então verificada. Todo este procedimento é iterativamente repetido até que uma condição de parada seja encontrada.

No HS MGA–MLP, como cada indivíduo da população do MGA representa uma rede MLP, a codificação do seu cromossomo consiste nos seguintes parâmetros [60, 61]:

- W_{ij} : matriz com os pesos das conexões da camada de entrada i para a camada intermediária j da rede MLP;
- W_{jk} : matriz com os pesos das conexões da camada intermediária j para a camada de saída k da rede MLP;

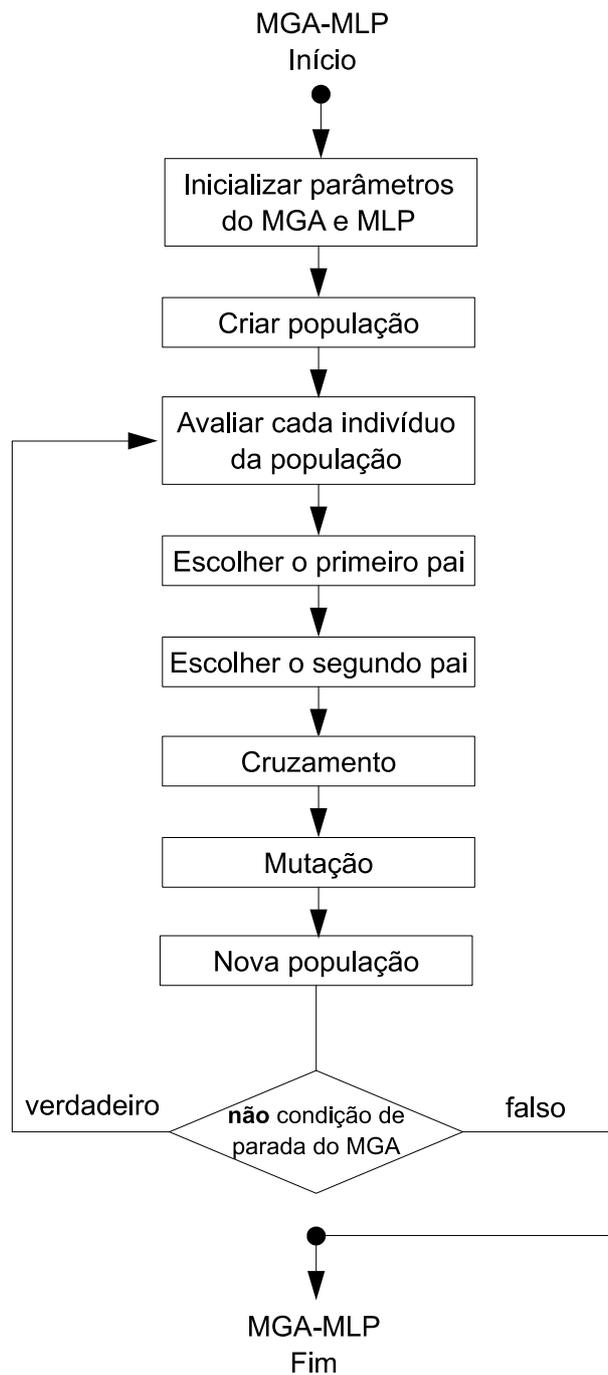


Figura 3.4 Esquema do sistema híbrido MGA-MLP.

- \mathbf{b}_j : vetor com os *bias* das unidades de processamento da camada intermediária j da rede MLP;
- \mathbf{b}_k : vetor com os *bias* das unidades de processamento da camada de saída k da rede MLP;
- $MlpMod$: modelo da MLP;

- *MlpHiddenUnit*: quantidade de unidades de processamento na camada escondida da rede MLP;
- **lag**: vetor contendo no máximo *MaxLags* elementos, onde cada um destes é utilizado para determinar se um retardo temporal específico será utilizado ($lag_i \geq 0$) ou não ($lag_i < 0$).

Neste tipo de HS, cada indivíduo da população do MGA é avaliado através de uma função de avaliação, definida por [60, 61]:

$$\text{Fitness} = \frac{1}{1 + \text{MSE}}, \quad (3.43)$$

onde MSE é o erro médio quadrático (*mean squared error*, MSE), formalmente definido no Capítulo 4.

Os critérios de parada do HS MGA–MLP são [60, 61]:

1. A quantidade máxima de gerações (*MgaMaxGer*);
2. O aumento no erro de validação ou *generalization loss* (*Gl*) [110]: $Gl > 5\%$;
3. A queda no erro de treinamento ou *process training* (*Pt*) [110]: $Pt \leq 10^{-6}$.

Três formas distintas ($MlpMod = 1, 2, 3$) para modelagem da rede MLP são utilizadas neste HS, e são descritas formalmente a seguir [60, 61].

A primeira arquitetura para modelagem da rede MLP ($MlpMod = 1$) utiliza função de ativação sigmóide logística para todas as unidades de processamento escondidas. A unidade de processamento de saída utiliza função de ativação linear com seu bias passando por função sigmóide logística. A saída da rede MLP é dada por:

$$y_k(t) = \sum_{j=1}^{n_h} W_{jk} \text{Sig} \left[\sum_{i=1}^{n_{in}} W_{ij} x_i(t) + b_j^1 \right] + \text{Sig}(b_k^2), \quad (3.44)$$

onde $x_i(t)$ ($i = 1, 2, \dots, n_{in}$) representa os valores de entrada da rede MLP (retardos temporais), n_{in} e n_h denotam a quantidade de entradas da rede MLP e a quantidade de unidades de processamento na camada escondida, respectivamente. Como a previsão pretendida é de um passo, utiliza-se apenas uma unidade de processamento na camada de saída ($k = 1$). O termo $\text{Sig}(\cdot)$ é uma função sigmóide logística definida por:

$$\text{Sig}(x) = \frac{1}{1 + \exp(-x)}. \quad (3.45)$$

O segundo modelo de rede MLP ($MlpMod = 2$) consiste em utilizar função de ativação sigmóide logística nas unidades de processamento na camada escondida e utilizar função de ativação linear na unidade de processamento na camada de saída. A saída da rede MLP é dada por:

$$y_k(t) = \sum_{j=1}^{n_h} W_{jk} \text{Sig} \left[\sum_{i=1}^{n_{in}} W_{ij} x_i(t) + b_j^1 \right] + b_k^2. \quad (3.46)$$

A terceira arquitetura de rede MLP ($MlpMod = 3$) aplica a função de ativação sigmóide logística em todas as unidades de processamento. A saída da rede MLP é dada por:

$$y_k(t) = Sig \left\{ \sum_{j=1}^{n_h} W_{jk} Sig \left[\sum_{i=1}^{n_{in}} W_{ij} x_i(t) + b_j^1 \right] + b_k^2 \right\}. \quad (3.47)$$

3.6.2 Otimizador por Exame de Partículas e Redes Neurais Multicamadas

Este tipo de HS, definido como PSO–MLP [25], é composto por PSO e redes MLP e visa integrar o destacável mecanismo de busca por exame de partículas (uma vez que o mesmo apresenta um maior poder de busca para otimização de redes MLP quando comparado aos GA padrão e o MGA utilizado no HS exposto na Seção 3.6.1) com as bem conhecidas redes MLP. O PSO representa uma classe de algoritmos de otimização em espaços de estados que realizam buscas baseadas em enxame de partículas. A combinação MLP e PSO gera, portanto, uma solução híbrida que também otimiza o desempenho do modelo de previsão, uma vez que o PSO busca por uma configuração sub-ótima (ou até mesmo ótima) da rede MLP para o problema em questão.

Neste tipo de solução híbrida, cada partícula do enxame do PSO representa uma rede MLP candidata à solução final do problema, onde os parâmetros de cada rede MLP são evoluídos a cada iteração do PSO. Da mesma forma que o HS MGA–MLP, o HS PSO–MLP exposto aqui também segue o paradigma de intensificação, onde o ponto forte do PSO supre o ponto fraco das redes MLP.

O HS PSO–MLP consiste em: três modelos distintos são utilizados para descrever a arquitetura da rede MLP, onde cada modelo é treinado e ajustado pelo PSO, que determina (da mesma forma que o MGA–MLP) [25]:

1. A quantidade mínima de (e seus específicos) retardos temporais relevantes para representar a série temporal – inicialmente a quantidade máxima de retardos temporais ($MaxLags$) é previamente definida e o PSO pode escolher qualquer valor no intervalo $[1, MaxLags]$;
2. A quantidade de unidades de processamento na camada intermediária da rede MLP – inicialmente a quantidade máxima de unidades de processamento é previamente definida ($MaxHidden$) e o PSO pode escolher qualquer valor no intervalo $[1, MaxHidden]$;
3. O modelo, a configuração dos parâmetros, pesos e *bias* da rede MLP.

A Figura 3.5 ilustra o esquema do HS PSO–MLP [25].

De acordo com a Figura 3.5, os parâmetros do PSO e MLP são inicializados. Após este procedimento, o enxame do PSO é gerado, isto é, um conjunto de redes MLP candidatas a solução do problema. Posteriormente, cada partícula do enxame do PSO é avaliada. Então, a velocidade, a posição e a melhor posição individual de cada partícula são atualizadas, bem como a melhor posição global encontrada pelo enxame. A condição de parada é então verificada. Todo este procedimento é iterativamente repetido até que uma condição de parada seja encontrada.

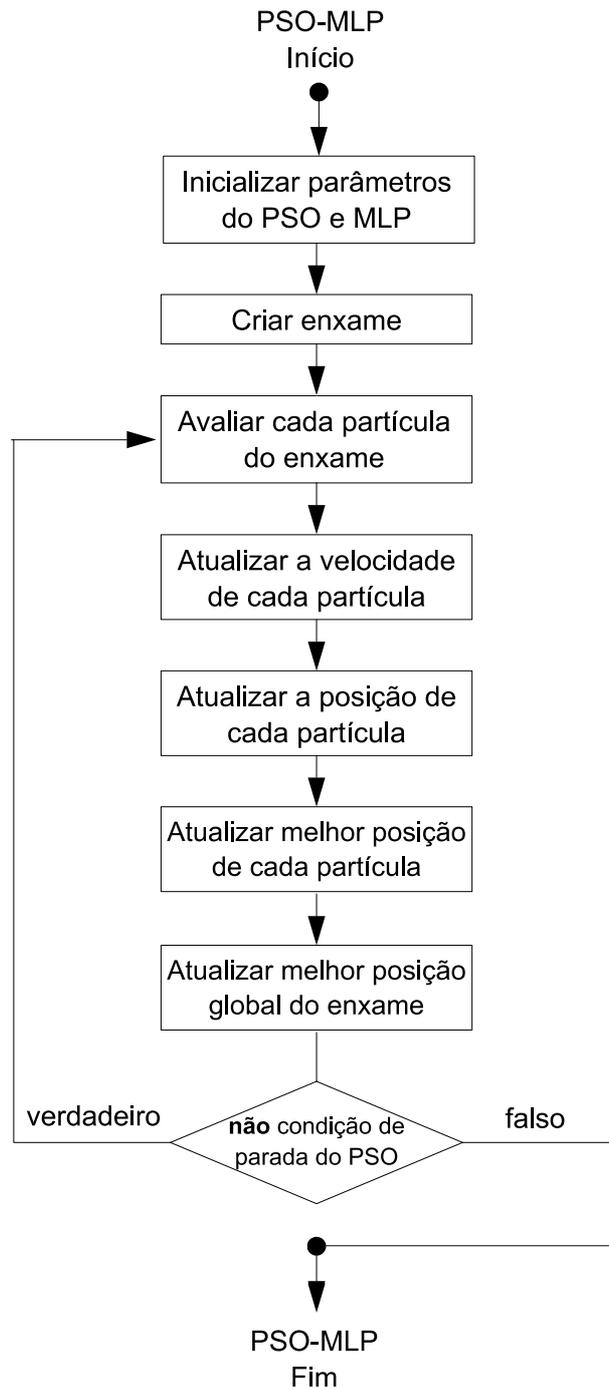


Figura 3.5 Esquema do sistema híbrido PSO-MLP.

No HS PSO-MLP, como cada partícula do enxame do PSO representa uma rede MLP, a codificação do cromossomo de cada partícula do enxame do PSO consiste nos mesmos parâmetros descritos no HS MGA-MLP. Neste HS PSO-MLP, cada partícula do enxame do PSO é avaliada através da mesma função de avaliação utilizada no HS MGA-MLP.

Os critérios de parada do HS PSO–MLP são [25]:

1. A quantidade máxima de iterações (*PsoMaxIter*);
2. O aumento no erro de validação ou *generalization loss* (*Gl*) [110]: $Gl > 5\%$;
3. A queda no erro de treinamento ou *process training* (*Pt*) [110]: $Pt \leq 10^{-6}$.

As três formas distintas para modelagem da rede MLP apresentadas na Seção 3.6.1 também são utilizadas no HS PSO–MLP.

Note que ambos os HSs, MGA–MLP e PSO–MLP otimizam os mesmos parâmetros da rede MLP, onde a ideia é apresentar diferentes técnicas de busca para solucionar o mesmo problema em questão, para desta forma poder inferir uma análise comparativa justa entre as técnicas [25].

3.7 Sistemas Híbridos Otimizados

Os sistemas híbridos expostos nas Seções 3.6.1 e 3.6.2 representam combinações entre algoritmos de busca e redes neurais para otimizar o desempenho de previsão das redes MLP. No entanto, estes sistemas não abordam o meio convencional de aprendizagem de redes MLP, que é aprendizagem supervisionada baseada no método do gradiente descendente [74,75]. Assim, é importante que as propostas de sistemas híbridos incorporem ambas as técnicas de busca global (MGA e PSO) e local (algoritmos baseado em gradiente descendente), uma vez que a técnica global atua sobre todo o espaço de busca e a técnica local refina esta busca sobre uma região específica deste espaço. Esta seção apresentará dois HSs que utilizam esta abordagem e representam as melhores técnicas, baseadas em seus resultados preditivos, de previsão atualmente apresentadas na literatura [28, 33, 36, 41].

3.7.1 O método *time-delay added evolutionary forecasting* (TAEF)

O método *time-delay added evolutionary forecasting* (TAEF) [60,61] tenta reconstruir o espaço de fase de uma série temporal realizando uma busca pela mínima dimensionalidade necessária para reproduzir o fenômeno da série temporal. O método TAEF é um HS baseado em arquiteturas de rede MLP treinadas e ajustadas pelo MGA descrito na Seção 3.5.1.1 que não apenas determina os parâmetros da rede MLP como também pela dimensão mínima adequada para representar a série temporal.

Seguindo este princípio, os parâmetros importantes que método TAEF determina são [60, 61]:

1. A quantidade mínima de (e seus específicos) retardos temporais relevantes para representar a série temporal – inicialmente a quantidade máxima de retardos temporais (*MaxLags*) é previamente definida e o MGA pode escolher qualquer valor no intervalo $[1, MaxLags]$;
2. A quantidade de unidades de processamento na camada intermediária da rede MLP – inicialmente a quantidade máxima de unidades de processamento é previamente definida (*MaxHidden*) e o MGA pode escolher qualquer valor no intervalo $[1, MaxHidden]$;

3. O modelo e arquitetura da rede MLP (Equações 3.44–3.47);
4. O algoritmo de treinamento da rede MLP (*resilient back-propagation* [117], *levenberg marquardt* [71], *scaled conjugate gradient* [97] e *one step secant conjugate gradient* [5]);

A Figura 3.6 ilustra o esquema do método TAEF [60, 61].

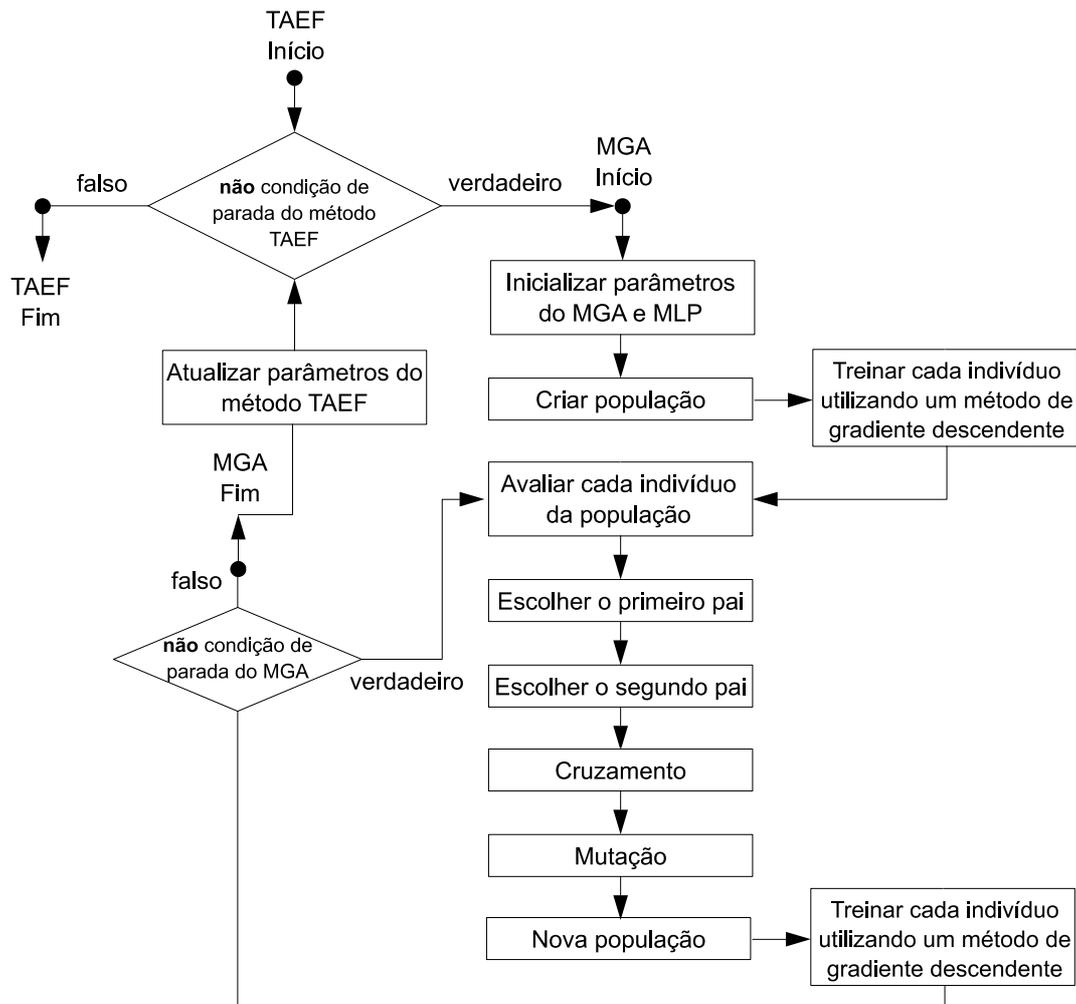


Figura 3.6 Esquema do sistema híbrido otimizado TAEF.

De acordo com a Figura 3.6, o método TAEF executa iterativamente o processo do MGA para determinar o modelo de previsão. Inicialmente, em cada iteração do método, os parâmetros do MGA são inicializados. Após este procedimento, a população do MGA é criada, isto é, um conjunto de redes MLP candidatas a solução do problema. Cada indivíduo (rede MLP) é então treinado por um método baseado em gradiente descendente (definido individualmente para cada indivíduo da população do MGA). Posteriormente, cada indivíduo da população do MGA é avaliado. Um par de indivíduos pais são escolhidos e os operadores de cruzamento e mutação são aplicados, gerando uma nova população. Cada indivíduo da nova população (rede

MLP) é então treinado por um método baseado em gradiente descendente (definido individualmente para cada indivíduo da nova população do MGA). A condição de parada do MGA é então verificada. Todo este procedimento é iterativamente repetido até que uma condição de parada do MGA seja encontrada. Após o término do processo do MGA, os parâmetros do método TAEF são atualizados e este procedimento é iterativamente repetido até que uma condição de parada do método TAEF seja encontrada.

No método TAEF, como cada indivíduo da população do MGA representa uma rede MLP, a codificação do seu cromossomo consiste nos seguintes parâmetros [60, 61]:

- *MlpMod*: modelo da MLP;
- *MlpHiddenUnit*: quantidade de unidades de processamento na camada escondida da rede MLP;
- *MlpTrain*: algoritmo de treinamento da rede MLP;
- **lag**: vetor contendo no máximo *MaxLags* elementos, onde cada um destes é utilizado para determinar se um retardo temporal específico será utilizado ($lag_i \geq 0$) ou não ($lag_i < 0$).

Três formas distintas ($MlpMod = 1, 2, 3$) para modelagem da MLP são utilizadas neste método, sendo as mesmas utilizadas nos HSs MGA-MLP e PSO-MLP.

No método TAEF, cada indivíduo da população do MGA é avaliado através de uma função de avaliação, definida por [60, 61]:

$$\text{Fitness} = \frac{1}{1 + \text{MSE}}, \quad (3.48)$$

onde MSE é o erro médio quadrático (*mean squared error*, MSE), formalmente definido no Capítulo 4.

Os critérios de parada do MGA utilizado no método TAEF são [60, 61]:

1. A quantidade máxima de gerações (*MgaMaxGer*);
2. O aumento no erro de validação ou *generalization loss* (*Gl*) [110]: $Gl > 5\%$;
3. A queda no erro de treinamento ou *process training* (*Pt*) [110]: $Pt \leq 10^{-6}$.

Os critérios de parada do método TAEF são [60, 61]:

1. A quantidade máxima de iterações (*iterMax*);
2. Valor mínimo da função de avaliação (*minFit*);

Os critérios de parada do treinamento de cada rede MLP são [60, 61]:

1. A quantidade máxima de épocas (*MlpEpoch*);
2. O aumento no erro de validação ou *generalization loss* (*Gl*) [110]: $Gl > 5\%$;

3. A queda no erro de treinamento ou *process training* (P_t) [110]: $P_t \leq 10^{-6}$.

Ao fim do método TAEF, o indivíduo com o maior valor de avaliação no conjunto de validação dentre todos gerados por cada execução do MGA é escolhido para ser o modelo de previsão gerado pelo método [60, 61].

3.7.2 O método *swarm-based hybrid intelligent forecasting* (SHIF)

A partir da definição do método TAEF, é possível notar que ele incorpora duas técnicas de busca, uma global para definição dos retardos temporais, modelo e algoritmo de treinamento da rede MLP, e outra local para treinar a rede MLP. No entanto, no método TAEF os pesos e *bias* da rede MLP são inicializados aleatoriamente para cada indivíduo criado, e tal fato induz uma limitação no processo evolucionário executado pelo MGA.

Na tentativa de superar esta limitação do método TAEF, Araújo [25] propôs o método *swarm-based hybrid intelligent forecasting* (SHIF), que da mesma forma do método TAEF [60, 61], reconstrói o espaço de fase de uma série temporal através de uma busca pela mínima dimensionalidade necessária para reproduzir o fenômeno da série temporal. O método SHIF é um HS baseado em arquiteturas de rede MLP treinadas e ajustadas pelo PSO descrito na Seção 3.5.2 que não apenas determina os parâmetros da rede MLP como também pela dimensão mínima adequada para representar a série temporal.

Portanto, os parâmetros relevantes que método SHIF determina são [25]:

1. A quantidade mínima de (e seus específicos) retardos temporais relevantes para representar a série temporal – inicialmente a quantidade máxima de retardos temporais ($MaxLags$) é previamente definida e o PSO pode escolher qualquer valor no intervalo $[1, MaxLags]$;
2. A quantidade de unidades de processamento na camada intermediária da rede MLP – inicialmente a quantidade máxima de unidades de processamento é previamente definida ($MaxHidden$) e o PSO pode escolher qualquer valor no intervalo $[1, MaxHidden]$;
3. O modelo e arquitetura da rede MLP (Equações 3.49–3.52);
4. Os pesos, *bias* e chaves das conexões (procedimento de poda) da rede MLP;
5. O algoritmo de treinamento da rede MLP (*resilient back-propagation* [117], *levenberg marquardt* [71], *scaled conjugate gradient* [97] e *one step secant conjugate gradient* [5]);

A Figura 3.7 ilustra o esquema do método SHIF [25].

De acordo com a Figura 3.7, o método SHIF executa iterativamente o processo do PSO para determinar os parâmetros ótimos do modelo de previsão. Inicialmente, em cada iteração do método, os parâmetros do PSO são inicializados. Após este procedimento, o enxame do PSO é criado, isto é, um conjunto de redes MLP candidatas a solução do problema. Cada partícula (rede MLP) é então treinada por um método baseado em gradiente descendente (definido distintamente para cada partícula do enxame do PSO). Posteriormente, cada partícula do enxame do PSO é avaliada. A posição e a velocidade de cada partícula são atualizadas. A melhor posição individual e global são então atualizadas. Tem-se então um novo enxame, que

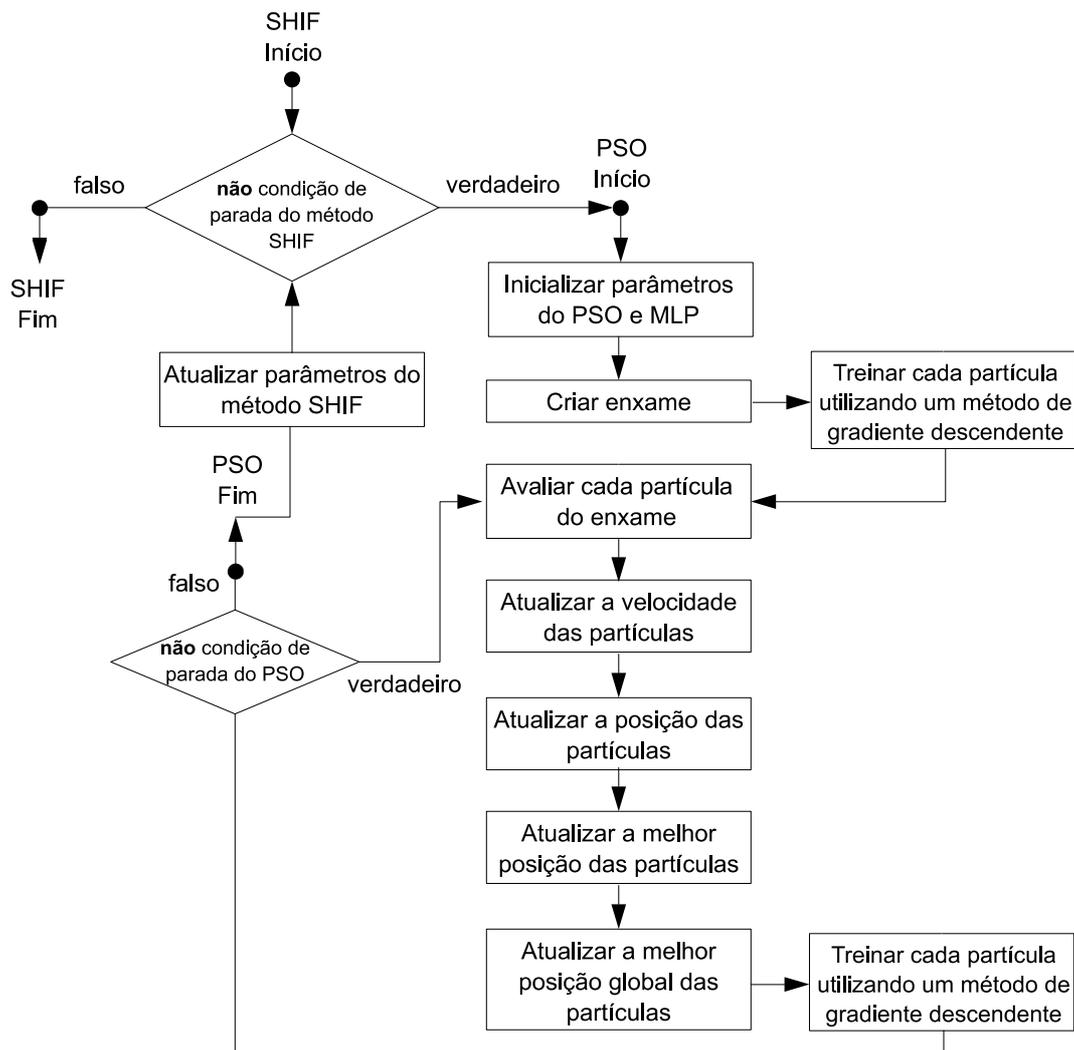


Figura 3.7 Esquema do sistema híbrido otimizado SHIF.

é treinado por um método baseado em gradiente descendente (definido individualmente para cada partícula do enxame do PSO). A condição de parada do PSO é então verificada. Todo este procedimento é iterativamente repetido até que uma condição de parada do PSO seja encontrada. Após o término do processo do PSO, os parâmetros do método SHIF são atualizados e este procedimento é iterativamente repetido até que uma condição de parada do método SHIF seja encontrada.

No método SHIF, como cada partícula do enxame do PSO representa uma rede MLP, a codificação de cada partícula consiste nos seguintes parâmetros [25]:

- *MlpMod*: modelo da MLP;
- *MlpHiddenUnit*: quantidade de unidades de processamento na camada escondida da rede MLP;

- \mathbf{W}_{ij} : matriz com os pesos das conexões da camada de entrada i para a camada intermediária j da rede MLP;
- \mathbf{S}_{ij} : matriz com as chaves das conexões da camada de entrada i para a camada intermediária j da rede MLP;
- \mathbf{W}_{jk} : matriz com os pesos das conexões da camada intermediária j para a camada de saída k da rede MLP;
- \mathbf{S}_{jk} : matriz com as chaves das conexões da camada intermediária j para a camada de saída k da rede MLP;
- \mathbf{b}_j : vetor com os *bias* das unidades de processamento da camada intermediária j da rede MLP;
- \mathbf{S}_j^1 : vetor com os *bias* das unidades de processamento da camada intermediária j da rede MLP;
- \mathbf{b}_k : vetor com os *bias* das unidades de processamento da camada de saída k da rede MLP;
- \mathbf{S}_k^2 : vetor com os *bias* das unidades de processamento da camada de saída k da rede MLP;
- *MlpTrain*: algoritmo de treinamento da rede MLP;
- **lag**: vetor contendo no máximo *MaxLags* elementos, onde cada um destes é utilizado para determinar se um retardo temporal específico será utilizado ($lag_i \geq 0$) ou não ($lag_i < 0$).

Três formas distintas ($MlpMod = 1, 2, 3$) para modelagem da MLP são utilizadas neste método, e são descritas formalmente a seguir. Note que diferentemente do método TAEF, o método SHIF usa um procedimento de poda na modelagem da MLP, incluindo chaves entre as conexões da rede. Tais chaves podem estar ligadas ou desligadas, representando a atividade ou inatividade de uma determinada conexão na rede. Este procedimento de poda oferece uma capacidade efetiva para determinação da rede MLP mais compacta, reduzindo o custo computacional e a probabilidade de *overfitting* da rede.

A primeira arquitetura para modelagem da MLP ($MlpMod = 1$) utiliza função de ativação sigmóide logística para todas as unidades de processamento escondidas. A unidade de processamento de saída utiliza função de ativação linear com seu *bias* passando por função sigmóide logística. A saída da MLP é dada por:

$$y_k = \sum_{j=1}^{n_h} step(S_{jk}^2) W_{jk} Sig \left[\sum_{i=1}^{n_m} (step(S_{ij}^1) W_{ij} x_i(t) - step(S_j^1) b_j^1) \right] - step(S_k^2) Sig(b_k^2), \quad (3.49)$$

onde o termo *step* é uma função degrau definida por:

$$step(x) = \begin{cases} 1, & \text{se } x \geq 0, \\ 0, & \text{caso contrário.} \end{cases} \quad (3.50)$$

O segundo modelo ($MlpMod = 2$) consiste em utilizar função de ativação sigmóide logística nas unidades de processamento na camada escondida e utilizar função de ativação linear na unidade de processamento na camada de saída. A saída da MLP é dada por:

$$y_k = \sum_{j=1}^{n_h} step(S_{jk}^2) W_{jk} Sig \left[\sum_{i=1}^{n_{in}} (step(S_{ij}^1) W_{ij} x_i(t) - step(S_j^1) b_j^1) \right] - step(S_k^2) (b_k^2). \quad (3.51)$$

A terceira arquitetura ($MlpMod = 3$) aplica a função de ativação sigmóide logística em todas as unidades de processamento. A saída da MLP é dada por:

$$y_k = Sig \left\{ \sum_{j=1}^{n_h} step(S_{jk}^2) W_{jk} Sig \left[\sum_{i=1}^{n_{in}} (step(S_{ij}^1) W_{ij} x_i(t) - step(S_j^1) b_j^1) \right] - step(S_k^2) (b_k^2) \right\}. \quad (3.52)$$

No método SHIF, cada partícula do enxame do PSO é avaliada através de uma função de avaliação, definida por [25]:

$$Fitness = \frac{POCID}{1 + MSE + MAPE + THEIL + ARV}, \quad (3.53)$$

onde POCID é a previsão na mudança de direção (*prediction of change in direction*, POCID), MAPE é o erro percentual absoluto médio (*mean absolute percentage error*, MAPE), MAPE é a estatística U de Theil (*u of theil statistics*, THEIL), e a variância relativa média (*average relative variance*, ARV), formalmente definidos no Capítulo 4.

Os critérios de parada do PSO utilizado no método SHIF são [25]:

1. A quantidade máxima de iterações do pso ($PsoMaxIter$);
2. O aumento no erro de validação ou *generalization loss* (Gl) [110]: $Gl > 5\%$;
3. A queda no erro de treinamento ou *process training* (Pt) [110]: $Pt \leq 10^{-6}$.

Os critérios de parada do método SHIF são [25]:

1. A quantidade máxima de iterações ($iterMax$);
2. Valor mínimo da função de avaliação ($minFit$);

Os critérios de parada do treinamento de cada rede MLP são [25]:

1. A quantidade máxima de épocas ($MlpEpoch$);
2. O aumento no erro de validação ou *generalization loss* (Gl) [110]: $Gl > 5\%$;
3. A queda no erro de treinamento ou *process training* (Pt) [110]: $Pt \leq 10^{-6}$.

Ao fim do método SHIF, o indivíduo com o maior valor de avaliação no conjunto de validação dentre todos gerados por cada execução do PSO, é escolhido para ser o modelo de previsão gerado pelo método [25].

3.8 Considerações Finais

Segundo Araújo, Ferreira e Mattos [25, 28, 33, 36, 41, 49, 60, 61], é possível identificar que séries temporais são comumente governadas por fenômenos não-lineares, portanto era de se esperar um desempenho superior dos modelos não-lineares em relação aos modelos lineares. No entanto, acredita-se que os modelos não-lineares não são capazes de obter desempenho superior devido à grande quantidade de características complexas no fenômeno gerador de séries temporais, que impedem o modelo de capturar tal fenômeno. No entanto, o problema de previsão de séries temporais ainda é considerado um problema bastante difícil, devido à grande quantidade de características complexas que, frequentemente, estão presentes em tais séries (irregularidades, volatilidade, tendências, ruído, etc.).

Devido à complexidade do relacionamento entre os dados, surgem várias limitações dos modelos não-lineares quando aplicados em situações reais. Uma destas limitações é a alta complexidade matemática e computacional, fator que limita os modelos não-lineares a um desempenho equivalente aos modelos lineares, bem como a necessidade, na maioria das vezes, de um especialista no problema para validar as previsões geradas pelo modelo [14]. Tais fatores sugerem que novas abordagens devem ser desenvolvidas com o intuito de aperfeiçoar o desempenho de previsão. Portanto, não é surpreendente o grande interesse no desenvolvimento de novos modelos para a previsão de séries temporais, a partir da utilização de novas abordagens e paradigmas aplicados ao problema exposto.

3.9 Resumo do Capítulo

Neste capítulo foram apresentados os modelos de previsão estatísticos clássicos lineares e não-lineares encontrados na literatura de previsão de séries temporais. Também, foram abordados modelos de previsão mais robustos, baseados em métodos e técnicas da inteligência artificial. Posteriormente, foram apresentados modelos de previsão híbridos que representam o que há de mais recente neste segmento de pesquisa. Entretanto, mesmo com diversas técnicas propostas na literatura, o dilema do passeio aleatório no caso particular de séries temporais financeiras ainda persiste, e tal fato sugere o desenvolvimento de novos modelos de previsão na tentativa de superar o problema.

Caracterização das Séries Temporais Utilizadas

Neste capítulo serão descritas as séries temporais financeiras utilizadas como conjunto de testes ao longo deste trabalho. Para cada série temporal analisada, é apresentada uma breve descrição da série e a motivação para sua utilização. Ao final deste capítulo, são apresentadas cinco métricas e uma função de avaliação utilizadas para mensurar o desempenho de previsão dos modelos investigados neste trabalho.

4.1 Séries Temporais Financeiras

O principal objetivo deste trabalho é o desenvolvimento de um novo modelo de previsão para solucionar o dilema do passeio aleatório no problema de previsão de séries temporais financeiras. Por este motivo, faz-se necessário a utilização de um conjunto de séries temporais relevantes que sirva como teste (*benchmark*) para comparações com os resultados obtidos com o modelo proposto e modelos propostos na literatura e investigados neste trabalho. Para tal, foram utilizadas séries temporais financeiras comumente utilizadas na literatura [25–30, 33, 36, 38, 39, 41, 49, 60, 61]: Índice da Bolsa de Valores de São Paulo, Índice *Dow Jones Industrial Average*, Índice *National Association of Securities Dealers Automated Quotation*, Índice *Financial Times and London Stock Exchange 100*, Preço das Ações do Bradesco PN, Preço das Ações da Gol PN, Preço das Ações do Itaú Unibanco PN, Preço das Ações da Petrobras PN, Preço das Ações da Usiminas PNA e Preço das Ações da Vale PNA.

4.1.1 Índice da Bolsa de Valores de São Paulo (BVSP)

O Índice da Bolsa de Valores de São Paulo (BVSP) é considerado o mais importante indicador do desempenho médio das cotações das ações com maior volume negociadas na Bolsa de Valores de São Paulo. A série BVSP utilizada é composta por observações diárias do valor de fechamento do índice a partir de 09/10/2006 até 28/10/2010, que podem ser ilustradas na Figura 4.1.

Algumas medidas estatísticas básicas para a série BVSP utilizada são apresentadas na Tabela 4.1.

A Figura 4.2 apresenta o gráfico *lagplot* para a série BVSP.

4.1.2 Índice *Dow Jones Industrial Average* (DJI)

O Índice *Dow Jones Industrial Average* (DJI) é considerado o mais importante índice de mercado financeiro internacional, que mostra como as trinta maiores empresas de capital aberto

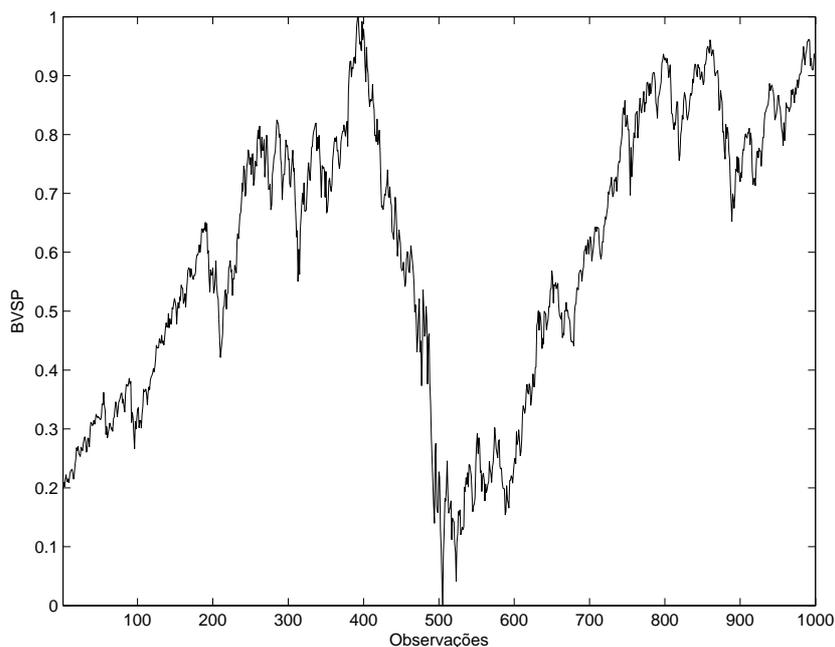


Figura 4.1 Gráfico da série BVSP.

Tabela 4.1 Estatísticas da série BVSP.

Estatística	Valor
Quantidade de pontos	1000,00
Valor máximo	73517,00
Valor mínimo	29435,00
Média	55733,29
Variância	113010203,45
Desvio padrão	10630,63

sediadas nos Estados Unidos têm sido negociadas durante o pregão do mercado de ações. A série DJI utilizada é composta por observações diárias do valor de fechamento do índice a partir de 08/11/2006 até 28/10/2010, que podem ser ilustradas na Figura 4.3.

A Tabela 4.2 apresenta algumas estatísticas básicas para a série DJI.

Tabela 4.2 Estatísticas da série DJI.

Estatística	Valor
Quantidade de pontos	1000,00
Valor máximo	14164,53
Valor mínimo	6547,05
Média	11024,80
Variância	3401094,56
Desvio padrão	1844,21

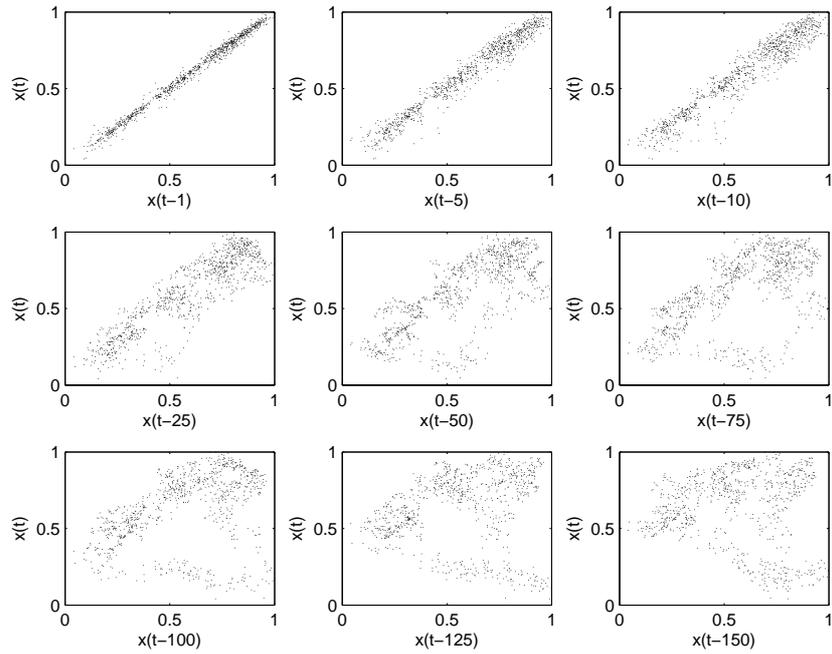


Figura 4.2 Lagplot da série BVSP.

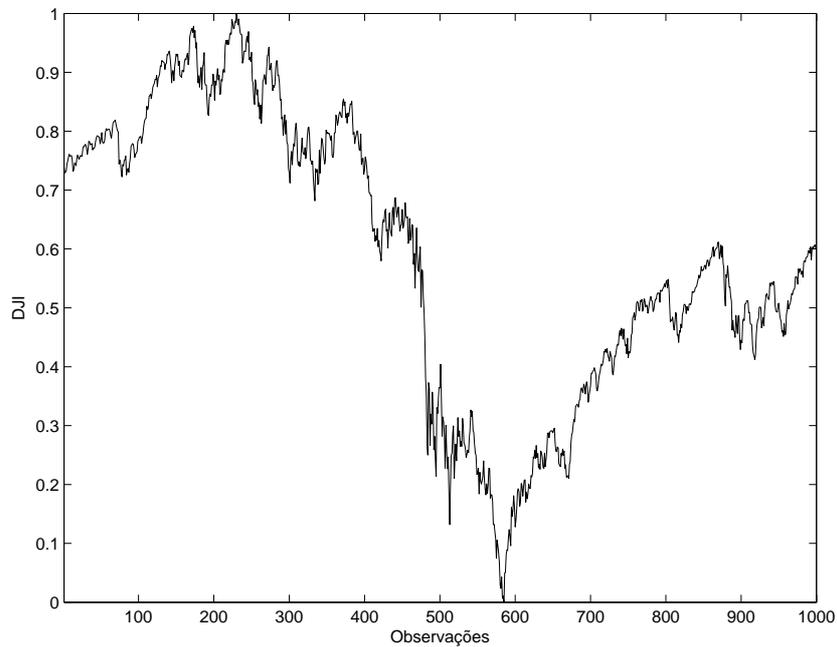


Figura 4.3 Gráfico da série DJI.

A Figura 4.4 apresenta o gráfico *lagplot* para a série DJI.

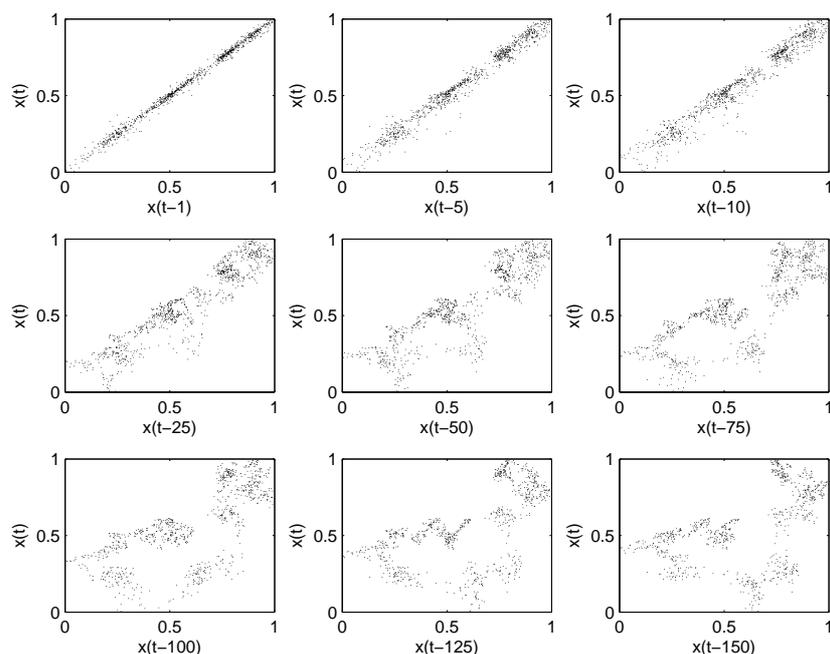


Figura 4.4 Lagplot da série DJI.

4.1.3 Índice *National Association of Securities Dealers Automated Quotation* (IXIC)

O Índice *National Association of Securities Dealers Automated Quotation* (IXIC) é o maior mercado de capitais dos Estados Unidos, sendo possível negociar ações de mais de três mil empresas. Atualmente, o índice possui o maior volume de negociação de todas as bolsas de valores mundiais. A série IXIC utilizada é composta por observações diárias do valor de fechamento do índice a partir de 08/11/2006 até 28/10/2010, que podem ser ilustradas na Figura 4.5.

Na Tabela 4.3 algumas estatísticas básicas são apresentadas para a série IXIC.

Tabela 4.3 Estatísticas da série IXIC.

Estatística	Valor
Quantidade de pontos	1000,00
Valor máximo	2859,12
Valor mínimo	1268,64
Média	2225,24
Variância	125415,17
Desvio padrão	354,14

A Figura 4.6 apresenta o gráfico *lagplot* para a série IXIC.

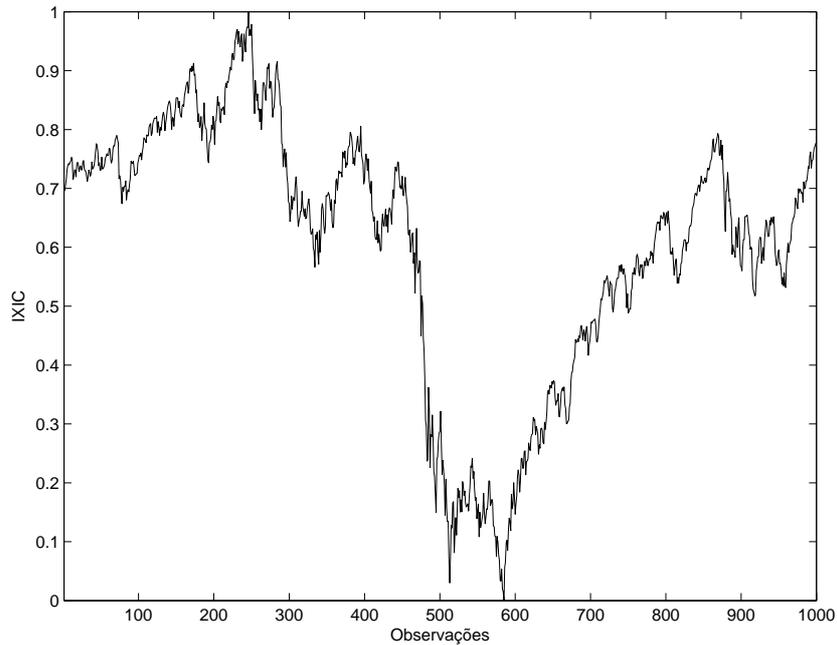


Figura 4.5 Gráfico da série IXIC.

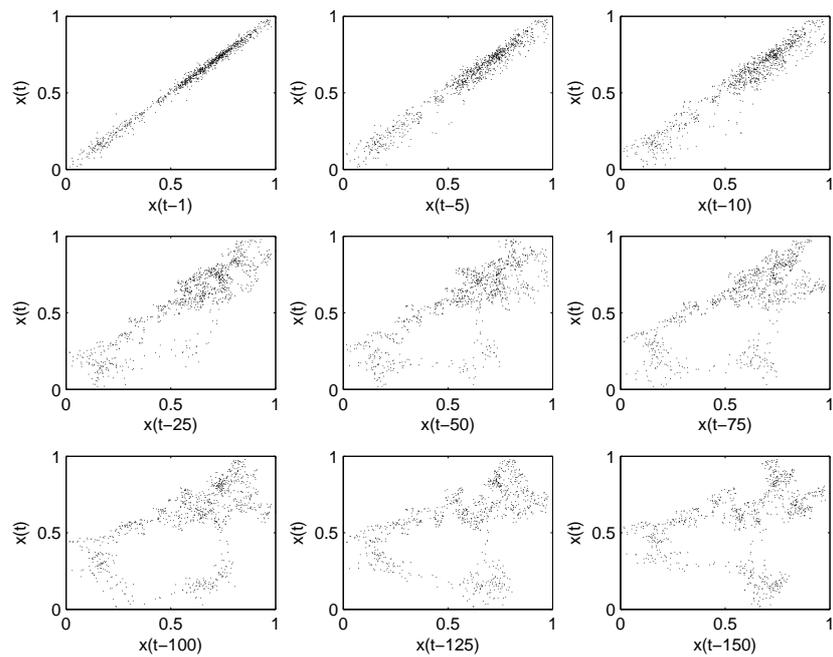


Figura 4.6 Lagplot da série IXIC.

4.1.4 Índice *Financial Times and London Stock Exchange 100* (FTSE)

O Índice *Financial Times and London Stock Exchange 100* (FTSE) é calculado pela *FTSE The Index Company* e é composto por uma carteira das cem ações mais representativas da Bolsa de

Valores de Londres, tendo como principal objetivo a detecção de tendências de alta ou baixa na cotação das ações. A série FTSE utilizada é composta por observações diárias do valor de fechamento do índice a partir de 14/11/2006 até 28/10/2010, que podem ser ilustradas na Figura 4.7.

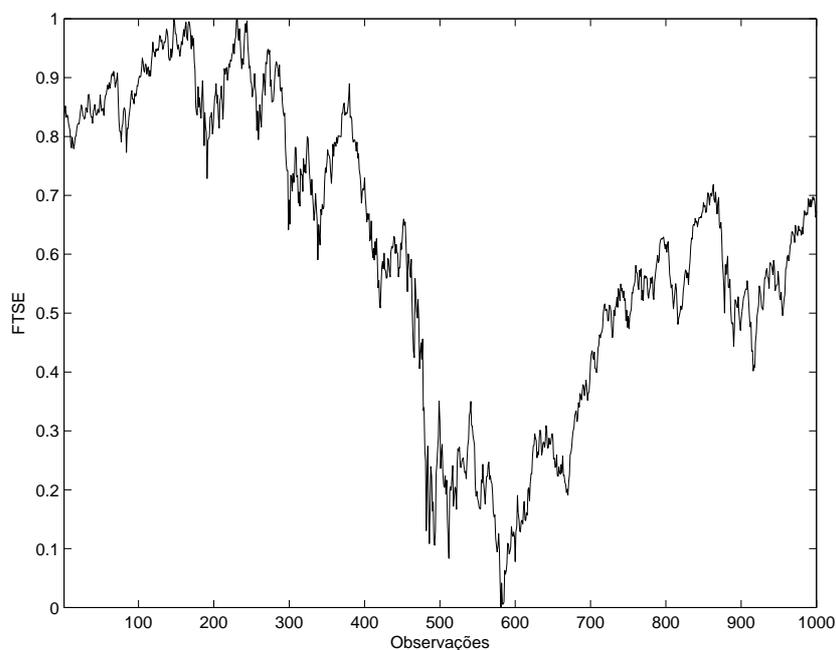


Figura 4.7 Gráfico da série FTSE.

A Tabela 4.4 apresenta algumas estatísticas básicas para a série FTSE.

Tabela 4.4 Estatísticas da série FTSE.	
Estatística	Valor
Quantidade de pontos	1000,00
Valor máximo	6732,40
Valor mínimo	3512,10
Média	5457,69
Variância	662310,35
Desvio padrão	813,83

A Figura 4.8 apresenta o gráfico *lagplot* para a série FTSE.

4.1.5 Preço das Ações do Bradesco PN (BBDC4)

O preço das ações do Bradesco PN (BBDC4) é constituído por registros diários referentes ao valor de fechamento das ações do Bradesco PN no período de 01/12/2006 até 28/10/2010, que podem ser ilustradas na Figura 4.9.

A Tabela 4.5 apresenta algumas estatísticas básicas para a série BBDC4.

A Figura 4.10 apresenta o gráfico *lagplot* para a série BBDC4.

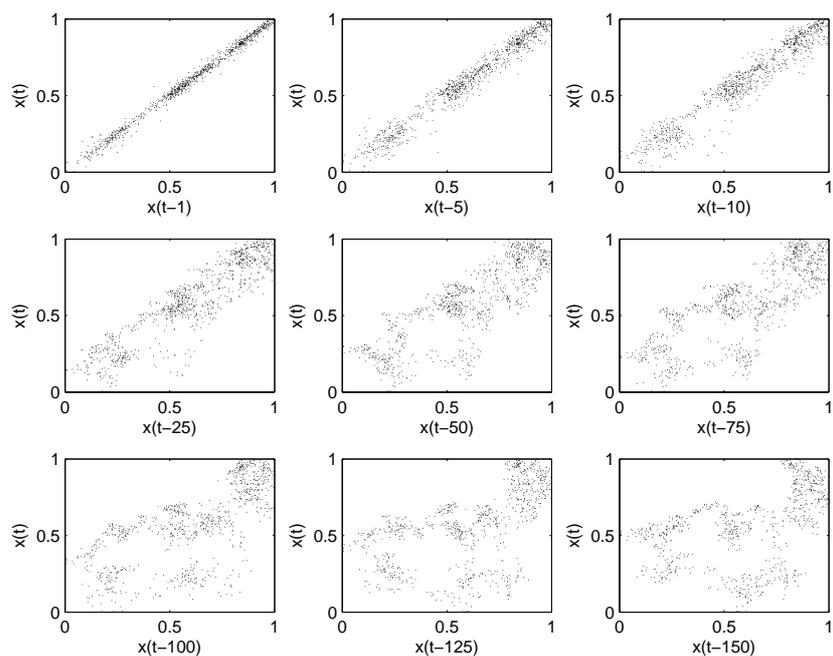


Figura 4.8 Lagplot da série FTSE.

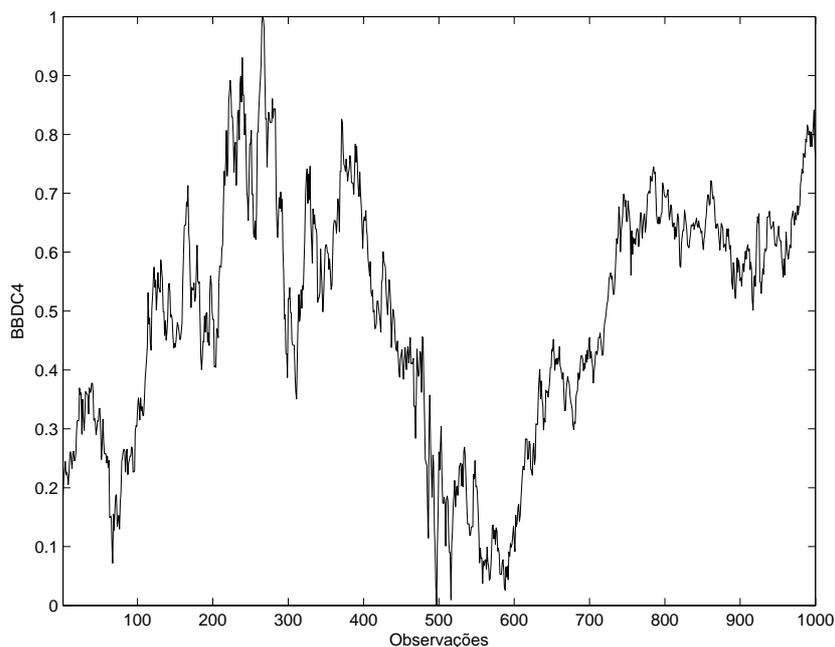


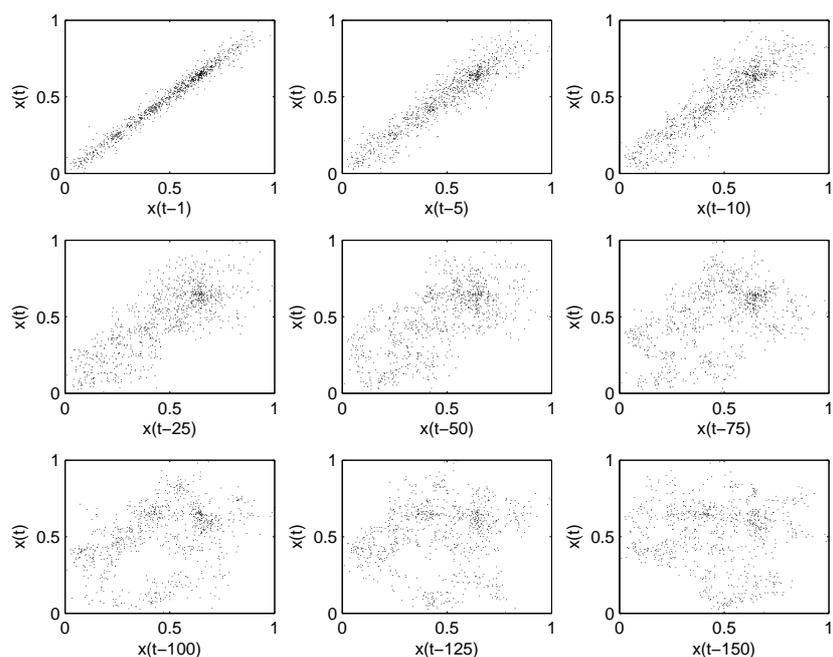
Figura 4.9 Gráfico da série BBDC4.

4.1.6 Preço das Ações da GOL PN (GOLL4)

O preço das ações da GOL PN (GOLL4) é constituído por registros diários referentes ao valor de fechamento das ações da GOL PN no período de 28/09/2006 até 28/10/2010, que podem ser

Tabela 4.5 Estatísticas da série BBDC4.

Estatística	Valor
Quantidade de pontos	1000,00
Valor máximo	41,00
Valor mínimo	15,14
Média	27,76
Variância	29,03
Desvio padrão	5,39

**Figura 4.10** Lagplot da série BBDC4.

ilustradas na Figura 4.11.

A Tabela 4.6 apresenta algumas estatísticas básicas para a série GOLL4.

Tabela 4.6 Estatísticas da série GOLL4.

Estatística	Valor
Quantidade de pontos	1000,00
Valor máximo	79,60
Valor mínimo	6,58
Média	30,33
Variância	356,77
Desvio padrão	18,89

A Figura 4.12 apresenta o gráfico lagplot para a série GOLL4.

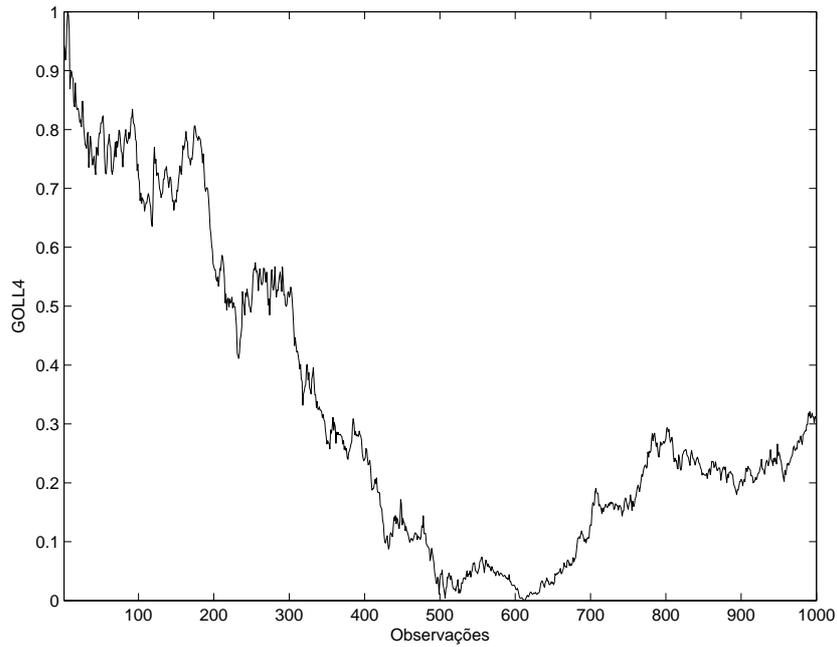


Figura 4.11 Gráfico da série GOLL4.

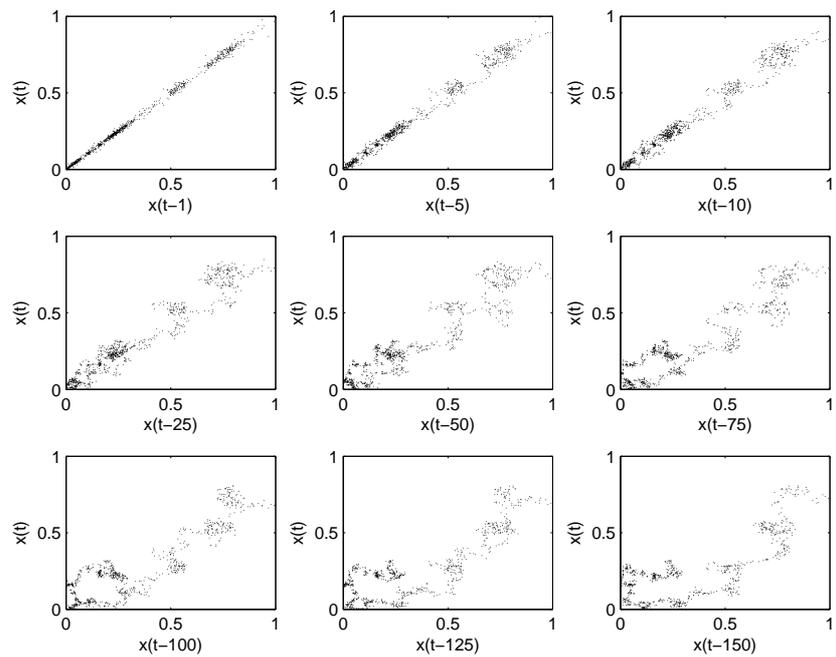


Figura 4.12 Lagplot da série GOLL4.

4.1.7 Preço das Ações do Itaú Unibanco PN (ITUB4)

O preço das ações do Itaú Unibanco PN (ITUB4) é constituído por registros diários referentes ao valor de fechamento das ações do Itaú Unibanco PN no período de 01/12/2006 até

28/10/2010, que podem ser ilustradas na Figura 4.13.

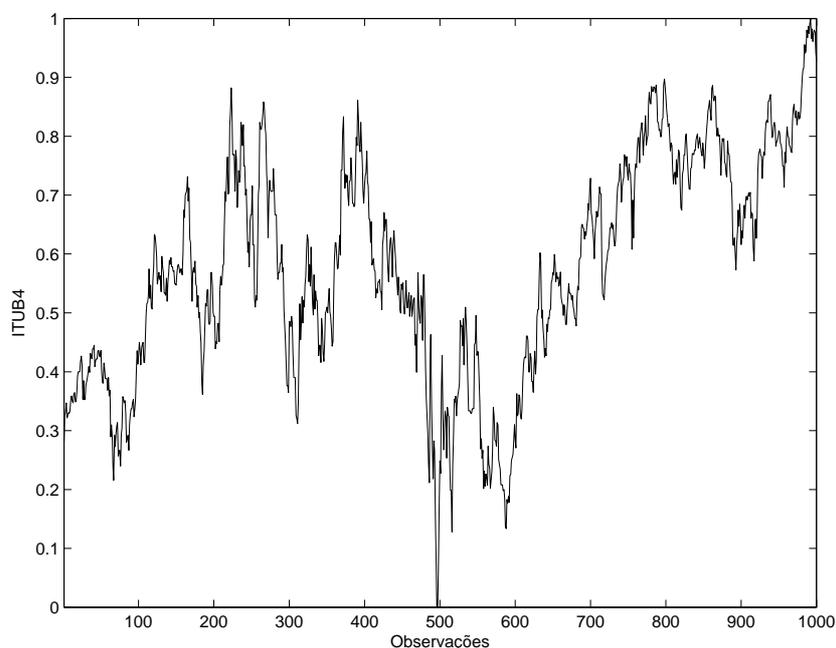


Figura 4.13 Gráfico da série ITUB4.

A Tabela 4.7 apresenta algumas estatísticas básicas para a série ITUB4.

Tabela 4.7 Estatísticas da série ITUB4.

Estatística	Valor
Quantidade de pontos	1000,00
Valor máximo	43,00
Valor mínimo	17,50
Média	32,15
Variância	23,33
Desvio padrão	4,83

A Figura 4.14 apresenta o gráfico *lagplot* para a série ITUB4.

4.1.8 Preço das Ações da Petrobras PN (PETR4)

O preço das ações da Petrobras PN (PETR4) é constituído por registros diários referentes ao valor de fechamento das ações da Petrobras PN no período de 13/11/2006 até 28/10/2010, que podem ser ilustradas na Figura 4.15.

A Tabela 4.8 apresenta algumas estatísticas básicas para a série PETR4.

A Figura 4.16 apresenta o gráfico *lagplot* para a série PETR4.

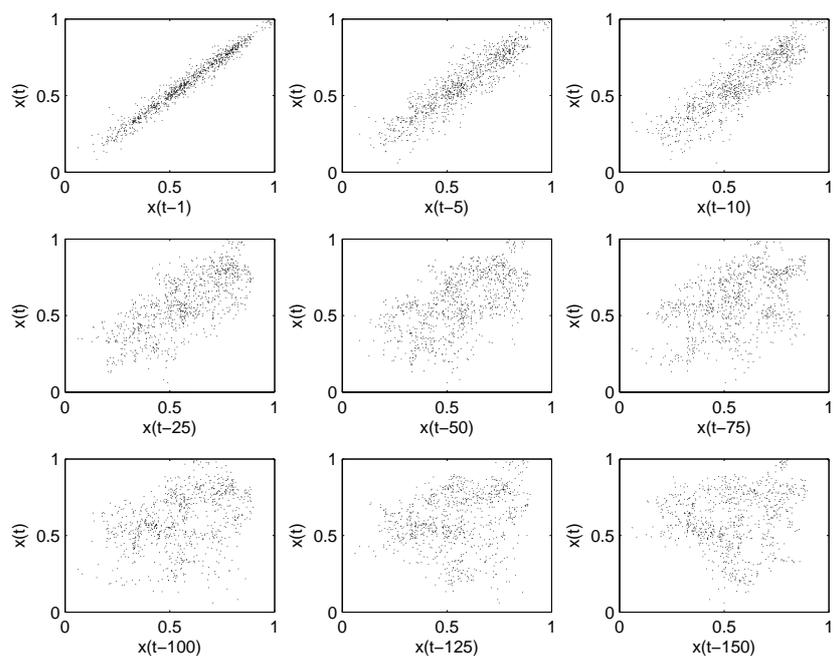


Figura 4.14 Lagplot da série ITUB4.

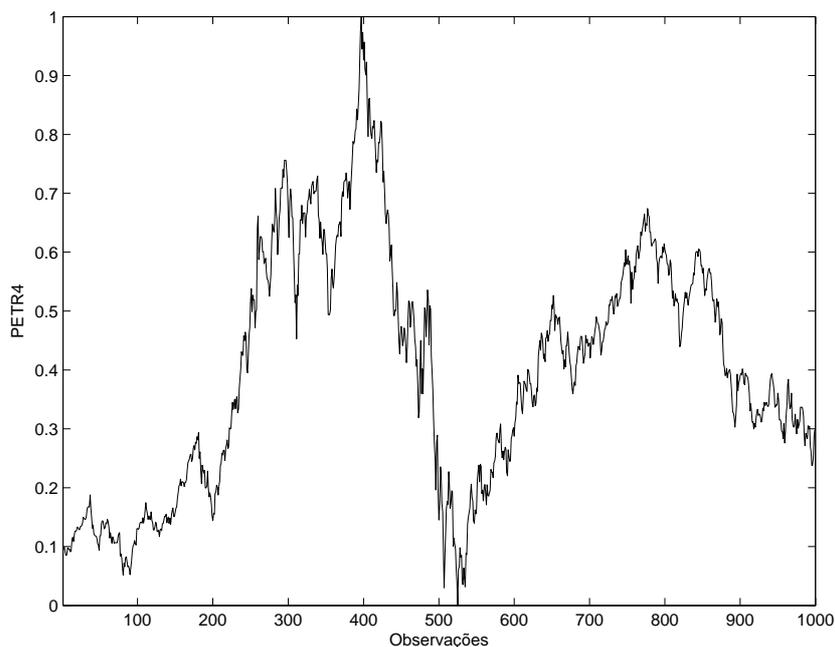


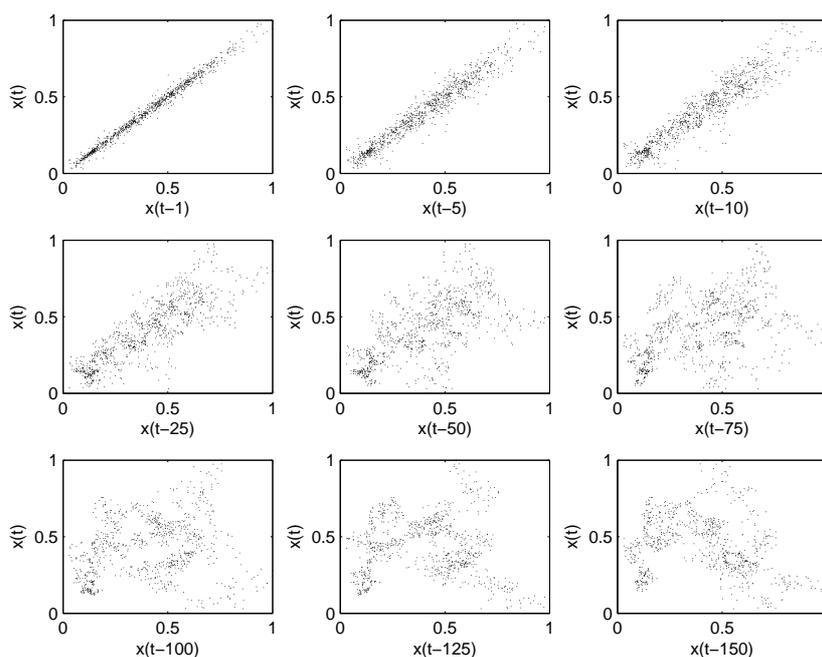
Figura 4.15 Gráfico da série PETR4.

4.1.9 Preço das Ações da Usiminas PNA (USIM5)

O preço das ações da Usiminas PNA (USIM5) é constituído por registros diários referentes ao valor de fechamento das ações da Usiminas PNA no período de 17/04/2006 até 24/03/2010,

Tabela 4.8 Estatísticas da série PETR4.

Estatística	Valor
Quantidade de pontos	1000,00
Valor máximo	49,82
Valor mínimo	16,17
Média	29,57
Variância	47,46
Desvio padrão	6,89

**Figura 4.16** Lagplot da série PETR4.

totalizando 1000 observações, que podem ser ilustradas na Figura 4.17.

A Tabela 4.9 apresenta algumas estatísticas básicas para a série USIM5.

Tabela 4.9 Estatísticas da série USIM5.

Estatística	Valor
Quantidade de pontos	1000,00
Valor máximo	138,00
Valor mínimo	19,15
Média	70,26
Variância	875,54
Desvio padrão	29,59

A Figura 4.18 apresenta o gráfico *lagplot* para a série USIM5.

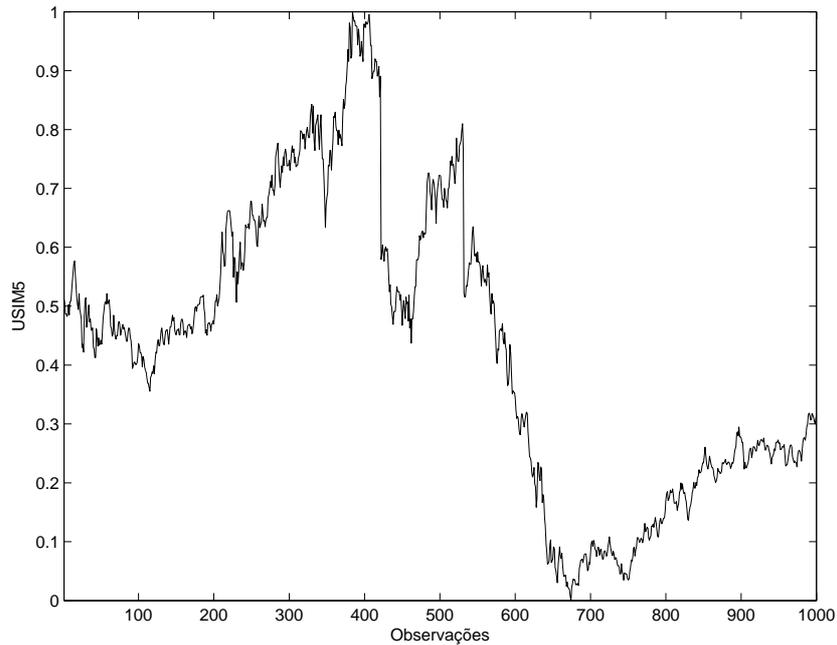


Figura 4.17 Gráfico da série USIM5.

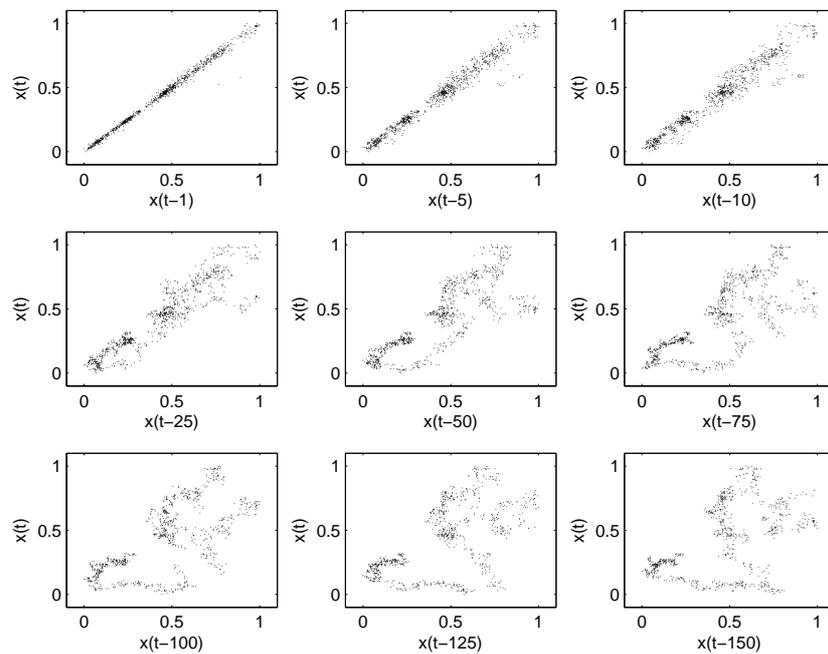


Figura 4.18 Lagplot da série USIM5.

4.1.10 Preço das Ações da Vale PNA (VALE5)

O preço das ações da Vale PN (VALE5) é constituído por registros diários referentes ao valor de fechamento das ações da Vale PN no período de 10/11/2006 até 28/10/2010, que podem ser

ilustradas na Figura 4.19.

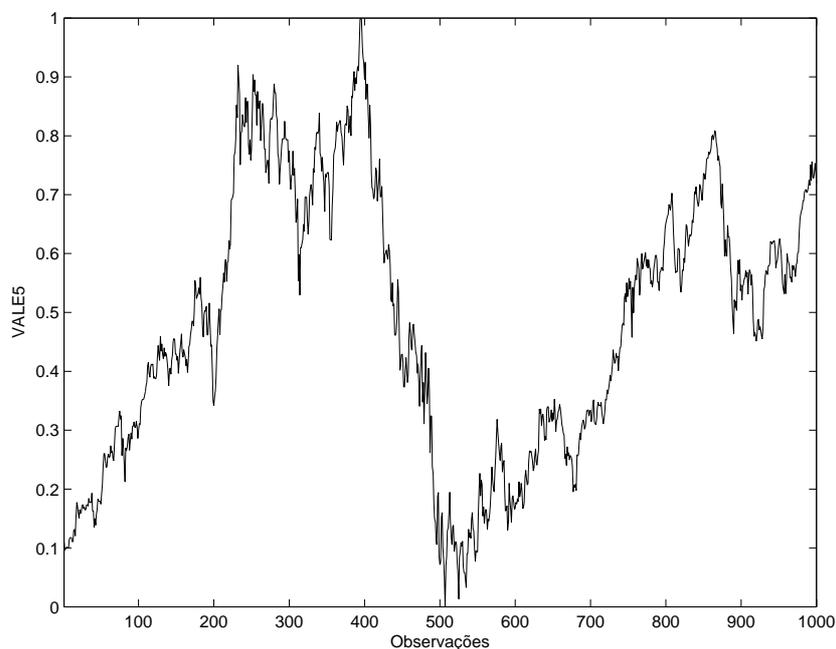


Figura 4.19 Gráfico da série VALE5.

A Tabela 4.10 apresenta algumas estatísticas básicas para a série VALE5.

Tabela 4.10 Estatísticas da série VALE5.

Estatística	Valor
Quantidade de pontos	1000,00
Valor máximo	58,70
Valor mínimo	20,24
Média	38,72
Variância	77,69
Desvio padrão	8,81

A Figura 4.20 apresenta o gráfico *lagplot* para a série VALE5.

4.1.11 Considerações

Vale mencionar a partir da análise das Figuras 4.2, 4.4, 4.6, 4.8, 4.10, 4.12, 4.14, 4.16, 4.18 e 4.20 (que ilustram os gráficos *lagplots* para todas as série estudadas neste trabalho), verificou-se a existência de um relacionamento linear dominante entre os retardos 1–10. Entretanto, ao aumentar a ordem dos retardos (entre 25–150), nota-se o aparecimento de uma estrutura não-linear complexa que caracteriza um relacionamento não-linear subdominante entre os retardos. Tal fato nos leva a crer que estas séries financeiras não são geradas por processos puramente aleatórios.

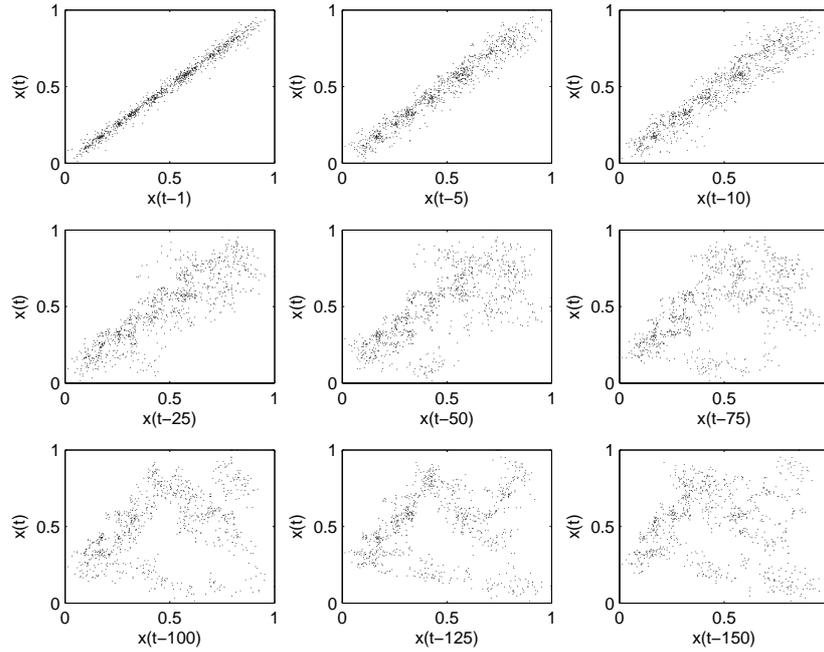


Figura 4.20 Lagplot da série VALES5.

4.2 Métricas para Desempenho de Previsão

Para o problema de previsão de séries temporais, a medida de desempenho natural é a própria previsão alcançada. Entretanto, não existe uma forma universal adotada pelos pesquisadores da área para avaliação da previsão. As métricas para desempenho são frequentemente definidas em termos do erro instantâneo de previsão, que é definido por

$$e_j = x_j - \hat{x}_j \quad (4.1)$$

onde x_j e \hat{x}_j representam, respectivamente, o valor real e previsto da série temporal no tempo j .

4.2.1 Erro Médio Quadrático

A principal e mais utilizada métrica para avaliação da previsão é o erro médio quadrático (*mean squared error*, MSE), dada por

$$\text{MSE} = \frac{1}{N} \sum_{j=1}^N (e_j)^2, \quad (4.2)$$

onde N é a quantidade de padrões, e e_j é o erro instantâneo para o padrão j .

A métrica MSE pode ser usada eficientemente para guiar o modelo de previsão em seu processo de aprendizagem. No entanto, esta não pode ser considerada, sozinha, como uma medida conclusiva para comparação entre diversos modelos de previsão [15]. Por esta razão, outras métricas para desempenho devem ser consideradas para permitir uma avaliação de desempenho mais robusta.

4.2.2 Erro Médio Percentual Absoluto

Dentre as métricas adotadas, o erro médio percentual absoluto (*mean absolute percentage error*, MAPE) é uma métrica que permite identificar precisamente os desvios do modelo, dada por

$$\text{MAPE} = \frac{1}{N} \sum_{j=1}^N \left| \frac{e_j}{x_j} \right|. \quad (4.3)$$

4.2.3 Estatística u de Theil

O modelo de passeio aleatório (*random walk model*, RWM) pode ser usado como preditor simples ($x_{t+1} = x_t$), sendo comumente aplicado a alguns problemas de previsão de séries temporais. Um forma de avaliar o modelo em relação a um RWM é utilizando a métrica estatística u de theil (*u of theil statistic*, THEIL), que associa o desempenho do modelo de previsão com um RWM, dada por

$$\text{THEIL} = \frac{\sum_{j=1}^{N-1} (e_j)^2}{\sum_{j=1}^{N-1} (x_j - x_{j+1})^2}, \quad (4.4)$$

onde, se $\text{THEIL} = 1$, implica que o modelo de previsão tem o mesmo desempenho de um RWM. Se $\text{THEIL} > 1$ implica que o modelo de previsão tem desempenho pior que um RWM, e se $\text{THEIL} < 1$ implica que o modelo de previsão tem um desempenho melhor que um RWM. Note que em um modelo de previsão perfeito, temos que $\text{THEIL} \rightarrow 0$.

4.2.4 Previsão de Mudança na Direção

Uma métrica interessante que mapeia o acerto na previsão (em termos percentuais) da direção futura da série temporal, ou mais especificamente, a capacidade do modelo de previsão prever se o valor futuro da série vai subir ou cair em relação ao valor passado da série. Esta métrica é conhecida como previsão de mudança na direção (*prediction of change in direction*, POCID), dada por

$$\text{POCID} = \frac{100}{N} \sum_{j=1}^N DP_j, \quad (4.5)$$

na qual

$$DP_j = \begin{cases} 1, & \text{se } (x_j - x_{j-1})(\hat{x}_j - x_{j-1}) > 0 \\ 0, & \text{caso contrário} \end{cases} \quad (4.6)$$

4.2.5 Variância Relativa Média

Esta métrica utilizada associa o desempenho do modelo de previsão com a média da série temporal, conhecida como variância relativa média (*average relative variance*, ARV), e dada

por

$$\text{ARV} = \frac{\sum_{j=1}^N (e_j)^2}{\sum_{j=1}^N (\hat{x}_j - \bar{x})^2}, \quad (4.7)$$

onde \bar{x} representa o valor médio da série temporal.

Note que se $\text{ARV} = 1$, implica que o modelo de previsão tem o mesmo desempenho da previsão da média da série. Se $\text{ARV} > 1$ implica que o modelo de previsão tem desempenho pior que a da previsão da média da série, e se $\text{ARV} < 1$ implica que o modelo de previsão tem um desempenho melhor que a da previsão da média da série. Pode-se verificar que em um modelo de previsão perfeito, temos que $\text{ARV} \rightarrow 0$.

4.2.6 Função de Avaliação

Na tentativa avaliar globalmente o desempenho preditivo do modelo, foi utilizada a função de avaliação (*evaluation function*, EF) proposta por [25–27, 29, 30, 38, 39], que representa uma combinação das métricas apresentadas nas seções anteriores, e sendo formalmente definida por

$$\text{EF} = \frac{\text{POCID}}{1 + \text{MSE} + \text{MAPE} + \text{THEIL} + \text{ARV}}. \quad (4.8)$$

Vale mencionar que a EF varia no intervalo [0, 100], onde quanto maior o valor de EF mais eficaz é o modelo de previsão.

4.3 Resumo do Capítulo

Neste capítulo foi descrito um conjunto de séries temporais financeiras com características complexas, tais como irregularidades, volatilidade, tendências e ruído, que será utilizado para realização dos experimentos com o modelo proposto. Também foi apresentado, a partir da análise dos *lagplots*, que séries temporais financeiras são geradas por componentes lineares dominantes combinadas com componentes não-lineares subdominantes, e tal fato indica que tais séries não são processos aleatórios e podem ser eficientemente previstas. Ao final, foram definidas cinco métricas e uma função de avaliação, as quais serão utilizadas para avaliar a qualidade das previsões alcançadas pelo modelo proposto neste trabalho.

O Modelo Proposto: *Dilation-Erosion-Linear Perceptron* (DELP)

Neste capítulo será apresentado um modelo de previsão, chamado de *perceptron* de dilatação-erosão-linear (*dilation-erosion-linear perceptron*, DELP), sendo baseado em conceitos da morfologia matemática sob o contexto de teoria de reticulados. Posteriormente, será apresentado um método de gradiente descendente para treinamento do modelo proposto utilizando ideias do algoritmo de retropropagação do erro, uma abordagem sistemática para superar o problema da não-diferenciabilidade das operações morfológicas de dilatação e erosão, e um procedimento de correção de fase para ajustar distorções de fase temporais que ocorrem na reconstrução do espaço de fase de fenômenos financeiros.

5.1 Introdução

A análise dos *lagplots* das séries temporais financeiras, apresentada no capítulo anterior, pôde revelar a presença de um relacionamento linear dominante bastante claro entre os retardos temporais de baixa ordem. Entretanto, com o aumento da ordem dos retardos temporais, pode-se identificar uma estrutura bastante complexa nos gráficos, que indica a presença de um relacionamento não-linear subdominante entre os retardos, o que leva a considerar que séries temporais financeiras não são geradas por processos aleatórios, uma vez que tais estruturas não-lineares subdominantes estão embutidas nas estruturas lineares dominantes, que garante, de acordo com [25–27, 29, 30, 38, 39, 49, 60, 61], que estas séries, diferentemente de séries geradas por processos aleatórios, podem ser descritas por $x_t = x_{t-1} + g(t) + r_t$, em que x_{t-1} representa a componente linear, $g(t)$ representa a componente não-linear, e r_t representa o termo de ruído gerado por uma distribuição Gaussiana com média zero e desvio padrão dado por σ ($\mathcal{N}(0, \sigma)$).

Vale mencionar que, de acordo com Araújo [25–27, 29, 30, 38, 39, 61], a magnitude da componente não-linear é bastante inferior à magnitude da componente linear. Assim, é de se esperar que um modelo de previsão acurado para este tipo particular de séries temporais seja capaz de estimar, de forma balanceada, ambas as componentes linear e não-linear, uma vez que o percentual esperado de utilização da componente linear é bastante superior ao percentual esperado de utilização da componente não-linear. Além disso, Araújo [26, 27, 29, 30] mostrou que a componente não-linear $g(t)$ destas séries temporais financeiras pode ser eficientemente estimada utilizando operadores crescentes (sendo esta característica fundamental dos operadores morfológicos de dilatação e erosão).

Portanto, este capítulo apresenta um modelo de previsão que atenda a todos estes quesitos. O modelo proposto consistirá de uma combinação linear entre operadores não-lineares

crescentes (dilatação e erosão morfológicas) e um operador linear do tipo resposta finita ao impulso, tendo a habilidade de determinar automaticamente os percentuais de utilização das suas componentes linear e não-linear. Nas seções a seguir serão apresentados os conceitos e teorias fundamentais para o entendimento do modelo proposto.

5.2 Fundamentos e Teorias Relevantes

A morfologia matemática (*mathematical morphology*, MM) [91–93, 132] representa uma teoria aplicada inicialmente para processamento e análise de imagens [76, 132, 133, 138] utilizando elementos estruturantes (*structuring elements*), uma vez que esta utiliza operadores e funções baseadas em conceitos fundamentais de álgebra, topologia e geometria integral [70, 95]. A teoria da MM foi inicialmente desenvolvida por Matheron e Serra [91–93, 132] sendo baseada nos trabalhos de Minkowski e Hadwiger [70, 95], e abrangendo áreas de pesquisa como filtragem, segmentação e reconstrução de imagens [42, 44, 53, 73, 83, 105, 129, 130, 157, 163], reconhecimento de padrões [80], reconhecimento automático de alvos [82], reconhecimento de caracteres manuscritos [104–106], detecção de minas terrestres [63], autolocalização [113, 114], reconhecimento de características [137], análise hiper-espectral de imagens [68, 112, 124], decomposição de sinais [11], e mais recentemente em estimativa de custo para desenvolvimento de software [31, 34, 45] e previsão de séries temporais [26, 27, 29, 30, 37–40, 43, 46, 151, 153].

Inicialmente, os operadores morfológicos desenvolvidos por Matheron e Serra [91–93, 132] foram projetados para imagens binárias (representadas matematicamente em termos de conjuntos). Entretanto, desde o seu desenvolvimento, a MM necessitava de uma teoria mais geral, com a inclusão de outros espaços de funções para descrição de imagens não-binárias. Para tal, baseado em fundamentos da umbra, Sternberg [141, 143] generalizou os operadores fundamentais da MM binária, para o caso de imagens não-binárias. Vale mencionar a existência de outras abordagens para generalização dos operadores morfológicos binários, baseados em operadores planos [76], e teoria de conjuntos nebulosos (*fuzzy set theory*) [3, 48, 51, 52, 152, 154]. Entretanto, uma observação feita por inicialmente por Serra [133] e posteriormente por Ronse, Banon e Heijmans [4, 76, 127] para a MM binária e não-binária, é que estas possuem como base a teoria de reticulados completos (*complete lattice theory*), uma vez que reticulados completos representam o contexto mais geral onde a MM pode ser conduzida.

De acordo com Ritter [119, 125, 126], a álgebra de imagens (*image algebra*) [142] representa uma teoria matemática que está relacionada com a transformação e análise de imagens, podendo ser utilizada para o desenvolvimento das bem conhecidas redes neurais artificiais (*artificial neural networks*, ANNs) [74, 75]. A partir desta observação, iniciou-se o desenvolvimento de um modelo particular de ANNs, referido como rede neural morfológica (*morphological neural network*, MNN) [21, 23, 24, 120], que podem ser descritas em termos de produtos matriciais definidos na álgebra minimax [17, 18]. A álgebra de imagens [142], no contexto matemático, representa uma álgebra do tipo heterogênea, que engloba ambas as álgebras linear e minimax como sub-álgebras [8, 22], bem como representa uma álgebra reticulados cuja origem é definida a partir de formulações matemáticas em problemas de pesquisa operacional [13, 17, 18, 65, 167].

Vale mencionar que, de acordo com Davidson [22], a MM binária e não-binária podem ser embutidas na álgebra minimax. Assim como a álgebra linear, a álgebra minimax está relacio-

onada com espaços matriciais e vetoriais associados aos seus produtos matriciais e vetoriais, respectivamente, cujas entradas são definidas em termos de um grupo ordenado de reticulado limitado [17, 18], que pode ser exemplificado pelo conjunto dos números reais estendidos $\mathbb{R}_{\pm\infty} = \mathbb{R} \cup \{+\infty, -\infty\}$ ou pelo conjunto dos números inteiros estendidos $\mathbb{Z}_{\pm\infty} = \mathbb{Z} \cup \{+\infty, -\infty\}$.

De forma geral, pode-se definir uma MNN como um tipo particular de ANN cujas operações elementares da MM entre reticulados completos são executadas em cada unidade de processamento da rede, uma vez que a teoria de reticulados completos provê um *framework* algébrico para a MM, sendo esta observação feita em estágios posteriores ao desenvolvimento da MM [4, 76, 127, 133]. Assumindo que a computação nas unidades de processamento de uma ANN pode ser expressa em um produto vetor-matriz na álgebra linear com entradas definidas em \mathbb{R} , tal computação em MNNs são definidas em termos de produtos vetor-matriz na álgebra minimax com entradas definidas em $\mathbb{R}_{\pm\infty}$ ou $\mathbb{Z}_{\pm\infty}$ (por exemplo, do reticulado completo $[0, 1]^n$ para o reticulado completo $[0, 1]$). Portanto, as MNNs podem ser, também, classificadas como redes neurais da álgebra minimax (*minimax algebra neural networks*, MANN) [120, 122, 147]. Vale mencionar que a utilização de uma função de ativação não é um passo obrigatoriamente necessário nas unidades de processamento em MNNs [52]. Também, vale mencionar que o termo “morfológicas” combinadas com o termo “redes neurais” no acrônimo MNNs que dizer que este tipo particular de ANN incorpora as ideias geométricas da MM, onde os elementos estruturantes são representados em termos dos pesos sinápticos da rede [148–150].

A seguir, serão apresentadas informações gerais sobre teoria de reticulados [7], álgebra minimax [17, 18], e MM no contexto de teoria de reticulados completos [76, 127, 133], que são conceitos importantes para a definição do modelo proposto nesta dissertação.

A teoria de reticulados (*lattice theory*) surgiu em 1987 a partir do trabalho de Dedekind [50]. Posteriormente, no trabalho de Birkhoff [7], foi mostrado que o conceito da teoria de reticulados está presente em toda a álgebra moderna. Diversos modelos de ANNs, em particular as MNNs, estão explicitamente ou implicitamente baseados na teoria de reticulados [80, 107, 118, 121, 123, 150], uma vez que a teoria de reticulados completos formam a base da MM [76, 127, 133]. Nesta dissertação serão apresentados apenas os conceitos necessários para o desenvolvimento do modelo proposto. Detalhes adicionais sobre a teoria de reticulados podem ser encontrados em [7, 50]. Para tal, serão recordados alguns conceitos fundamentais.

Definição 5.1. *Seja A um conjunto não vazio, e seja \leq uma relação binária em A . Então \leq é uma ordem parcial se e somente se as seguintes propriedades sejam válidas ($\forall a, b, c \in A$):*

1. $a \leq a$ (reflexividade);
2. $a \leq b$ e $b \leq a \Rightarrow a = b$ (anti-simetria);
3. $a \leq b$ e $b \leq c \Rightarrow a \leq c$ (transitividade)

Assim, pode-se denotar (A, \leq) como um conjunto parcialmente ordenado (*partially ordered set*). Por questões de simplicidade, assume-se que um conjunto parcialmente ordenado é um conjunto não vazio [69]. Adicionalmente, se $a \leq b$ ou $b \leq a \forall a, b \in A$, então A é dito ser um conjunto totalmente ordenado (*totally ordered set*) e é conhecido como cadeia (*chain*).

Dado um conjunto parcialmente ordenado (A, \leq) , e seja B um subconjunto de A ($B \subseteq A$). O elemento $a \in A$ é chamado de limitante superior de B se $b \leq a, \forall b \in B$. O menor limitante

superior de B , quando existir, é referido de supremo de B e denotado por $\bigvee B$. Mais precisamente, $a \in A$ é supremo de B se a é um limitante superior de B e, dado $c \in A$ outro limitante superior de B , tem-se que $a \leq c$. Denota-se o supremo de B por $\bigvee_{i \in I} b^i$ em vez de $\bigvee B$ quando $B = \{b^i, i \in I\}$ para um conjunto de índices I . Analogamente, o elemento a é chamado de limitante inferior de B se $a \leq b, \forall b \in B$. O maior limitante inferior de B , quando existir, é definido como ínfimo de B e denotado por $\bigwedge B$. Portanto, a é ínfimo de B se a é um limitante inferior de B e, dado c outro limitante inferior de B , tem-se que $c \leq a$. Denota-se o ínfimo de B por $\bigwedge_{i \in I} b^i$ em vez de $\bigwedge B$ quando $B = \{b^i, i \in I\}$.

Definição 5.2. Um conjunto parcialmente ordenado A é chamado de reticulado (lattice) se e somente se todo subconjunto finito de A admite um supremo e um ínfimo em A , isto é, $\forall B \subseteq A$ finito tem-se que $\bigvee B \in A$ e $\bigwedge B \in A$ [7].

Note que toda cadeia é um reticulado [7]. Se A_1, \dots, A_n são reticulados, então uma ordem parcial sobre $A^n = A_1 \times \dots \times A_n$ pode ser definida por

$$(a_1, \dots, a_n) \leq (b_1, \dots, b_n) \Leftrightarrow a_i \leq b_i, i = 1, \dots, n, \quad (5.1)$$

em que $a_i, b_i \in A_i$.

O conjunto parcialmente ordenado A^n também é um reticulado, sendo denotado por produto de reticulados (*product lattice*) com os constituintes A_1, \dots, A_n . O produto de reticulados corresponde ao produto de n cópias de A [148–150].

Definição 5.3. Sejam A e B reticulados. Um operador $\Psi : A \rightarrow B$ é chamado de crescente (increasing) se e somente se a seguinte condição seja verdadeira ($\forall a, b \in A$) [29]:

$$a \leq b \Rightarrow \Psi(a) \leq \Psi(b). \quad (5.2)$$

Definição 5.4. Similarmente, um operador $\Psi : A \rightarrow B$ é definido como decrescente (decreasing) se e somente se a seguinte condição seja verdadeira ($\forall a, b \in A$) [29]:

$$a \leq b \Rightarrow \Psi(b) \leq \Psi(a). \quad (5.3)$$

Definição 5.5. Sejam A e B reticulados e seja $\varphi : A \rightarrow B$ um operador. Assim, define-se um homomorfismo de reticulados (lattice homomorphism) se o operador φ satisfaz as seguintes condições ($a \in A$ e $b \in B$) [148–150]:

$$\varphi(a \vee b) = \varphi(a) \vee \varphi(b), \quad (5.4)$$

e

$$\varphi(a \wedge b) = \varphi(a) \wedge \varphi(b). \quad (5.5)$$

Um isomorfismo de reticulados (*lattice isomorphism*) pode ser visto como um homomorfismo bijetivo de reticulados (*bijjective lattice homomorphism*) [148–150]. De maneira similar, tem-se um isomorfismo de reticulados se o operador φ for bijetivo e isótono (*isotone*), isto é, que preserva a ordem ($\varphi(a) \leq \varphi(b), \forall a \leq b$). Note que quando $A = B$, tem-se um caso especial referido como autoisomorfismo (*lattice autoisomorphism*) [7].

Um reticulado A é dito limitado (*bounded lattice*) se e somente se tiver um menor elemento denotado por 0_A e um maior elemento denotado por 1_A [29]. Note que o produto de reticulados $A^n = A_1 \times \dots \times A_n$ é limitado se e somente se seus constituintes reticulados A_i , $i = 1, \dots, n$ forem limitados. Um reticulado A é dito condicionalmente completo (*conditionally complete lattice*) se e somente se cada subconjunto limitado não vazio tenha um supremo e ínfimo em A [29]. Um exemplo de um reticulado condicionalmente completo é o conjunto dos números reais \mathbb{R} .

Definição 5.6. Um reticulado A é dito completo (*complete lattice*) se e somente se todo subconjunto finito ou infinito, possui um supremo e um ínfimo em A [7], isto é, $\forall B \subseteq A$ finito ou infinito tem-se que $\bigvee B \in A$ e $\bigwedge B \in A$.

Note que todo reticulado completo é limitado [7]. Se A é um reticulado completo então o produto de reticulados completos é também completo porque para cada conjunto de índices J e $a^j \in A$ tem-se que [148–150]

$$\bigwedge_{j \in J} a^j = \bigwedge_{j \in J} (a_1^j, \dots, a_n^j) = (\bigwedge_{j \in J} a_1^j, \dots, \bigwedge_{j \in J} a_n^j), \quad (5.6)$$

e

$$\bigvee_{j \in J} a^j = \bigvee_{j \in J} (a_1^j, \dots, a_n^j) = (\bigvee_{j \in J} a_1^j, \dots, \bigvee_{j \in J} a_n^j). \quad (5.7)$$

O conjunto de números reais estendidos $\mathbb{R}_{\pm\infty}$ e o intervalo unitário $[0, 1]$ representam exemplos específicos de reticulados completos, que também são cadeias [7].

Definição 5.7. Um reticulado completo A é dito infinitamente distributivo (*infinitely distributive*) se e somente se $(\forall a, b_j \in A$ e $j \in J$ um conjunto de índices) [7]

$$a \wedge \left(\bigvee_{j \in J} b_j \right) = \bigvee_{j \in J} (a \wedge b_j) \quad (5.8)$$

e

$$a \vee \left(\bigwedge_{j \in J} b_j \right) = \bigwedge_{j \in J} (a \vee b_j) \quad (5.9)$$

Novamente, o conjunto de números reais estendidos $\mathbb{R}_{\pm\infty}$ e o intervalo unitário $[0, 1]$ representam exemplos específicos de reticulados completos infinitamente distributivos [7].

Definição 5.8. Sejam A e B reticulados completos, e sejam os operadores $\Phi : A \rightarrow B$ e $\Psi : A \rightarrow B$. O par (Φ, Ψ) é denominado uma adjunção entre A e B se $\Phi(b) \leq a \Leftrightarrow b \leq \Psi(a)$, em que $a \in A$ e $b \in B$ [7].

Pode-se observar que $\Phi(\bigwedge B) = \bigwedge \Phi(B)$ e $\Psi(\bigvee A) = \bigvee \Psi(A)$ [7].

Definição 5.9. Sejam A e B reticulados completos. Um operador ν é uma negação em A se é uma bijeção involutiva que reverte a ordem parcial em A . Além disso, seja o operador $\Phi : A \rightarrow B$ e seja ν_A e ν_B as negações em A e B , respectivamente, então o operador $\Phi^\nu : A \rightarrow B$ é dito a negação de Φ com relação a ν_A e ν_B e formalmente definido por [148–150]

$$\Phi^\nu = \nu_B \circ \Phi \circ \nu_A. \quad (5.10)$$

Seja A um reticulado completo, a função $\varphi : A \rightarrow A$ é definida como um automorfismo dual (*dual automorphism*) quando esta inverte a ordem parcial. Note que quando tem-se uma negação do reticulado completo A , tem-se uma bijeção involutiva (*involutive bijection*) que inverte a ordem parcial da relação \leq de A , e tal fato define um automorfismo dual involutivo (*involutive dual automorphism*) [148–150].

Por fim, a teoria de reticulados completos tem sido aceita como *framework* teórico apropriado para os operadores fundamentais da morfologia matemática [76, 127, 133], que serão formalmente definidos a seguir.

Definição 5.10. *Seja A e B reticulados completos, seja δ (dilatação) e ε (erosão) operadores do reticulado completo A para o reticulado completo B , e seja $X \subseteq A$. Então tem-se que [4, 76, 133]*

1. *O operador δ é dito ser uma dilatação algébrica se e somente se*

$$\delta(\bigvee X) = \bigvee_{x \in X} \delta(x). \quad (5.11)$$

2. *O operador ε é dito ser uma erosão algébrica se e somente se*

$$\varepsilon(\bigwedge X) = \bigwedge_{x \in X} \varepsilon(x). \quad (5.12)$$

Note que os operadores δ (dilatação) e ε (erosão) são crescentes [4]. Pode-se verificar que certas composições entre operadores morfológicos elementares também representam operadores elementares, como pode ser visto a seguir.

Lema 5.1. *Seja δ e ε os operadores morfológicos elementares de dilatação e erosão, respectivamente, de um reticulado completo A para um reticulado completo B , e dado o operador $f : A \rightarrow C$, com C sendo um reticulado completo. Então tem-se que*

1. *se f é definido como uma dilatação, então $f \circ \delta$ é uma dilatação.*
2. *se f é definido como uma erosão, então $f \circ \varepsilon$ é uma erosão.*

Uma questão central neste cenário é a decomposição de mapeamentos entre reticulados completos em termos de operadores elementares. Banon e Barrera [4] forneceram diversos teoremas sobre a decomposição de mapeamentos entre reticulados completos em termos de operações elementares da MM. Dentre estes teoremas, será abordada a decomposição de operadores crescentes, uma vez que de acordo com Araújo [26, 27, 29, 30] a componente não-linear presente em séries temporais financeiras têm comportamento crescente. A decomposição proposta por Banon e Barrera [4] para operadores morfológicos crescentes leva ao seguinte teorema:

Teorema 5.1. *Seja $\Psi : A \rightarrow B$ um mapeamento crescente entre os reticulados completos A e B . Então tem-se que*

1. existem dilatações δ^i para um conjunto de índices I de forma que

$$\Psi = \bigwedge_{i \in I} \delta^i. \quad (5.13)$$

2. existem erosões ε^i para um conjunto de índices I de forma que

$$\Psi = \bigvee_{i \in I} \varepsilon^i. \quad (5.14)$$

Este teorema proposto por Banon e Barrera [4] será utilizado como base para construção do modelo proposto, porque de acordo com Araújo [26, 27, 29, 30, 38, 39], o relacionamento não-linear das séries temporais financeiras pode ser estimado a partir de operadores crescentes. Portanto, um modelo baseado no Teorema 5.1 seria capaz de estimar, mais precisamente, o relacionamento não-linear destas séries devido ao seu comportamento inerentemente crescente. Entretanto, para o desenvolvimento do modelo proposto fundamentado na decomposição de Banon e Barrera, os operadores morfológicos elementares de dilatação e erosão requerem uma estrutura algébrica adicional além da estrutura de reticulados completos, onde tais operadores devem ser redefinidos em termos dos conceitos de supremo e ínfimo entre reticulados completos [29]. Esta estrutura algébrica adicional é conhecida como álgebra minimax [17], sendo formada a partir de um conjunto munido das operações de supremo, ínfimo e adição [8] (possuindo características semelhantes às da álgebra linear, tais como independência linear, autovalores e autovetores). Entretanto a abordagem utilizada aqui foi uma extensão para matrizes e vetores proposta por Cuninghame-Green [18]. A seguir serão apresentados alguns conceitos fundamentais para o entendimento desta abordagem.

Vale mencionar que devido ao fato do problema de séries temporais financeiras poder ser modelado em termos de operadores crescentes do tipo $\mathbb{R}_{\pm\infty}^n \rightarrow \mathbb{R}_{\pm\infty}$ (n representa a dimensionalidade dos retardos temporais para uma correta caracterização da série), o foco aqui será dado a cadeia $\mathbb{R}_{\pm\infty}$, que também é um grupo (*group*).

Definição 5.11. *Seja A um conjunto não vazio, e seja \leq uma relação de ordem parcial. Então A é definido como um grupo se satisfizer as seguintes condições:*

1. A for um reticulado.
2. $\forall a, b, x, y \in A$, tem-se que $a \leq b \Rightarrow xay \leq xby$.

Lembrando que \mathbb{R} , o conjunto dos elementos finitos de $\mathbb{R}_{\pm\infty}$, forma um grupo com a operação de adição convencional.

Definição 5.12. *Seja A um reticulado que também seja um grupo, e seja “+” uma operação de soma entre grupos. Então, o grupo A constitui um grupo ordenado de reticulado (lattice ordered group) se toda translação de grupo utilizando a operação entre grupos “+” for isótona, isto é, se satisfizer a seguinte condição ($\forall a, b \in A$ de forma que $a \leq b$) [7]:*

$$x + a + y \leq x + b + y, \quad \forall x, y \in A. \quad (5.15)$$

Também, de forma geral, um grupo é dito ser um grupo parcialmente ordenado (*partially ordered group*) se este constitui um conjunto parcialmente ordenado [7]. O único grupo parcialmente ordenado munido de limites universais é o grupo trivial $\{0\}$, onde 0 denota o elemento neutro em relação a operação de adição entre grupos. Portanto, o grupo trivial $\{0\}$ é também um grupo ordenado de reticulado que representa um reticulado limitado [148–150].

A álgebra minimax considera grupos ordenados de reticulado limitado (*bounded lattice ordered group*, BLOG) [17, 18]. De fato um BLOG é dado por um reticulado limitado A cujo conjunto de elementos finitos B formam um grupo, definido por $A \setminus \{+\infty, -\infty\}$ cujos termos $+\infty$ e $-\infty$ denotam $\bigvee A$ (o maior elemento de A) e $\bigwedge A$ (o menor elemento de A), respectivamente. No caso especial onde A é um reticulado completo cujo conjunto de elementos finitos formam um grupo, A é definido como um grupo ordenado de reticulado completo (*complete lattice ordered group*) [148–150]. Da mesma maneira, um reticulado condicionalmente completo pode formar um grupo, sendo então definido como grupo ordenado de reticulado condicionalmente completo (*conditionally complete lattice ordered group*) [148–150].

No entanto, é necessário definir a operação “+” em um grupo ordenado de reticulado limitado A (com o conjunto de elementos finitos B formando um grupo) sobre $(A \times A) \setminus (B \times B)$, uma vez que toda translação de grupo é representada como um automorfismo de reticulados e porque $-\infty \leq b \leq +\infty, \forall b \in B$. Portanto, tendo em vista manter a isotonicidade as seguintes operações são definidas [148–150]:

$$b + (+\infty) = +\infty + b = +\infty, \forall b \in B \cup +\infty, \quad (5.16)$$

e

$$b + (-\infty) = -\infty + b = -\infty, \forall b \in B \cup -\infty. \quad (5.17)$$

Fica claro que a isotonicidade das translações de grupo utilizando $+\infty$ não podem ser preservadas em A , mesmo se escolhermos definir, por exemplo, $(-\infty) + (+\infty) = (+\infty) + (-\infty)$. Para tal, existem algumas alternativas bastante similares [148–150]:

$$(-\infty) + (+\infty) = (+\infty) + (-\infty) = +\infty, \quad (5.18)$$

e

$$(-\infty) + (+\infty) = (+\infty) + (-\infty) = -\infty. \quad (5.19)$$

Desta forma, uma operação adicional “+’” necessita ser introduzida, que é bastante similar a operação “+” sobre $(A \times A) \setminus \{(+\infty, -\infty), (-\infty, +\infty)\}$. As principais diferenças entre elas são definidas pelas seguintes regras [148–150]:

$$(-\infty) + (+\infty) = (+\infty) + (-\infty) = -\infty, \quad (5.20)$$

e

$$(-\infty) +' (+\infty) = (+\infty) +' (-\infty) = +\infty. \quad (5.21)$$

Por fim, um grupo ordenado de reticulado limitado é definido por $(A, \vee, \wedge, +, +')$. Os grupos ordenados de reticulado completo $(\mathbb{R}_{\pm\infty}, \vee, \wedge, +, +')$ e $(\mathbb{Z}_{\pm\infty}, \vee, \wedge, +, +')$ que também são cadeias, são chamados de grupos ordenados totalmente completos (*complete totally ordered group*) [7]. Note que as operações $\vee, \wedge, +$ e $+'$ definidas em \mathbb{R} podem ser estendidas para

\mathbb{R}^n com $n > 1$, incluindo o maior elemento e o menor elemento em \mathbb{R}^n , dados por “ $+\infty$ ” e “ $-\infty$ ”, respectivamente, e definindo “ $+'$ ” e “ $+$ ” sobre $(\mathbb{R}^n \times \mathbb{R}^n)$, resultando em um grupo ordenado de reticulado completo $(\mathbb{R}^n \cup \{-\infty, +\infty\}, \vee, \wedge, +, +')$. Note que esta extensão não é uma cadeia [148–150]. Os grupos ordenados de reticulados limitados podem ser construídos a partir de uma estrutura algébrica do tipo anel chamada *belt*, mais precisamente *belts* de divisão (*division belts*), utilizando a noção de conjugação [148–150].

Definição 5.13. *Seja A um grupo ordenado de reticulado completo. O conjugado de um elemento $a \in A$, dado por a^* , é definido por [148–150]*

$$a^* = \begin{cases} -a, & \text{se } a \in A \setminus \{-\infty, +\infty\} \\ +\infty, & \text{se } a = -\infty \\ -\infty, & \text{se } a = +\infty \end{cases}, \quad (5.22)$$

em que $-a$, em um determinado grupo, representa o inverso de a .

A operação de conjugação origina uma negação v_* sobre A^n que mapeia o i -ésimo componente x_i de $X \in A^n$ ao seu conjugado [148–150]:

$$(v_*(X))_i = (x_i)^*, \quad \forall i = 1, 2, \dots, n. \quad (5.23)$$

Note que se A é um grupo ordenado de reticulado completo então A^n é também completo para todo n [148–150].

De acordo com [18, 146], os operadores fundamentais da morfologia matemática (dilatação e erosão) do tipo $A^n \rightarrow A^m$ podem ser definidos por produtos matriciais de máximo e mínimo da algebra minimax. A seguir serão definidos os produtos matriciais da algebra minimax [148–150].

Definição 5.14. *Seja A um grupo ordenado de reticulado completo. Seja a matriz $X \in A^{m \times p}$ e a matriz $Y \in A^{p \times n}$, então tem-se que o produto máximo (max product) de X e Y é dado por*

$$B = X \vee Y. \quad (5.24)$$

Um elemento da matriz B é dado por

$$b_{ij} = \bigvee_{k=1}^p (x_{ik} + y_{kj}). \quad (5.25)$$

Analogamente, o produto mínimo (min product) de X e Y é dado por

$$C = X \wedge Y. \quad (5.26)$$

Um elemento da matriz C é dado por

$$c_{ij} = \bigwedge_{k=1}^p (x_{ik} +' y_{kj}). \quad (5.27)$$

Definição 5.15. *Seja A um grupo ordenado de reticulado completo. Seja o operador $\delta_X : A^n \rightarrow A^m$ para $X, Y \in A^{n \times m}$. Então, tem-se que*

$$\delta_X(Y) = Y^T \vee X, \quad (5.28)$$

onde “ T ” representa a operação de transposição.

Analogamente, seja o operador $\varepsilon_X : A^n \rightarrow A^m$. Então, tem-se que

$$\varepsilon_X(Y) = Y^T \wedge X. \quad (5.29)$$

Os operadores δ_X e ε_X representam, respectivamente, uma dilatação e uma erosão de A^n para A^m .

Vale mencionar que o grupo ordenado de reticulado completo é um caso especial de uma estrutura algébrica denominada monóide duplo ordenado de reticulado completo (*complete lattice ordered double monoid*, CLODUM) [90]. Como uma imediata consequência do teorema de representação proposto por Maragos [90], tem-se que toda dilatação $\delta : A^n \rightarrow A^m$ (δ_X para $X \in A^{m \times n}$) é invariante a translação em relação a operação $+$. Similarmente, toda erosão $\varepsilon : A^n \rightarrow A^m$ (ε_X para $X \in A^{m \times n}$) é invariante a translação em relação a operação $+$.

Assim, o Teorema 5.1 para operadores crescentes revela que existem as matrizes A^i e B^j para um conjunto de índices I e J de forma que

$$\Psi = \bigwedge_{i \in I} \delta_{A^i}, \quad (5.30)$$

e

$$\Psi = \bigvee_{j \in J} \varepsilon_{B^j}. \quad (5.31)$$

As Equações (5.30) e (5.31) sugerem que um operador crescente $\Psi : \mathbb{R}^n \rightarrow \mathbb{R}$, pode ser aproximado em termos dos vetores $\mathbf{a}^i, \mathbf{b}^j \in \mathbb{R}^n$ para um conjunto finito de índices \bar{I} e \bar{J} :

$$\Psi \simeq \bigwedge_{i \in \bar{I}} \delta_{\mathbf{a}^i}, \quad (5.32)$$

e

$$\Psi \simeq \bigvee_{j \in \bar{J}} \varepsilon_{\mathbf{b}^j}. \quad (5.33)$$

No caso particular quando o conjunto de índices $\bar{I} = 1$ e $\bar{J} = 1$, um operador crescente $\Psi : \mathbb{R}^n \rightarrow \mathbb{R}$ pode então ser aproximado em termos dos vetores $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$:

$$\Psi \simeq \delta_{\mathbf{a}}, \quad (5.34)$$

e

$$\Psi \simeq \varepsilon_{\mathbf{b}}. \quad (5.35)$$

Implicitamente, tomando como base o estudo de Araújo [26, 27, 29, 30] para estimação de séries financeiras por operadores crescentes, assume-se que a componente não-linear $g(t)$ pode ser estimada em termos de operadores morfológicos de dilatação e erosão (do tipo crescentes). Portanto, as Equações (5.34) e (5.35) fornecem a base para aproximação da componente não-linear a partir deste tipo particular de operadores morfológicos crescentes.

5.3 Definição do DELP

O modelo proposto consiste de uma combinação linear entre operadores não-lineares (do tipo morfológicos via dilatação e erosão) e um operador linear (do tipo resposta finita ao impulso). A seguir será apresentada a definição formal do DELP.

Definição 5.16. *Seja $\mathbf{x} = (x_1, x_2, \dots, x_d) \in \mathbb{R}^d$ um sinal de entrada (retardos temporais) sobre uma i -ézima janela deslizante e seja y a saída (previsão) do DELP, que representa um sistema morfológico-linear híbrido com regra de transformação local do tipo $\mathbf{x} \rightarrow y$. Então, tem-se que o modelo DELP pode ser formalmente definido por*

$$y = \lambda \alpha + (1 - \lambda) \beta, \quad \lambda \in [0, 1], \quad (5.36)$$

onde

$$\beta = \mathbf{x} \cdot \mathbf{p}^T = x_1 p_1 + x_2 p_2 + \dots + x_d p_d, \quad (5.37)$$

e

$$\alpha = \theta \varphi + (1 - \theta) \omega, \quad \theta \in [0, 1], \quad (5.38)$$

em que

$$\varphi = \delta_{\mathbf{a}}(\mathbf{x}) = \bigvee_{i=1}^d (x_i + a_i), \quad (5.39)$$

e

$$\omega = \varepsilon_{\mathbf{b}}(\mathbf{x}) = \bigwedge_{i=1}^d (x_i + b_i), \quad (5.40)$$

onde o termo d denota a dimensionalidade do sinal de entrada (\mathbf{x}), os termos $\lambda, \theta \in \mathbb{R}$, e os termos $\mathbf{a}, \mathbf{b}, \mathbf{p} \in \mathbb{R}^d$. O vetor \mathbf{p} representa os coeficientes (pesos) do operador linear. O termo β representa a saída do operador linear de resposta finita ao impulso. O termo α representa a combinação linear entre os operadores morfológicos de dilatação e erosão (o termo de combinação é dado por θ). Os termos φ e ω representam a saída dos operadores morfológicos de dilatação e erosão, respectivamente. Os vetores \mathbf{a} e \mathbf{b} representam os elementos estruturantes (pesos) dos operadores de dilatação ($\delta_{\mathbf{a}}(\mathbf{x})$) e erosão ($\varepsilon_{\mathbf{b}}(\mathbf{x})$) empregados no módulo não-linear do DELP. Note que a saída y do DELP é dada por uma combinação linear entre o operador linear e uma combinação linear de operadores não-lineares crescentes (o termo de combinação da saída do DELP é dado por λ). A Figura 5.1 descreve a arquitetura do DELP.

5.3.1 Fundamentos para o Processo de Aprendizagem

Definição 5.17. *A r -ézima função de posto (rank function) de um vetor $\mathbf{x} = (x_1, x_2, \dots, x_n)^T \in \mathbb{R}^n$ é dada pelo r -ézimo elemento do vetor \mathbf{x} ordenado de forma decrescente ($x_{(1)} \geq x_{(2)} \geq \dots \geq x_{(n)}$). Portanto, a r -ézima função de posto do vetor \mathbf{x} é definida por [106]*

$$\mathcal{R}_r(\mathbf{x}) = x_{(r)}, \quad r = 1, 2, \dots, n. \quad (5.41)$$

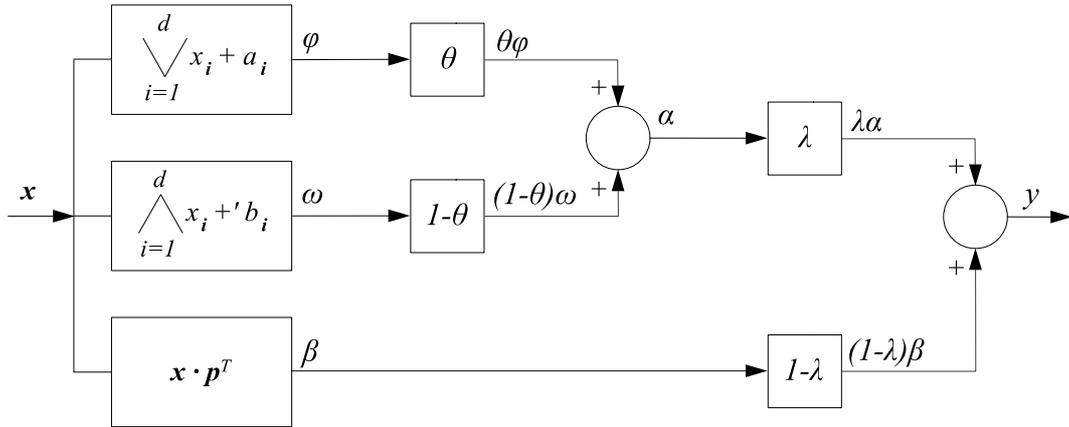


Figura 5.1 Arquitetura do DELP.

Definição 5.18. A função de impulso unitário (unit impulse function), $q(x)$, é definida por [106]

$$q(x) = \begin{cases} 1, & \text{se } x = 0, \\ 0, & \text{caso contrário.} \end{cases} \quad (5.42)$$

na qual $x \in \mathbb{R}$.

Quando se aplica a função de impulso unitário a um vetor \mathbf{x} , esta retorna um vetor de funções de impulso unitário ($Q(\mathbf{x})$), dada por [106]

$$Q(\mathbf{x}) = [q(x_1), q(x_2), \dots, q(x_n)]. \quad (5.43)$$

Definição 5.19. O r -ésimo vetor indicador de posto (rank indicator vector) \mathbf{c} de um vetor \mathbf{x} é definido por [106]

$$\mathbf{c}(\mathbf{x}, r) = \frac{Q(z \cdot \mathbf{1} - \mathbf{x})}{Q(z \cdot \mathbf{1} - \mathbf{x}) \cdot \mathbf{1}^T}, \quad (5.44)$$

na qual $z = \mathcal{R}_r(\mathbf{x})$ e $\mathbf{1} = (1, 1, \dots, 1)$.

Note que o vetor indicador de posto marca as posições dos elementos do vetor \mathbf{x} onde o valor de z ocorre. Portanto, este obrigatoriamente deve ter uma área unitária [106].

Proposição 5.1. Sejam $\mathbf{x} \in \mathbb{R}^n$, $r \in \{1, 2, \dots, n\}$ e $\mathbf{c} = \mathbf{c}(\mathbf{x}, r)$. Então, tem-se que [106]

1. $\mathbf{c} \cdot \mathbf{1}^T = 1$ (área unitária)
2. $\mathbf{c} \cdot \mathbf{x}^T = \mathcal{R}_r(\mathbf{x})$ (representação via produto interno)
3. $\mathbf{c} \cdot (z\mathbf{1} - \mathbf{x})^T = 0$ onde $z = \mathcal{R}_r(\mathbf{x})$
4. Se r é fixado, então \mathbf{c} é uma função constante por partes do vetor \mathbf{x} com exatamente $2^n - 1$ valores distintos possíveis. Além disso, para todos os pontos $\mathbf{x}_0 \in \mathbb{R}^n$ dotados com

componentes desiguais, $x_{0,i} \neq x_{0,j} \forall i \neq j$, é possível verificar uma vizinhança em torno delas, de modo que

$$\|\mathbf{x} - \mathbf{x}_0\|_\infty < \frac{1}{2} \min_{i \neq j} |x_{0,i} - x_{0,j}|, \quad (5.45)$$

onde o termo \mathbf{c} é constante e, dado por $\mathbf{c}(\mathbf{x}, r) = \mathbf{c}(\mathbf{x}_0, r)$.

Proposição 5.2. Seja $\mathbf{c} = \mathbf{c}(\mathbf{x}, r)$. Para um dado valor de r fixo, se \mathbf{c} é constante em uma vizinhança de algum \mathbf{x}_0 , então a r -ézima função de posto $\mathcal{R}_r(\mathbf{x})$ é diferenciável em \mathbf{x}_0 [106]:

$$\left. \frac{\partial \mathcal{R}_r(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}_0} = \left. \frac{\partial}{\partial \mathbf{x}} (\mathbf{c}(\mathbf{x}_0, r) \cdot \mathbf{x}^T) \right|_{\mathbf{x}=\mathbf{x}_0} = \mathbf{c}(\mathbf{x}_0, r). \quad (5.46)$$

Caso contrário, a função de posto $\mathcal{R}_r(\mathbf{x})$ não é diferenciável [106]. Neste caso, é possível torná-la diferenciável atribuindo ao vetor \mathbf{c} um valor unilateral da descontinuidade $\frac{\partial z}{\partial \mathbf{x}}$. Portanto, o vetor indicador de posto pode ser utilizado para estimar os gradientes baseado na descontinuidade da função de impulso unitário. Os fundamentos teóricos que suportam esta afirmação podem ser encontrados em [106].

Como o modelo proposto neste trabalho utiliza os operadores morfológicos de dilatação e erosão, a principal questão é como estimar seus gradientes nos pontos de não-diferenciabilidade. Uma maneira simples de solucionar este problema é utilizar uma extensão do vetor indicador de posto para os operadores de dilatação e erosão. Foi possível verificar que estes operadores podem ser vistos como casos particulares da função de posto ($\mathcal{R}_r(\mathbf{x} + \mathbf{a}) = \vee \mathbf{x} + \mathbf{a}$ quando $r = 1$ (operador de dilatação), e $\mathcal{R}_r(\mathbf{x} + \mathbf{b}) = \wedge \mathbf{x} + \mathbf{b}$ quando $r = n$ (operador de erosão), com $\mathbf{x}, \mathbf{a}, \mathbf{b} \in \mathbb{R}^n$). Entretanto, utilizar apenas a extensão do vetor indicador de posto para estimar aos gradientes dos operadores morfológicos de dilatação e erosão pode levar a mudanças abruptas na estimativa do gradiente e conseqüentemente comprometendo sua robustez numérica.

Uma forma simples de superar este problema é substituir a função de impulso unitário por uma função de impulso suave (*smoothed impulse function*), $q_\sigma(x)$, que depende de um fator de escala $\sigma \geq 0$. A opção de utilizar a função de impulso suave leva à aproximação da função de posto (e conseqüentemente os operadores morfológicos de dilatação de erosão) em termos de funções diferenciáveis para o cálculo de seus gradientes. A função de impulso suave deve ter pelo menos as seguintes propriedades:

$$\begin{aligned} q_\sigma(x) &= q_\sigma(-x) \text{ (simetria)} \\ q_\sigma(x) &\rightarrow q(x) \quad \forall x \text{ com } \sigma \rightarrow 0, \\ q_\sigma(x) &\rightarrow 1 \quad \forall x \text{ com } \sigma \rightarrow \infty. \end{aligned} \quad (5.47)$$

A função $\exp[-\frac{1}{2}(\frac{x}{\sigma})^2]$ é uma boa escolha para $q_\sigma(x)$. Da mesma forma que a função de impulso unitário de um vetor $Q(\mathbf{x})$, a função de impulso unitário suave (*smoothed unit impulse function*) de um vetor é dada por

$$Q_\sigma(\mathbf{x}) = [q_\sigma(x_1), q_\sigma(x_2), \dots, q_\sigma(x_n)]. \quad (5.48)$$

Definição 5.20. Se $q_\sigma(x)$ é uma função de impulso suave que satisfaz as condições impostas na Equação (5.47), então a r -ézima função de posto suave é definida por

$$\mathcal{R}_{r,\sigma}(\mathbf{x}) = \mathbf{c}_\sigma \cdot \mathbf{x}^T, \quad (5.49)$$

onde

$$\mathbf{c}_\sigma(\mathbf{x}, r) = \frac{Q_\sigma(z \cdot \mathbf{1} - \mathbf{x})}{Q_\sigma(z \cdot \mathbf{1} - \mathbf{x}) \cdot \mathbf{1}^T}, \quad z = \mathcal{R}_r(\mathbf{x}), \quad (5.50)$$

em que \mathbf{c}_σ representa o vetor indicador de posto suave (*smoothed rank indicator vector*).

Desta forma, a função de posto suave \mathbf{c}_σ representa uma aproximação da função de posto \mathbf{c} . A seguir serão apresentadas algumas propriedades da função de posto suave e do vetor indicador de posto suave.

Proposição 5.3. Sejam $\mathbf{x} \in \mathbb{R}^n$, $r \in \{1, 2, \dots, n\}$, $\mathbf{c}_\sigma = \mathbf{c}_\sigma(\mathbf{x}, r)$, $z = \mathcal{R}_r(\mathbf{x})$ e $z_\sigma = \mathcal{R}_{r,\sigma}(\mathbf{x})$. Então, tem-se que

1. $\mathbf{c}_\sigma \cdot \mathbf{1}^T = 1$ (área unitária);
2. $\lim_{\sigma \rightarrow 0} \mathbf{c}_\sigma = \mathbf{c}$;
3. $\lim_{\sigma \rightarrow 0} z_\sigma = z$;
4. $\lim_{\sigma \rightarrow \infty} \mathbf{c}_\sigma = \frac{1}{n}$;
5. $\lim_{\sigma \rightarrow \infty} z_\sigma = \frac{1}{n} \sum_{j=1}^n t_j$.

5.3.2 Processo de Aprendizagem com Ajuste Automático de Fase Temporal

A partir da definição do modelo DELP, é possível verificar que este requer o ajuste dos parâmetros $\mathbf{a}, \mathbf{b}, \mathbf{p} \in \mathbb{R}^d$ e $\lambda, \theta \in \mathbb{R}$. Portanto, o vetor de pesos \mathbf{w} (note que $\mathbf{w} \in \mathbb{R}^{3d+2}$) do modelo DELP é definido por

$$\mathbf{w} = (\mathbf{a}, \mathbf{b}, \mathbf{p}, \lambda, \theta). \quad (5.51)$$

Durante o processo de treinamento proposto, os pesos do modelo DELP são iterativamente ajustados de acordo com um critério de erro até que a convergência seja alcançada. Para isto, é necessário definir uma função objetivo em termos dos pesos $J(\mathbf{w})$ a ser minimizada durante o processo de treinamento (representada pelo erro de previsão efetuado pelo modelo DELP utilizando o vetor de pesos \mathbf{w}), sendo definida por

$$J(\mathbf{w}) = \sum_{m=1}^M e^2(m), \quad (5.52)$$

em que M representa a quantidade de padrões apresentados ao modelo no processo de treinamento e $e(m)$ representa o erro instantâneo do modelo para o m -ésimo padrão de treinamento, sendo definido por

$$e(m) = t(m) - y(m), \quad (5.53)$$

onde $t(m)$ e $y(m)$ representam a saída desejada e a saída do modelo, respectivamente.

Note que a função objetivo gera uma superfície de erro que reside no espaço \mathbb{R}^{3d+2} . Portanto o principal problema para minimização de $J(\mathbf{w})$ é encontrar um ponto de ótimo neste espaço que minimiza o erro entre a saída produzida pelo modelo DELP e a saída desejada, isto é, determinar \mathbf{w} que obtenha $\arg \min[J(\mathbf{w})]$. Assim, pode-se verificar que o processo de treinamento supervisionado do modelo DELP será equivalente a um problema de otimização não-linear irrestrito, em que a superfície de erro será minimizada a partir do ajuste iterativo do vetor de pesos do modelo. Nesta dissertação propõe-se um método de gradiente descendente utilizando ideias do algoritmo de retropropagação do erro (*back propagation*, BP) [74,75], que é empregado para se obter o vetor gradiente para ajuste do vetor de pesos do modelo (note que cada elemento do vetor gradiente, que está associado ao seu respectivo elemento do vetor de pesos do modelo DELP, indica o quanto a saída do modelo é influenciada por uma variação incremental deste parâmetro).

O processo treinamento do modelo DELP atualiza o vetor de pesos \mathbf{w} baseado no método do gradiente descendente. O ajuste de \mathbf{w} para o m -ésimo padrão de treinamento é realizado de acordo com a fórmula iterativa a seguir:

$$\mathbf{w}(i+1) = \mathbf{w}(i) - \mu \nabla J(\mathbf{w}), \quad (5.54)$$

onde $i \in \{1, 2, \dots\}$ e o termo μ representa o tamanho do passo de aprendizagem ou taxa de aprendizagem, que é responsável por regular o equilíbrio entre a estabilidade e velocidade de convergência do processo de treinamento do modelo, partindo de uma condição inicial $\mathbf{w}(0)$, que é iterativamente atualizada pela Equação (5.54) utilizando informações de direção local de primeira ordem. O termo $\nabla J(\mathbf{w})$ é dado pelo gradiente de J em relação a \mathbf{w} nos pontos onde o gradiente existe. Neste caso, tem-se que:

$$\nabla J(\mathbf{w}) = \frac{\partial J}{\partial \mathbf{w}} = \begin{pmatrix} \frac{\partial J}{\partial \mathbf{a}} \\ \frac{\partial J}{\partial \mathbf{b}} \\ \frac{\partial J}{\partial \mathbf{p}} \\ \frac{\partial J}{\partial \lambda} \\ \frac{\partial J}{\partial \theta} \end{pmatrix}, \quad (5.55)$$

em que

$$\frac{\partial J}{\partial \mathbf{a}} = -2e(m) \frac{\partial y}{\partial \mathbf{a}}, \quad (5.56)$$

$$\frac{\partial J}{\partial \mathbf{b}} = -2e(m) \frac{\partial y}{\partial \mathbf{b}}, \quad (5.57)$$

$$\frac{\partial J}{\partial \mathbf{p}} = -2e(m) \frac{\partial y}{\partial \mathbf{p}}, \quad (5.58)$$

$$\frac{\partial J}{\partial \lambda} = -2e(m) \frac{\partial y}{\partial \lambda}, \quad (5.59)$$

$$\frac{\partial J}{\partial \theta} = -2e(m) \frac{\partial y}{\partial \theta}. \quad (5.60)$$

Note que a existência do gradiente de J em relação a \mathbf{w} depende da existência dos gradientes $\frac{\partial y}{\partial \mathbf{a}}$, $\frac{\partial y}{\partial \mathbf{b}}$, $\frac{\partial y}{\partial \mathbf{p}}$, $\frac{\partial y}{\partial \lambda}$ e $\frac{\partial y}{\partial \theta}$. A seguir serão apresentadas as fórmulas para o cálculo dos gradientes.

O termo $\frac{\partial y}{\partial \lambda}$ é calculado por

$$\frac{\partial y}{\partial \lambda} = \alpha - \beta. \quad (5.61)$$

O termo $\frac{\partial y}{\partial \mathbf{p}}$ é calculado por

$$\frac{\partial y}{\partial \mathbf{p}} = \frac{\partial y}{\partial \beta} \frac{\partial \beta}{\partial \mathbf{p}}, \quad (5.62)$$

em que

$$\frac{\partial y}{\partial \beta} = 1 - \lambda, \quad (5.63)$$

e

$$\frac{\partial \beta}{\partial \mathbf{p}} = \mathbf{x}, \quad (5.64)$$

onde \mathbf{x} representa o sinal de entrada do modelo (m -ésimo padrão de treinamento).

O termo $\frac{\partial y}{\partial \theta}$ é calculado por

$$\frac{\partial y}{\partial \theta} = \frac{\partial y}{\partial \alpha} \frac{\partial \alpha}{\partial \theta}, \quad (5.65)$$

em que

$$\frac{\partial y}{\partial \alpha} = \lambda, \quad (5.66)$$

e

$$\frac{\partial \alpha}{\partial \theta} = \varphi - \omega. \quad (5.67)$$

Os termos $\frac{\partial y}{\partial \mathbf{a}}$ e $\frac{\partial y}{\partial \mathbf{b}}$ são estimados utilizando o conceito de vetor indicador de posto suave apresentado na Seção 5.3.1, devido a não-diferenciabilidade dos operadores morfológicos de dilatação e erosão empregados no modelo proposto, onde escolheu-se a função de impulso suave $Q_\sigma(\mathbf{x}) = [q_\sigma(x_1), q_\sigma(x_2), \dots, q_\sigma(x_d)]$, porque esta satisfaz as propriedades definidas na Equação (5.47), em que

$$q_\sigma(x_i) = \exp \left[\frac{1}{2} \left(\frac{x_i}{\sigma} \right)^2 \right], \quad \forall i = 1, \dots, d. \quad (5.68)$$

Note que embora a escolha do fator de escala σ afete diretamente a estimação e interpolação dos gradientes $\frac{\partial y}{\partial \mathbf{a}}$ e $\frac{\partial y}{\partial \mathbf{b}}$, o processo de treinamento do modelo DELP funciona mesmo com $\sigma \rightarrow 0$, uma vez que, neste caso particular, o gradiente será estimado em termos do vetor indicador de posto usual.

Assim, o termo $\frac{\partial y}{\partial \mathbf{a}}$ pode ser calculado por:

$$\frac{\partial y}{\partial \mathbf{a}} = \frac{\partial y}{\partial \alpha} \frac{\partial \alpha}{\partial \varphi} \frac{\partial \varphi}{\partial \mathbf{a}} = \lambda \frac{\partial \alpha}{\partial \varphi} \frac{\partial \varphi}{\partial \mathbf{a}}, \quad (5.69)$$

em que

$$\frac{\partial \alpha}{\partial \varphi} = \theta, \quad (5.70)$$

e

$$\frac{\partial \varphi}{\partial \mathbf{a}} = \frac{Q_{\sigma}(\varphi \cdot \mathbf{1} - (\mathbf{x} + \mathbf{a}))}{Q_{\sigma}(\varphi \cdot \mathbf{1} - (\mathbf{x} + \mathbf{a})) \cdot \mathbf{1}^T}. \quad (5.71)$$

Também, o termo $\frac{\partial y}{\partial \mathbf{b}}$ pode ser calculado por:

$$\frac{\partial y}{\partial \mathbf{b}} = \frac{\partial y}{\partial \alpha} \frac{\partial \alpha}{\partial \omega} \frac{\partial \omega}{\partial \mathbf{b}} = \lambda \frac{\partial \alpha}{\partial \omega} \frac{\partial \omega}{\partial \mathbf{b}}, \quad (5.72)$$

em que

$$\frac{\partial \alpha}{\partial \omega} = 1 - \theta, \quad (5.73)$$

e

$$\frac{\partial \omega}{\partial \mathbf{b}} = \frac{Q_{\sigma}(\omega \cdot \mathbf{1} - (\mathbf{x} + \mathbf{b}))}{Q_{\sigma}(\omega \cdot \mathbf{1} - (\mathbf{x} + \mathbf{b})) \cdot \mathbf{1}^T}. \quad (5.74)$$

Finalmente, foi incluído um procedimento de correção de fase [25–27, 29, 30, 38, 39, 60, 61] no processo de treinamento do modelo DELP na tentativa de ajustar automaticamente distorções de fase temporais que ocorrem em problemas de previsão de séries temporais financeiras. A Figura 5.2 ilustra o procedimento de correção de fase.

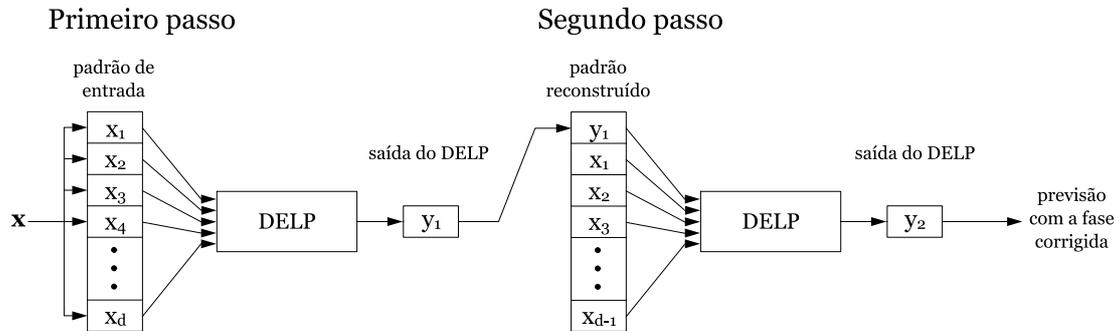


Figura 5.2 Procedimento de correção de fase.

De acordo com a Figura 5.2, no primeiro passo, o padrão de entrada $\mathbf{x} = (x_1, \dots, x_d)$ é apresentado ao modelo DELP, que produz a saída y_1 . Posteriormente, o valor de y_1 é incluído no padrão de entrada dimensionalmente reduzido $(x_1, \dots, x_{d-1}) \in \mathbb{R}^{d-1}$, que gera o padrão reconstruído $(y_1, x_1, \dots, x_{d-1})$. Este novo padrão é apresentado ao mesmo modelo DELP, produzindo a previsão com a fase corrigida y_2 . A Figura 5.3 ilustra os passos do algoritmo de treinamento do modelo DEP incluindo o procedimento de correção de fase.

Vale mencionar que três condições de parada são utilizados para no algoritmo de treinamento do modelo DELP:

```

Algoritmo de Treinamento do DELP() {
  inicializar parâmetros ( $\mu$ ,  $\sigma$ ,  $\mathbf{w}$  e a condição de parada);
   $\tau = 0$ ;          //  $\tau$ : época de treinamento atual
  enquanto ( não condição de parada ) {
     $\tau = \tau + 1$ ;
    para  $i=1$  ate  $M$  {
      calcular a saída  $y_1$  para ao  $m$ -ésimo padrão de treinamento;
      calcular o erro instantâneo ( $e(m)$ );
      calcular o gradiente ( $\nabla J(\mathbf{w})$ );
      utilizar a saída  $y_1$  para reconstrução do  $m$ -ésimo padrão de treinamento;
      calcular a saída  $y_2$  para o  $m$ -ésimo padrão de treinamento reconstruído;
      calcular a função objetivo ( $J(\mathbf{w})$ ) utilizando a saída  $y_2$ ;
      atualizar o vetor de pesos  $\mathbf{w}$ ;
    }
  }
}

```

Figura 5.3 Passos do algoritmo de treinamento do modelo proposto.

1. A quantidade máxima de épocas de treinamento;
2. A diminuição do erro de treinamento (*process training*, Pt) [110];
3. O aumento do erro de validação (*generalization loss*, Gl) [110].

5.4 Resumo do Capítulo

Neste capítulo foi apresentada uma classe particular de *perceptrons* híbridos para solucionar o problema de previsão de séries temporais financeiras. Inicialmente foram introduzidos os conceitos e teorias fundamentais necessários para o desenvolvimento do modelo proposto. Então, a partir do estudo das séries temporais financeiras apresentado no capítulo anterior, foi apresentada a proposta do modelo de previsão, chamado de *perceptron* de dilatação-erosão-linear (*dilation-erosion-linear perceptron*, DELP), que é baseado em conceitos da morfologia matemática sob o contexto de teoria de reticulados e consiste de uma combinação linear entre o operador linear de resposta finita ao impulso e uma combinação linear de operadores não-lineares morfológicos crescentes (dilatação e erosão). Posteriormente, foi apresentado um método de gradiente descendente para treinamento do modelo proposto utilizando ideias do algoritmo de retropropagação do erro, uma abordagem sistemática para superar o problema da não-diferenciabilidade das operações morfológicas de dilatação e erosão, e um procedimento de correção de fase para ajustar distorções de fase temporais que ocorrem na reconstrução do espaço de fase de fenômenos financeiros.

Simulações e Resultados Experimentais

Neste capítulo será apresentado o processo empregado na realização dos experimentos. Posteriormente, os resultados preditivos alcançados pelo modelo proposto serão analisados, discutidos e comparados com aqueles encontrados na literatura.

6.1 Metodologia

Para a realização dos experimentos foram utilizadas dez séries temporais financeiras apresentadas no Capítulo 4 (BVSP, DJI, IXIC, FTSE, BBDC4, GOLL4, ITUB4, PETR4, USIM5 e VALE5). Para que as séries temporais possam ser utilizadas pelos modelos estudados neste trabalho, é necessário que as mesmas passem por um processo de normalização ou escalonamento. Esta etapa é formalmente definida como pré-processamento [165]. A normalização tem como principal objetivo dar conformidade, em termos de domínio, entre os valores da série temporal e os valores possíveis do modelo de previsão.

Na literatura, existem diversos meios de realizar a normalização dos dados [165], sendo esta escolha dependente do tipo do problema em questão:

- Normalização linear para o intervalo $[0, 1]$:

$$xn_i = \frac{x_i - \min(\mathbf{x})}{\max(\mathbf{x}) - \min(\mathbf{x})}. \quad (6.1)$$

- Normalização linear para o intervalo $[A, B]$:

$$xn_i = (B - A) \frac{x_i - \min(\mathbf{x})}{\max(\mathbf{x}) - \min(\mathbf{x})} + A. \quad (6.2)$$

- Normalização estatística

$$xn_i = \frac{x_i - \text{mean}(\mathbf{x})}{\text{std}(\mathbf{x})} \quad (6.3)$$

- Normalização simples

$$xn_i = \frac{x_i}{\max(\mathbf{x})} \quad (6.4)$$

em que x_i , xn_i com $i = 1, 2, \dots, I$ sendo os valores reais e normalizados, respectivamente, da série temporal. Os termos $\min(\cdot)$, $\max(\cdot)$, $\text{mean}(\cdot)$ e $\text{std}(\cdot)$ representam as operações de mínimo, máximo, média e desvio padrão respectivamente, de um arranjo de elementos.

Nesta dissertação foi utilizada a normalização linear para o intervalo $[0, 1]$, dando a possibilidade de utilização de todo o domínio de atuação do DELP. Após a etapa de normalização, cada uma das séries foi dividida em três conjuntos de acordo com Prechelt [110] (trabalho que tenta padronizar a divisão e o percentual do conjunto de dados em problemas de classificação e previsão): i) conjunto de treinamento (utilizado no processo de aprendizagem do modelo de previsão), ii) conjunto de validação (utilizado para validar o processo de aprendizagem do modelo de previsão) e iii) conjunto de teste (utilizado para confirmar o desempenho prático do modelo de previsão). Para definição da cardinalidade de cada um destes conjuntos, foi utilizado o conjunto de regras também definido em [110], onde foi definido 50% dos dados para o conjunto de treinamento, 25% dos dados para o conjunto de validação e 25% dos dados para o conjunto de teste.

Tendo em vista comparar o desempenho preditivo do modelo proposto, foram utilizados modelos bem conhecidos e consolidados na literatura de previsão de séries temporais: i) modelo ARIMA [10], ii) modelo MLP [74, 75], iii) modelo híbrido MGA–MLP [60, 61], iv) modelo híbrido PSO–MLP [25], v) modelo híbrido otimizado TAEF [60, 61], e vi) modelo híbrido otimizado SHIF [25].

Para realização dos experimentos com o modelo ARIMA(p, q, d) foi utilizada a ferramenta *Matlab*, com a *toolbox* ARMASA, onde sua proposta básica é calcular os coeficientes dos operadores auto-regressivos e de médias móveis, otimizando-os a partir do desempenho de previsão alcançado. O termo de diferenciação utilizado foi $d = 1$ como sugerido por [10].

Para realização dos experimentos com o modelo MLP, foi necessário definir uma arquitetura básica para todos os experimentos, que consiste em uma MLP de três camadas (uma camada de entrada, uma camada intermediária e uma camada de saída), formalmente descrita utilizando a notação MLP $a-b-c$, onde a representa a quantidade de unidades de processamento na camada de entrada, b representa a quantidade de unidades de processamento na camada intermediária, e c representa a quantidade de unidades de processamento na camada de saída.

A quantidade de unidades de processamento na camada de entrada é definida pela quantidade de retardos temporais utilizados para a descrição da série. Para a definição dos retardos temporais foi utilizada uma metodologia empírica sugerida por [25, 60, 61], a partir dos quais foram escolhidos os valores 1, 3, 5 e 10. A quantidade de unidades de processamento na camada escondida foi determinada empiricamente através de uma série de experimentos, a partir dos quais foram escolhidos os valores 1, 5, 10, 25 e 50. A quantidade de unidades de processamento na camada de saída foi fixa em 1, uma vez que este trabalho foca apenas em previsões de um passo a frente, isto é, um horizonte de previsão unitário.

Em termos de arquitetura da rede neural MLP, foram investigadas três tipos distintos e formalmente definidas nas Equações (3.44)–(3.47). Para treinamento da rede MLP foi utilizado o algoritmo de Levenberg-Marquardt [71] como sugerido por [25, 61], utilizando os seguintes critérios de parada [110]: i) A quantidade máxima de épocas de treinamento (10^4), ii) O aumento no erro de validação ou *generalization loss* ($Gl > 5\%$), e iii) A queda no erro de treinamento ou *process training* ($Pt \leq 10^{-6}$).

Todos os experimentos com o modelo MLP foram desenvolvidos e implementados utilizando a ferramenta *Matlab*, com a *toolbox* *Neural Network*. Para cada arquitetura utilizada nos experimentos foram realizadas dez execuções distintas, tendo em vista se obter um com-

portamento médio do modelo MLP. Vale mencionar que estes experimentos foram realizados visando a obtenção de um modelo MLP com o melhor desempenho possível, de forma a garantir uma comparação justa entre os modelos estudados nesta dissertação.

Para realização dos experimentos com o modelo híbrido MGA–MLP, foram fixados os seguintes parâmetros como sugerido por [60, 61]: quantidade máxima de gerações do MGA (10000), probabilidade de mutação (0, 1), peso do operador de cruzamento (0, 9) e o tamanho da população é fixa com dez indivíduos, onde cada indivíduo possui uma arquitetura de rede MLP máxima (10-10-1), que denota um máximo de dez unidades de processamento na camada de entrada, um máximo de dez unidades de processamento na camada escondida e uma unidade na camada de saída (previsão de um passo).

Para realização dos experimentos com o modelo híbrido PSO–MLP, foram fixados os seguintes parâmetros como sugerido por [25]: quantidade máxima de iterações do PSO (10000), coeficientes de aceleração (2, 05), peso de inércia (0, 9), velocidade máxima (2, 048) e o enxame possuindo dez partículas, onde cada partículas possui uma arquitetura de rede MLP máxima (10-10-1), que denota um máximo de dez unidades de processamento na camada de entrada, um máximo de dez unidades de processamento na camada escondida e uma unidade na camada de saída (previsão de um passo).

Para realização dos experimentos com o modelo híbrido otimizado TAEF, foram fixados os parâmetros a seguir como sugerido por [60, 61]: valor da função de *fitness* mínima aceitável (0, 99), quantidade de iterações do método (10), quantidade máxima de retardos temporais (4), quantidade máxima de unidades de processamento na camada escondida da rede MLP (15). Note que como o método TAEF também utiliza o MGA no seu processo evolucionário, os parâmetros sugerido por [60, 61] são os mesmos utilizados no modelo híbrido MGA–MLP. Cada indivíduo do MGA representa uma rede MLP, que é adicionalmente treinada por um dos seguintes algoritmos: *resilient backpropagation* [117], *levenberg marquardt* [71], *scaled conjugate gradient* [97] e *one step secant conjugate gradient* [5], por um período máximo de 10^3 épocas de treinamento. Três condições de parada são utilizadas para parar o treinamento da rede MLP: i) quantidade épocas, ii) $Pt \leq 10^{-6}$, e iii) $Gl \geq 5\%$.

Para realização dos experimentos com o modelo híbrido otimizado SHIF, foram fixados os parâmetros a seguir como sugerido por [25]: valor da função de *fitness* mínima aceitável (40), quantidade de iterações do método (10), quantidade máxima de retardos temporais (10), quantidade máxima de unidades de processamento na camada escondida da rede MLP (20). Note que como o método SHIF também utiliza o PSO no seu processo evolucionário, os parâmetros sugerido por [25] são os mesmos utilizados no modelo híbrido PSO–MLP. Cada partícula do PSO representa uma rede MLP, que é adicionalmente treinada por um dos seguintes algoritmos: *resilient backpropagation* [117], *levenberg marquardt* [71], *scaled conjugate gradient* [97] e *one step secant conjugate gradient* [5], por um período máximo de 10^3 épocas de treinamento. Três condições de parada são utilizadas para parar o treinamento da rede MLP: i) quantidade máxima de épocas de treinamento (10^4), ii) $Pt \leq 10^{-6}$, e iii) $Gl \geq 5\%$.

Para realização dos experimentos com o modelo proposto, é necessário definir uma arquitetura básica, definida pela notação $DELPL(lags; \mu; \sigma)$, em que *lags* define os retardos temporais utilizados para representar o fenômeno temporal de interesse (caso sejam escolhidos os retardos 2, 5 e 7, temos que $lags=2,5,7$). Nesta dissertação foram utilizados os mesmos retardos

temporais para todos os experimentos (do retardo 2 até o retardo 151, isto é $lags=2-151$), uma vez que o intuito deste trabalho também é mostrar que o modelo aqui desenvolvido é capaz de prever eficientemente séries temporais financeiras independentemente dos retardos temporais utilizados para uma correta caracterização do fenômeno gerador da série. Note que a escolha destes retardos temporais se deu pela análise do *lagplot*, onde foi possível identificar que o relacionamento não-linear se apresenta melhor caracterizado (quando a estrutura aparece mais claramente no gráfico) quando a dimensionalidade dos retardos temporais é 150. Também, vale mencionar que o primeiro retardo temporal não é utilizado porque é necessário se criar uma estrutura de passeio aleatório para a utilização do procedimento de correção de fase utilizado no algoritmo de treinamento do DELP, uma vez que a etapa chave deste procedimento é a previsão de dois passos para o ajuste do atraso temporal das previsões. O valor de μ é determinado empiricamente através de uma série de experimentos, a partir dos quais foram escolhidas as quantidades 0,001, 0,01 e 0,1. O valor de σ também é determinado empiricamente através de uma série de experimentos, a partir dos quais foram eleitas as quantidades 0,005, 0,05 e 0,5. Os valores iniciais do vetor de pesos do modelo DELP são: $\mathbf{a}, \mathbf{b}, \mathbf{p} \in [-1, 1]$ e $\lambda, \theta \in [0, 1]$. No processo de treinamento do DELP três condições de parada são utilizadas: i) quantidade máxima de épocas de treinamento (10^4), ii) $Pt \leq 10^{-6}$, e iii) $Gl \geq 5\%$. Para cada arquitetura investigada, são realizados dez experimentos, onde são calculadas a média (MEAN) e o desvio padrão (STD) na tentativa de se obter um desempenho de previsão médio do modelo DELP. Além disso, são calculados todos os intervalos de confiança (CI) com a suposição de uma distribuição normal com 99% de grau de certeza.

Para o cálculo do ganho percentual, em termos de desempenho de previsão, do modelo proposto em relação aos outros modelos de investigados neste trabalho, foram utilizadas as seguintes equações

$$Ganho = 100 - 100 \frac{mmp}{mmi}, \quad (6.5)$$

$$Ganho = 100 \frac{mmp}{mmi} - 100, \quad (6.6)$$

em que *mmp* e *mmi* representam o valor da métrica encontrada para o modelo proposto e para o modelo investigado, respectivamente. Note que a Equação (6.5) deve ser utilizada para mensurar os ganhos obtidos para as métricas *MSE*, *THEIL*, *MAPE* e *ARV*, enquanto que a Equação (6.6) deve ser utilizada para mensurar os ganhos obtidos para as métricas *POCID* e *EF*.

Vale mencionar que o procedimento de correção de fase [25–27, 29, 30, 38, 39, 60, 61] foi aplicado a todos os modelos de previsão investigados neste trabalho de forma a garantir uma comparação justa entre eles. Vale mencionar que para todos os modelos investigados, aquele escolhido para representar uma determinada técnica será o que detiver o maior valor da métrica EF no conjunto de validação.

Nas seções subsequentes serão apresentados o desempenho preditivo de todos os modelos investigados neste trabalho.

6.2 Resultados

Nesta seção é apresentada uma análise comparativa, para cada uma das séries temporais investigadas, comentando o desempenho de todos os modelos de previsão estudados a partir das métricas formalmente definidas no Capítulo 4. Também, serão apresentados os gráficos preditivos tendo em vista uma análise visual dos resultados experimentais.

6.2.1 Índice da Bolsa de Valores de São Paulo (BVSP)

Para a série do Índice da Bolsa de Valores de São Paulo (BVSP), os modelos investigados com o melhor desempenho preditivo foram: i) modelo ARIMA com a configuração (2,1,1), ii) modelo MLP com a configuração 10-25-1 e arquitetura definida pela Equação (3.46), iii) o modelo MGA-MLP definiu os retardos temporais (1, 2, 4, 8, 9 e 10) e a rede MLP com a configuração 6-3-1 e arquitetura definida pela Equação (3.46), iv) o modelo PSO-MLP escolheu os retardos temporais (1, 2, 4, 5 e 7) e a rede MLP com a configuração 5-3-1 e arquitetura definida pela Equação (3.47), v) o modelo TAEF determinou os retardos temporais (3, 4, 7, 8 e 9) e a rede MLP com a configuração 5-9-1, arquitetura definida pela Equação (3.46) e o algoritmo *one step secant conjugate gradient* para treinamento da rede MLP, e vi) o modelo SHIF determinou os retardos temporais (2, 3, 4, 6, 7, 8 e 10) e a rede MLP com a configuração 7-2-1, arquitetura definida pela Equação (3.44) e o algoritmo *scaled conjugate gradient* para treinamento da rede MLP.

A Tabela 6.1 apresenta todos os experimentos realizados com o modelo DELP (utilizando o conjunto de treinamento) variando o valor de μ e σ , onde foram calculadas todas métricas, bem como as estatísticas MEAN, STD e CI para análise do desempenho de previsão médio do modelo proposto.

De acordo com a Tabela 6.1, pode-se verificar que os melhores valores de μ e σ para a série BVSP foram 0,1 e 0,5, respectivamente, uma vez que obtivemos o melhor desempenho de previsão em termos da função de avaliação global (EF). Portanto, assumindo essa suposição, escolheu-se o modelo DELP(0,1; 0,5) com a maior métrica EF no conjunto de validação, dentre os dez modelos criados neste experimento, para representar o modelo proposto. Neste modelo, o termo de combinação λ determinado foi 0,0031, significando que o DELP utilizou 99,69% da componente linear e 0,31% da componente não-linear, confirmando a hipótese (através da análise do *lagplot* apresentada no Capítulo 4) que séries temporais financeiras são geradas por componentes lineares dominantes e componentes não-lineares sub-dominantes. Vale mencionar que o termo de combinação da componente não-linear θ foi 0,4134, isto é, utilização de 41,31% do operador de dilatação e 58,69% do operador de erosão. A Tabela 6.2 apresenta os resultados (referentes ao conjunto de teste) com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP.

Analisando os resultados obtidos nos experimentos realizados com a série BVSP, claramente pode-se verificar um desempenho preditivo superior do modelo DELP para todas as métricas investigadas. Vale a pena observar a métrica THEIL, onde o modelo proposto obteve um valor bastante próximo a zero, mostrando que o dilema do passeio aleatório foi superado. A Tabela 6.3 apresenta os ganhos percentuais (do conjunto de teste) do modelo DELP, com todas as métricas de desempenho, em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP,

Tabela 6.1 Resultados de todos os experimentos utilizando o modelo DELP para a série BVSP (conjunto de treinamento)

DELP ($\mu; \sigma$)	Estatística	Métricas					
		MSE	MAPE	THEIL	ARV	POCID	EF
(0,001; 0,005)	MEAN	5,5588e-006	3,5424e-003	1,2420e-002	1,5121e-004	100,00	98,4291
	STD	5,4852e-006	1,8898e-003	1,2256e-002	1,4920e-004	0,00	1,3717
	CI	$\pm 6,3289e-006$	$\pm 2,1804e-003$	$\pm 1,4141e-002$	$\pm 1,7215e-004$	$\pm 0,00$	$\pm 1,5826$
(0,001; 0,05)	MEAN	1,8843e-005	5,5872e-003	4,2101e-002	5,1256e-004	100,00	95,7161
	STD	2,8679e-005	5,1240e-003	6,4078e-002	7,8010e-004	0,00	5,9327
	CI	$\pm 3,3090e-005$	$\pm 5,9121e-003$	$\pm 7,3934e-002$	$\pm 9,0009e-004$	$\pm 0,00$	$\pm 6,8452$
(0,001; 0,5)	MEAN	5,1570e-006	3,4517e-003	1,1522e-002	1,4028e-004	100,00	98,5190
	STD	3,8987e-006	1,7240e-003	8,7110e-003	1,0605e-004	0,00	1,0214
	CI	$\pm 4,4984e-006$	$\pm 1,9891e-003$	$\pm 1,0051e-002$	$\pm 1,2236e-004$	$\pm 0,00$	$\pm 1,1785$
(0,01; 0,005)	MEAN	5,0958e-006	3,7489e-003	1,1386e-002	1,3861e-004	100,00	98,4955
	STD	7,9752e-007	2,9196e-004	1,7819e-003	2,1693e-005	0,00	0,2032
	CI	$\pm 9,2019e-007$	$\pm 3,3686e-004$	$\pm 2,0560e-003$	$\pm 2,5030e-005$	$\pm 0,00$	$\pm 0,2345$
(0,01; 0,05)	MEAN	4,3399e-006	3,4636e-003	9,6967e-003	1,1805e-004	100,00	98,6893
	STD	4,9526e-007	1,9734e-004	1,1066e-003	1,3472e-005	0,00	0,1283
	CI	$\pm 5,7144e-007$	$\pm 2,2770e-004$	$\pm 1,2768e-003$	$\pm 1,5544e-005$	$\pm 0,00$	$\pm 0,1481$
(0,01; 0,5)	MEAN	4,4977e-006	3,5245e-003	1,0049e-002	1,2234e-004	100,00	98,6486
	STD	5,8837e-007	2,3068e-004	1,3146e-003	1,6004e-005	0,00	0,1520
	CI	$\pm 6,7887e-007$	$\pm 2,6616e-004$	$\pm 1,5168e-003$	$\pm 1,8466e-005$	$\pm 0,00$	$\pm 0,1753$
(0,1; 0,005)	MEAN	2,7189e-006	2,7426e-003	6,0750e-003	7,3958e-005	100,00	99,1185
	STD	2,5309e-007	1,2851e-004	5,6549e-004	6,8844e-006	0,00	0,0689
	CI	$\pm 2,9202e-007$	$\pm 1,4827e-004$	$\pm 6,5247e-004$	$\pm 7,9433e-006$	$\pm 0,00$	$\pm 0,0795$
(0,1; 0,05)	MEAN	3,1810e-006	2,9377e-003	7,1074e-003	8,6527e-005	100,00	98,9974
	STD	1,1051e-006	4,8137e-004	2,4691e-003	3,0060e-005	0,00	0,2912
	CI	$\pm 1,2751e-006$	$\pm 5,5541e-004$	$\pm 2,8489e-003$	$\pm 3,4683e-005$	$\pm 0,00$	$\pm 0,3360$
(0,1; 0,5)	MEAN	2,2523e-006	2,4909e-003	5,0324e-003	6,1266e-005	100,00	99,2471
	STD	4,0925e-007	2,1654e-004	9,1438e-004	1,1132e-005	0,00	0,1124
	CI	$\pm 4,7219e-007$	$\pm 2,4985e-004$	$\pm 1,0550e-003$	$\pm 1,2844e-005$	$\pm 0,00$	$\pm 0,1297$

Tabela 6.2 Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série BVSP.

Modelo	MSE	MAPE	THEIL	ARV	POCID	EF
ARIMA	6,4877e-004	3,4121e-002	1,4588e+000	1,7358e-002	48,19	19,1932
MLP	4,6455e-004	2,7422e-002	1,0649e+000	1,2975e-002	51,63	24,5164
MGA-MLP	4,2921e-004	2,6136e-002	9,8389e-001	1,1988e-002	52,03	25,7275
PSO-MLP	4,4072e-004	2,6313e-002	1,0103e+000	1,2310e-002	53,66	26,1832
TAEF	1,4864e-004	1,2189e-002	3,3959e-001	4,1093e-003	93,90	69,2480
SHIF	1,4849e-004	1,2184e-002	3,3925e-001	4,1051e-003	93,90	69,2658
DELP	2,1079e-006	2,4170e-003	4,7097e-003	5,7338e-005	100,00	99,2865

TAEF e SHIF.

Observando os ganhos percentuais apresentados na Tabela 6.3, pode-se notar que o modelo proposto obteve um desempenho preditivo bastante superior aos modelos MGA-MLP, PSO-MLP, MLP e ARIMA (apresentando ganho percentual superior a 86% para as todas as métricas), e superior aos modelos TAEF e SHIF (apresentando ganho percentual superior a 80% para as métricas MSE, MAPE, THEIL e ARV, e em torno de 6% para a métrica POCID). Em termos de desempenho preditivo global, pode-se verificar que o modelo proposto obteve ganho percentual acima de 279% em relação aos modelos ARIMA, MLP, MGA-MLP e PSO-MLP, e em torno de 43% em relação aos modelos TAEF e SHIF.

Tabela 6.3 Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série BVSP.

	MSE	MAPE	THEIL	ARV	POCID	EF
Ganho (%) DELP / ARIMA	99,68	92,92	99,68	99,67	107,51	417,30
Ganho (%) DELP / MLP	99,55	91,19	99,56	99,56	93,69	304,98
Ganho (%) DELP / MGA-MLP	99,51	90,75	99,52	99,52	92,20	285,92
Ganho (%) DELP / PSO-MLP	99,52	90,81	99,53	99,53	86,36	279,20
Ganho (%) DELP / TAEF	98,58	80,17	98,61	98,60	6,50	43,38
Ganho (%) DELP / SHIF	98,58	80,16	98,61	98,60	6,50	43,34

As Figuras 6.1 e 6.2 apresentam um gráfico comparativo entre os valores reais da série (linha sólida) e os valores preditos pelo modelo proposto (linha tracejada) para os últimos cem e dez pontos, respectivamente, conjunto de teste. Note que os valores preditos estão próximos aos valores reais da série, sem a presença do característico atraso de um passo da previsão, ou seja, a distorção de fase temporal que origina o dilema do passeio aleatório foi corretamente ajustada.

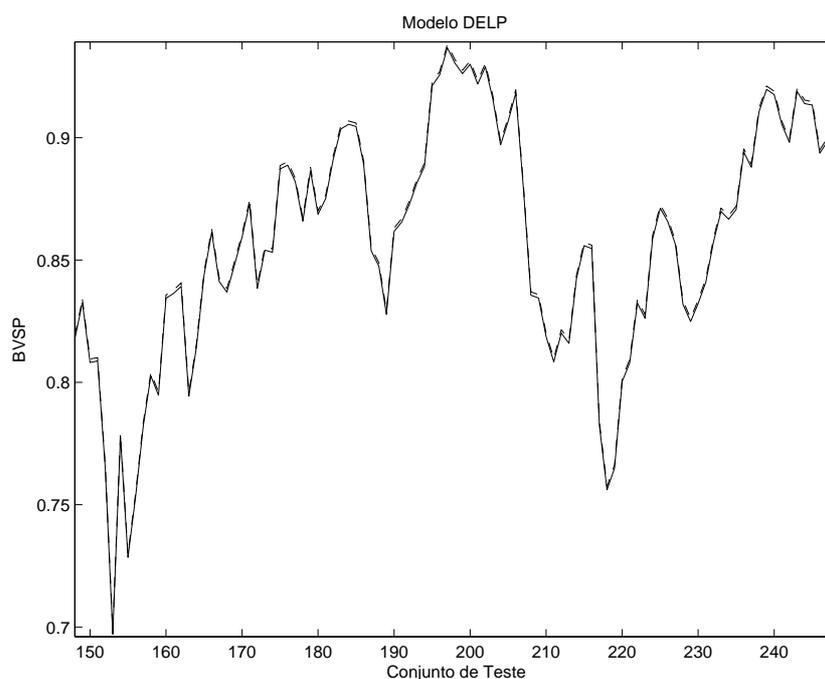


Figura 6.1 Previsão da série BVSP gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).

6.2.2 Índice Dow Jones Industrial Average (DJI)

Para a série do Índice Dow Jones Industrial Average (DJI), os modelos utilizados que obtiveram o melhor desempenho foram: i) modelo ARIMA com a configuração (5,1,1), ii) modelo MLP

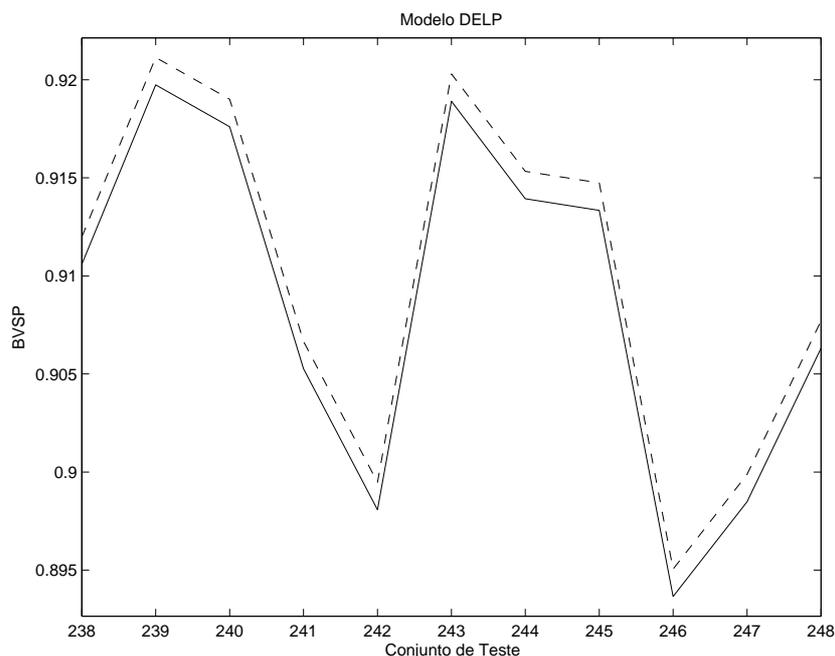


Figura 6.2 Previsão da série BVSP gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).

com a configuração 5-5-1 e arquitetura definida pela Equação (3.46), iii) o modelo MGA-MLP definiu os retardos temporais (1, 3, 5, 6, 7, 8 e 9) e a rede MLP com a configuração 7-6-1 e arquitetura definida pela Equação (3.44), iv) o modelo PSO-MLP escolheu os retardos temporais (1, 4, 6, 9 e 10) e a rede MLP com a configuração 5-8-1 e arquitetura definida pela Equação (3.44), v) o modelo TAEF determinou os retardos temporais (2, 5, 8, 9 e 10) e a rede MLP com a configuração 5-8-1, arquitetura definida pela Equação (3.44) e o algoritmo *resilient backpropagation* para treinamento da rede MLP, e vi) o modelo SHIF determinou os retardos temporais (5, 6, 9 e 10) e a rede MLP com a configuração 4-10-1, arquitetura definida pela Equação (3.44) e o algoritmo *levenberg marquardt* para treinamento da rede MLP.

A Tabela 6.4 apresenta todos os experimentos realizados com o modelo DELP (utilizando o conjunto de treinamento) variando o valor de μ e σ , onde foram calculadas todas métricas, bem como as estatísticas MEAN, STD e CI para análise do desempenho de previsão médio do modelo proposto.

De acordo com a Tabela 6.4, pode-se verificar que os melhores valores de μ e σ para a série DJI foram 0,01 e 0,5, respectivamente, uma vez que obtivemos o melhor desempenho de previsão em termos da função de avaliação global (EF). Portanto, assumindo essa suposição, escolheu-se o modelo DELP(0,01; 0,5) com a maior métrica EF no conjunto de validação, dentre os dez modelos criados neste experimento, para representar o modelo proposto. Neste modelo, o termo de combinação λ determinado foi 0,0425, significando que o DELP utilizou 95,75% da componente linear e 4,25% da componente não-linear, confirmando a hipótese (através da análise do *lagplot* apresentada no Capítulo 4) que séries temporais financeiras são geradas por componentes lineares dominantes e componentes não-lineares sub-dominantes.

Tabela 6.4 Resultados de todos os experimentos utilizando o modelo proposto para a série DJI (conjunto de treinamento)

DELDP ($\mu; \sigma$)	Estatística	Métricas					
		MSE	MAPE	THEIL	ARV	POCID	EF
(0,001; 0,005)	MEAN	7,9791e-006	7,2760e-003	4,7561e-002	5,7242e-004	100,00	94,9165
	STD	7,5833e-006	4,2022e-003	4,5203e-002	5,4403e-004	0,00	4,4228
	CI	$\pm 8,7498e-006$	$\pm 4,8485e-003$	$\pm 5,2155e-002$	$\pm 6,2771e-004$	$\pm 0,00$	$\pm 5,1031$
(0,001; 0,05)	MEAN	7,9785e-006	6,0002e-003	4,7558e-002	5,7239e-004	100,00	95,2260
	STD	1,1301e-005	6,2310e-003	6,7366e-002	8,1078e-004	0,00	6,4168
	CI	$\pm 1,3040e-005$	$\pm 7,1894e-003$	$\pm 7,7727e-002$	$\pm 9,3548e-004$	$\pm 0,00$	$\pm 7,4038$
(0,001; 0,5)	MEAN	4,2185e-006	5,2447e-003	2,5146e-002	3,0264e-004	100,00	97,0700
	STD	3,7389e-006	3,1526e-003	2,2287e-002	2,6823e-004	0,00	2,4227
	CI	$\pm 4,3140e-006$	$\pm 3,6375e-003$	$\pm 2,5715e-002$	$\pm 3,0949e-004$	$\pm 0,00$	$\pm 2,7953$
(0,01; 0,005)	MEAN	2,3987e-006	4,4862e-003	1,4298e-002	1,7209e-004	100,00	98,1396
	STD	2,2946e-007	2,1198e-004	1,3677e-003	1,6461e-005	0,00	0,1536
	CI	$\pm 2,6475e-007$	$\pm 2,4458e-004$	$\pm 1,5781e-003$	$\pm 1,8993e-005$	$\pm 0,00$	$\pm 0,1772$
(0,01; 0,05)	MEAN	2,5319e-006	4,6088e-003	1,5092e-002	1,8164e-004	100,00	98,0505
	STD	2,5526e-007	2,2588e-004	1,5216e-003	1,8313e-005	0,00	0,1694
	CI	$\pm 2,9453e-007$	$\pm 2,6062e-004$	$\pm 1,7556e-003$	$\pm 2,1130e-005$	$\pm 0,00$	$\pm 0,1955$
(0,01; 0,5)	MEAN	2,1877e-006	4,2443e-003	1,3041e-002	1,5695e-004	100,00	98,2870
	STD	6,1686e-007	6,8461e-004	3,6769e-003	4,4254e-005	0,00	0,4281
	CI	$\pm 7,1174e-007$	$\pm 7,8991e-004$	$\pm 4,2425e-003$	$\pm 5,1060e-005$	$\pm 0,00$	$\pm 0,4940$
(0,1; 0,005)	MEAN	4,8168e-006	6,1644e-003	2,8712e-002	3,4556e-004	100,00	96,6193
	STD	2,6100e-006	1,7631e-003	1,5558e-002	1,8724e-004	0,00	1,6286
	CI	$\pm 3,0115e-006$	$\pm 2,0342e-003$	$\pm 1,7951e-002$	$\pm 2,1605e-004$	$\pm 0,00$	$\pm 1,8791$
(0,1; 0,05)	MEAN	6,1387e-006	7,0279e-003	3,6591e-002	4,4039e-004	100,00	95,7985
	STD	2,4387e-006	1,6605e-003	1,4536e-002	1,7495e-004	0,00	1,5246
	CI	$\pm 2,8137e-006$	$\pm 1,9159e-003$	$\pm 1,6772e-002$	$\pm 2,0186e-004$	$\pm 0,00$	$\pm 1,7591$
(0,1; 0,5)	MEAN	4,3218e-006	5,9366e-003	2,5761e-002	3,1005e-004	100,00	96,9071
	STD	1,6499e-006	1,1639e-003	9,8349e-003	1,1837e-004	0,00	1,0429
	CI	$\pm 1,9037e-006$	$\pm 1,3430e-003$	$\pm 1,1348e-002$	$\pm 1,3657e-004$	$\pm 0,00$	$\pm 1,2033$

Vale mencionar que o termo de combinação da componente não-linear θ foi 0,4719, isto é, utilização de 47,19% do operador de dilatação e 52,81% do operador de erosão. A Tabela 6.5 apresenta os resultados (referentes ao conjunto de teste) com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP.

Tabela 6.5 Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série DJI.

Modelo	MSE	MAPE	THEIL	ARV	POCID	EF
ARIMA	2,4341e-004	3,8867e-002	1,4580e+000	1,7115e-002	51,00	20,2858
MLP	1,8729e-004	3,2314e-002	1,1453e+000	1,3722e-002	44,31	20,2186
MGA-MLP	2,0080e-004	3,2135e-002	1,2279e+000	1,4712e-002	48,37	21,2638
PSO-MLP	1,9474e-004	3,2823e-002	1,1908e+000	1,4268e-002	49,19	21,9771
TAEF	6,9064e-005	2,0547e-002	4,2062e-001	5,0847e-003	90,61	62,6502
SHIF	6,5303e-005	1,9962e-002	3,9771e-001	4,8079e-003	90,61	63,6971
DELP	1,0854e-006	3,0205e-003	6,4698e-003	7,7867e-005	100,00	99,0521

É possível verificar, de acordo com a Tabela 6.5, que o desempenho preditivo do modelo DELP nos experimentos realizados com a série DJI para todas as métricas investigadas é superior aos resultados encontrados na literatura. Note que, mais uma vez, a métrica THEIL obtida pelo modelo proposto foi um valor bastante próximo a zero, sendo este resultado um indicador

que o dilema do passeio aleatório foi superado. Os ganhos percentuais (do conjunto de teste) do modelo proposto com todas as métricas de desempenho em relação aos modelos ARIMA, MLP, MGA–MLP, PSO–MLP, TAEF e SHIF são apresentados na Tabela 6.6.

Tabela 6.6 Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA–MLP, PSO–MLP, TAEF e SHIF para a série DJI.

	MSE	MAPE	THEIL	ARV	POCID	EF
Ganho (%) DELP / ARIMA	99,55	92,23	99,56	99,55	96,08	388,28
Ganho (%) DELP / MLP	99,42	90,65	99,44	99,43	125,68	389,91
Ganho (%) DELP / MGA–MLP	99,46	90,60	99,47	99,47	106,74	365,83
Ganho (%) DELP / PSO–MLP	99,44	90,80	99,46	99,45	103,29	350,71
Ganho (%) DELP / TAEF	98,43	85,30	98,46	98,47	10,36	58,10
Ganho (%) DELP / SHIF	98,34	84,87	98,37	98,38	10,36	55,50

De acordo com a Tabela 6.6, pode-se verificar um desempenho preditivo bastante superior do modelo proposto em relação aos modelos MGA–MLP, PSO–MLP, MLP e ARIMA (obtendo um ganho percentual superior a 90% para todas as métricas) e superior aos modelos TAEF e SHIF (alcançando um ganho percentual superior a 84% para as métricas MSE, MAPE, THEIL e ARV, e em torno de 10% para a métrica POCID). Avaliando o modelo proposto em termos de desempenho preditivo global, pode-se verificar que obtivemos um ganho percentual superior a 350% em relação aos modelos ARIMA, MLP, MGA–MLP e PSO–MLP, e superior a 55% em relação aos modelos TAEF e SHIF.

As Figuras 6.3 e 6.4 mostram um gráfico comparativo entre os valores reais da série (linha sólida) e os valores preditos pelo modelo proposto (linha tracejada) para os cem e dez últimos pontos, respectivamente, do conjunto de teste. Note que os valores preditos estão bem próximos aos valores reais da série, onde o característico atraso de um passo da previsão não ocorreu, isto é, a distorção de fase temporal que origina o dilema do passeio aleatório foi ajustada com sucesso.

6.2.3 Índice *National Association of Securities Dealers Automated Quotation* (IXIC)

Para a série do Índice *National Association of Securities Dealers Automated Quotation* (IXIC), os modelos utilizados que obtiveram o melhor desempenho foram: i) modelo ARIMA com a configuração (4,1,1), ii) modelo MLP com a configuração 3-25-1 e arquitetura definida pela Equação (3.46), iii) o modelo MGA–MLP definiu os retardos temporais (1, 4, 5, 7, 9 e 10) e a rede MLP com a configuração 6-3-1 e arquitetura definida pela Equação (3.46), iv) o modelo PSO–MLP escolheu os retardos temporais (1, 2, 4, 6, 7, 8, e 9) e a rede MLP com a configuração 7-9-1 e arquitetura definida pela Equação (3.44), v) o modelo TAEF determinou apenas o retardo temporal (3) e a rede MLP com a configuração 1-6-1, arquitetura definida pela Equação (3.46) e o algoritmo *scaled conjugate gradient* para treinamento da rede MLP, e vi) o modelo SHIF determinou os retardos temporais (3, 4, 5, 7 e 10) e a rede MLP com a configuração 5-10-1, arquitetura definida pela Equação (3.46) e o algoritmo *scaled conjugate gradient* para treinamento da rede MLP.

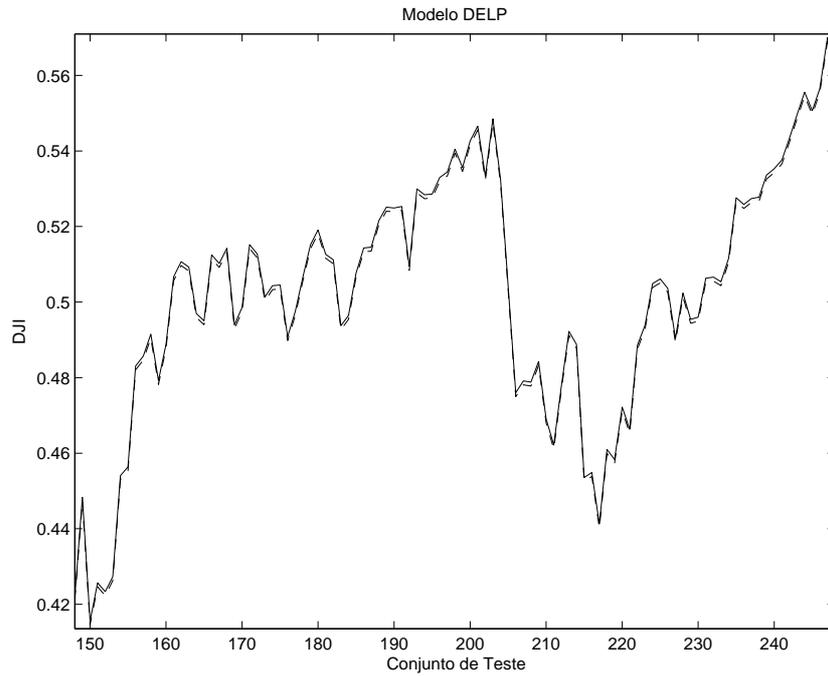


Figura 6.3 Previsão da série DJI gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).

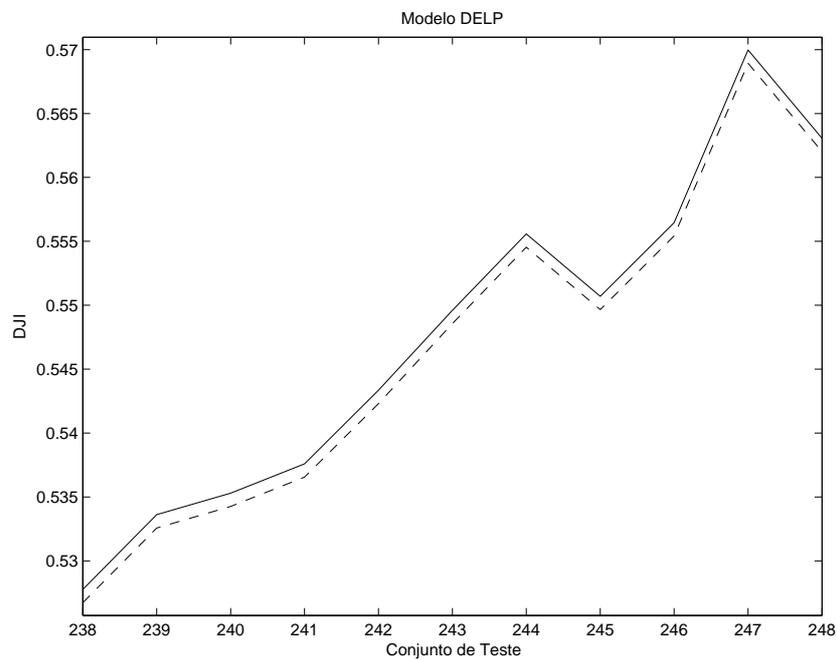


Figura 6.4 Previsão da série DJI gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).

A Tabela 6.7 apresenta todos os experimentos realizados com o modelo DELP (utilizando o conjunto de treinamento) variando o valor de μ e σ , onde foram calculadas todas métricas, bem como as estatísticas MEAN, STD e CI para análise do desempenho de previsão médio do modelo proposto.

Tabela 6.7 Resultados de todos os experimentos utilizando o modelo proposto para a série IXIC (conjunto de treinamento)

DELP ($\mu; \sigma$)	Estatística	Métricas					
		MSE	MAPE	THEIL	ARV	POCID	EF
(0,001; 0,005)	MEAN	1,0864e-005	5,2368e-003	4,6331e-002	5,6558e-004	100,00	95,6066
	STD	2,0661e-005	6,3186e-003	8,8108e-002	1,0756e-003	0,00	7,7550
	CI	$\pm 2,3839e-005$	$\pm 7,2905e-003$	$\pm 1,0166e-001$	$\pm 1,2410e-003$	$\pm 0,00$	$\pm 8,9478$
(0,001; 0,05)	MEAN	7,0869e-006	4,2570e-003	3,0222e-002	3,6893e-004	100,00	96,9023
	STD	1,3563e-005	5,0747e-003	5,7841e-002	7,0608e-004	0,00	5,5040
	CI	$\pm 1,5650e-005$	$\pm 5,8552e-003$	$\pm 6,6737e-002$	$\pm 8,1469e-004$	$\pm 0,00$	$\pm 6,3505$
(0,001; 0,5)	MEAN	6,3963e-006	4,7888e-003	2,7277e-002	3,3298e-004	100,00	96,9424
	STD	6,8958e-006	3,8758e-003	2,9407e-002	3,5898e-004	0,00	3,1216
	CI	$\pm 7,9564e-006$	$\pm 4,4719e-003$	$\pm 3,3930e-002$	$\pm 4,1420e-004$	$\pm 0,00$	$\pm 3,6017$
(0,01; 0,005)	MEAN	4,3057e-006	4,8498e-003	1,8361e-002	2,2414e-004	100,00	97,7098
	STD	1,6218e-007	9,0531e-005	6,9161e-004	8,4427e-006	0,00	0,0754
	CI	$\pm 1,8712e-007$	$\pm 1,0446e-004$	$\pm 7,9798e-004$	$\pm 9,7413e-006$	$\pm 0,00$	$\pm 0,0870$
(0,01; 0,05)	MEAN	9,0002e-006	6,3467e-003	3,8381e-002	4,6853e-004	100,00	95,8307
	STD	1,0473e-005	3,3351e-003	4,4662e-002	5,4520e-004	0,00	4,1906
	CI	$\pm 1,2084e-005$	$\pm 3,8480e-003$	$\pm 5,1531e-002$	$\pm 6,2906e-004$	$\pm 0,00$	$\pm 4,8352$
(0,01; 0,5)	MEAN	4,8700e-006	5,1405e-003	2,0768e-002	2,5352e-004	100,00	97,4516
	STD	9,5769e-007	4,8205e-004	4,0840e-003	4,9855e-005	0,00	0,4361
	CI	$\pm 1,1050e-006$	$\pm 5,5620e-004$	$\pm 4,7122e-003$	$\pm 5,7523e-005$	$\pm 0,00$	$\pm 0,5031$
(0,1; 0,005)	MEAN	1,0787e-005	7,6650e-003	4,6001e-002	5,6155e-004	100,00	94,8580
	STD	1,3618e-006	4,8602e-004	5,8074e-003	7,0893e-005	0,00	0,5728
	CI	$\pm 1,5713e-006$	$\pm 5,6077e-004$	$\pm 6,7006e-003$	$\pm 8,1797e-005$	$\pm 0,00$	$\pm 0,6609$
(0,1; 0,05)	MEAN	6,4820e-006	5,8075e-003	2,7642e-002	3,3744e-004	100,00	96,7509
	STD	3,5326e-006	1,4542e-003	1,5065e-002	1,8390e-004	0,00	1,5337
	CI	$\pm 4,0760e-006$	$\pm 1,6779e-003$	$\pm 1,7382e-002$	$\pm 2,1219e-004$	$\pm 0,00$	$\pm 1,7696$
(0,1; 0,5)	MEAN	1,0270e-005	7,4345e-003	4,3795e-002	5,3462e-004	100,00	95,0885
	STD	2,6934e-006	1,0271e-003	1,1486e-002	1,4021e-004	0,00	1,1511
	CI	$\pm 3,1076e-006$	$\pm 1,1851e-003$	$\pm 1,3252e-002$	$\pm 1,6178e-004$	$\pm 0,00$	$\pm 1,3281$

De acordo com a Tabela 6.7, pode-se verificar que os melhores valores de μ e σ para a série IXIC foram 0,01 e 0,005, respectivamente, uma vez que obtivemos o melhor desempenho de previsão em termos da função de avaliação global (EF). Portanto, assumindo essa suposição, escolheu-se o modelo DELP(0,01; 0,005) com a maior métrica EF no conjunto de validação, dentre os dez modelos criados neste experimento, para representar o modelo proposto. Neste modelo, o termo de combinação λ determinado foi 0,0365, significando que o DELP utilizou 96,35% da componente linear e 3,65% da componente não-linear, confirmando a hipótese (através da análise do *lagplot* apresentada no Capítulo 4) que séries temporais financeiras são geradas por componentes lineares dominantes e componentes não-lineares sub-dominantes. Vale mencionar que o termo de combinação da componente não-linear θ foi 0,7583, isto é, utilização de 24,17% do operador de dilatação e 75,83% do operador de erosão. A Tabela 6.8 apresenta os resultados (referentes ao conjunto de teste) com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP.

Os resultados obtidos nos experimentos realizados com a série IXIC mostram que o desempenho preditivo do modelo DELP para todas as métricas investigadas é superior aos resultados encontrados na literatura. Mais uma vez o valor da métrica THEIL foi próxima a zero, o que

Tabela 6.8 Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série IXIC.

Modelo	MSE	MAPE	THEIL	ARV	POCID	EF
ARIMA	3,7973e-004	3,9232e-002	1,6245e+000	1,9266e-002	52,61	19,6061
MLP	2,6533e-004	3,0814e-002	1,1515e+000	1,4168e-002	50,41	22,9458
MGA-MLP	2,6362e-004	3,1115e-002	1,1441e+000	1,4077e-002	53,66	24,5065
PSO-MLP	2,4816e-004	2,9924e-002	1,0770e+000	1,3251e-002	52,85	24,9221
TAEF	2,9907e-004	3,6009e-002	1,2748e+000	1,5836e-002	95,53	41,0532
SHIF	2,9905e-004	3,6008e-002	1,2747e+000	1,5836e-002	95,53	41,0544
DELP	4,1873e-006	4,7833e-003	1,7856e-002	2,1798e-004	100,00	97,7649

nos mostra que o dilema do passeio aleatório foi superado. A Tabela 6.9 apresenta os ganhos percentuais (do conjunto de teste) do modelo DELP proposto com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF.

Tabela 6.9 Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série IXIC.

	MSE	MAPE	THEIL	ARV	POCID	EF
Ganho (%) DELP / ARIMA	98,90	87,81	98,90	98,87	90,08	398,65
Ganho (%) DELP / MLP	98,42	84,48	98,45	98,46	98,37	326,07
Ganho (%) DELP / MGA-MLP	98,41	84,63	98,44	98,45	86,36	298,93
Ganho (%) DELP / PSO-MLP	98,31	84,02	98,34	98,35	89,21	292,28
Ganho (%) DELP / TAEF	98,60	86,72	98,60	98,62	4,68	138,14
Ganho (%) DELP / SHIF	98,60	86,72	98,60	98,62	4,68	138,14

Os ganhos percentuais do modelo DELP proposto em relação aos modelos investigados apresentados na Tabela 6.9 nos mostram que o modelo proposto obteve um desempenho bastante superior que os modelos SHIF, TAEF, MGA-MLP, PSO-MLP, MLP, ARIMA (ganho percentual superior a 84% para todas as métricas, exceto para a métrica POCID dos modelos SHIF e TAEF que obtiveram um ganho percentual em torno de 4,6%). Em termos globais, o modelo proposto obteve um ganho percentual superior a 138% em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF.

As Figuras 6.5 e 6.6 mostram um gráfico comparativo entre os valores reais da série (linha sólida) e os valores preditos pelo modelo proposto (linha tracejada) para os últimos cem e dez pontos, respectivamente, do conjunto de teste. Vale mencionar que a previsão está bastante próxima aos valores reais da série, o que comprova que o característico atraso de um passo da previsão não ocorreu e o dilema do passeio aleatório foi mais uma vez superado.

6.2.4 Índice *Financial Times and London Stock Exchange 100* (FTSE)

Para a série do Índice *Financial Times and London Stock Exchange 100* (FTSE), os modelos utilizados que obtiveram o melhor desempenho foram: i) modelo ARIMA com a configuração (6,1,1), ii) modelo MLP com a configuração 10-10-1 e arquitetura definida pela Equação (3.47),

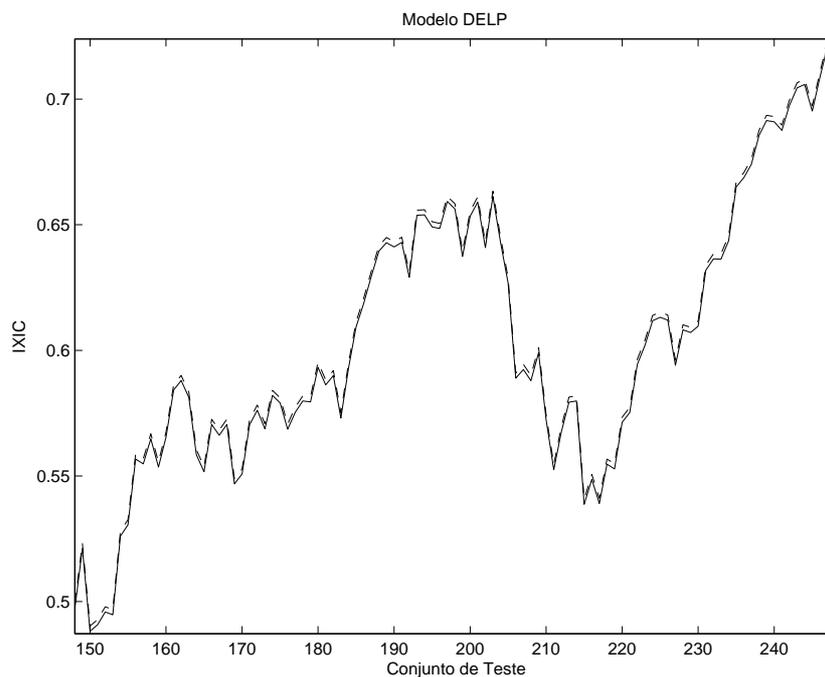


Figura 6.5 Previsão da série IXIC gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).

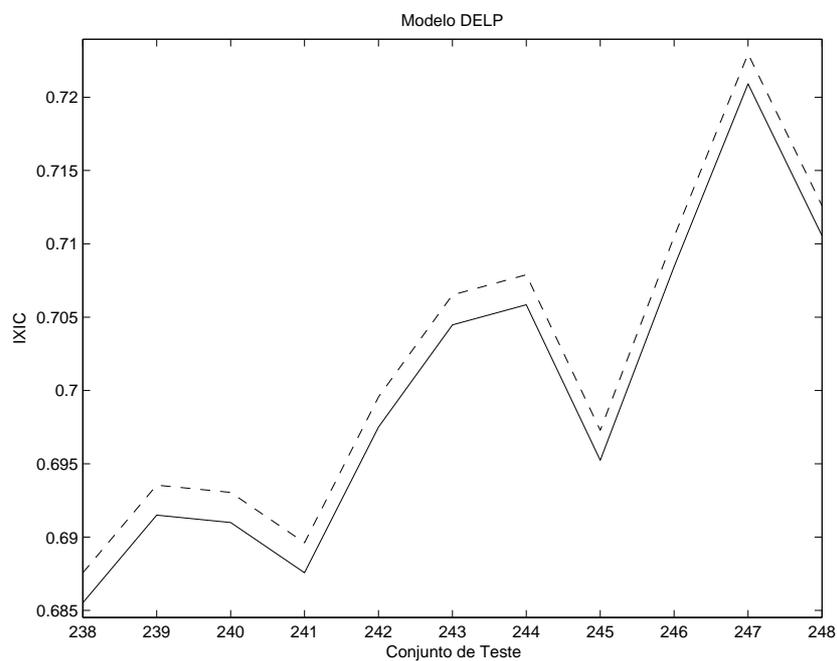


Figura 6.6 Previsão da série IXIC gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).

iii) o modelo MGA–MLP definiu os retardos temporais (1, 2, 3, 4, 5, 7 e 9) e a rede MLP com a configuração 7-3-1 e arquitetura definida pela Equação (3.46), iv) o modelo PSO–MLP escolheu os retardos temporais (1, 2, 3, 5, 7 e 8) e a rede MLP com a configuração 6-7-1 e arquitetura definida pela Equação (3.46), v) o modelo TAEF determinou os retardos temporais (4, 5, 6, 7 e 9) e a rede MLP com a configuração 5-4-1, arquitetura definida pela Equação (3.46) e o algoritmo *levenberg marquardt* para treinamento da rede MLP, e vi) o modelo SHIF determinou os retardos temporais (3, 5, 6, 9 e 10) e a rede MLP com a configuração 5-2-1, arquitetura definida pela Equação (3.46) e o algoritmo *levenberg marquardt* para treinamento da rede MLP.

A Tabela 6.10 apresenta todos os experimentos realizados com o modelo DELP (utilizando o conjunto de treinamento) variando o valor de μ e σ , onde foram calculadas todas métricas, bem como as estatísticas MEAN, STD e CI para análise do desempenho de previsão médio do modelo proposto.

Tabela 6.10 Resultados de todos os experimentos utilizando o modelo proposto para a série FTSE (conjunto de treinamento)

DELP ($\mu; \sigma$)	Estatística	Métricas					
		MSE	MAPE	THEIL	ARV	POCID	EF
(0,001; 0,005)	MEAN	9,2425e-006	6,8602e-003	3,2474e-002	3,9914e-004	100,00	96,3385
	STD	1,2197e-005	5,4518e-003	4,2855e-002	5,2673e-004	0,00	4,2980
	CI	$\pm 1,4073e-005$	$\pm 6,2904e-003$	$\pm 4,9446e-002$	$\pm 6,0774e-004$	$\pm 0,00$	$\pm 4,9590$
(0,001; 0,05)	MEAN	2,0851e-006	3,4697e-003	7,3263e-003	9,0047e-005	100,00	98,9292
	STD	1,9318e-006	2,2201e-003	6,7875e-003	8,3424e-005	0,00	0,8773
	CI	$\pm 2,2289e-006$	$\pm 2,5616e-003$	$\pm 7,8314e-003$	$\pm 9,6256e-005$	$\pm 0,00$	$\pm 1,0123$
(0,001; 0,5)	MEAN	4,9358e-006	5,5311e-003	1,7342e-002	2,1315e-004	100,00	97,7721
	STD	4,7208e-006	3,0079e-003	1,6587e-002	2,0387e-004	0,00	1,8777
	CI	$\pm 5,4469e-006$	$\pm 3,4706e-003$	$\pm 1,9138e-002$	$\pm 2,3522e-004$	$\pm 0,00$	$\pm 2,1665$
(0,01; 0,005)	MEAN	3,0852e-006	4,3944e-003	1,0840e-002	1,3323e-004	100,00	98,4924
	STD	1,8787e-006	2,3282e-003	6,6010e-003	8,1133e-005	0,00	0,8793
	CI	$\pm 2,1677e-006$	$\pm 2,6862e-003$	$\pm 7,6163e-003$	$\pm 9,3612e-005$	$\pm 0,00$	$\pm 1,0146$
(0,01; 0,05)	MEAN	3,6607e-006	5,1799e-003	1,2862e-002	1,5809e-004	100,00	98,2151
	STD	1,4068e-006	1,2383e-003	4,9428e-003	6,0752e-005	0,00	0,6072
	CI	$\pm 1,6232e-006$	$\pm 1,4287e-003$	$\pm 5,7031e-003$	$\pm 7,0097e-005$	$\pm 0,00$	$\pm 0,7006$
(0,01; 0,5)	MEAN	4,2287e-006	5,6931e-003	1,4858e-002	1,8262e-004	100,00	97,9684
	STD	4,9539e-008	3,3315e-005	1,7406e-004	2,1394e-006	0,00	0,0201
	CI	$\pm 5,7159e-008$	$\pm 3,8439e-005$	$\pm 2,0083e-004$	$\pm 2,4684e-006$	$\pm 0,00$	$\pm 0,0232$
(0,1; 0,005)	MEAN	1,0815e-005	8,9541e-003	3,7999e-002	4,6704e-004	100,00	95,4878
	STD	3,7197e-006	1,8430e-003	1,3069e-002	1,6063e-004	0,00	1,4010
	CI	$\pm 4,2918e-006$	$\pm 2,1264e-003$	$\pm 1,5079e-002$	$\pm 1,8534e-004$	$\pm 0,00$	$\pm 1,6165$
(0,1; 0,05)	MEAN	8,6222e-006	7,9389e-003	3,0294e-002	3,7235e-004	100,00	96,3013
	STD	4,0634e-006	1,9559e-003	1,4277e-002	1,7548e-004	0,00	1,5194
	CI	$\pm 4,6885e-006$	$\pm 2,2567e-003$	$\pm 1,6473e-002$	$\pm 2,0247e-004$	$\pm 0,00$	$\pm 1,7531$
(0,1; 0,5)	MEAN	9,3846e-006	8,3910e-003	3,2973e-002	4,0527e-004	100,00	95,9998
	STD	2,9711e-006	1,3791e-003	1,0439e-002	1,2831e-004	0,00	1,1022
	CI	$\pm 3,4280e-006$	$\pm 1,5912e-003$	$\pm 1,2045e-002$	$\pm 1,4804e-004$	$\pm 0,00$	$\pm 1,2717$

De acordo com a Tabela 6.10, pode-se verificar que os melhores valores de μ e σ para a série FTSE foram 0,001 e 0,05, respectivamente, uma vez que obtivemos o melhor desempenho de previsão em termos da função de avaliação global (EF). Portanto, assumindo essa suposição, escolheu-se o modelo DELP(0,001; 0,05) com a maior métrica EF no conjunto de validação, dentre os dez modelos criados neste experimento, para representar o modelo proposto. Neste modelo, o termo de combinação λ determinado foi 0,0032, significando que o DELP utilizou 99,68% da componente linear e 0,32% da componente não-linear, confirmando a hipótese

(através da análise do *lagplot* apresentada no Capítulo 4) que séries temporais financeiras são geradas por componentes lineares dominantes e componentes não-lineares sub-dominantes. Vale mencionar que o termo de combinação da componente não-linear θ foi 0,3358, isto é, utilização de 33,58% do operador de dilatação e 66,42% do operador de erosão. A Tabela 6.11 apresenta os resultados (referentes ao conjunto de teste) com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP.

Tabela 6.11 Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série FTSE.

Modelo	MSE	MAPE	THEIL	ARV	POCID	EF
ARIMA	4,2162e-004	4,7460e-002	1,4866e+000	1,7864e-002	47,79	18,7248
MLP	4,2019e-004	4,3442e-002	1,5452e+000	1,8330e-002	53,66	20,5791
MGA-MLP	3,9325e-004	4,3060e-002	1,4461e+000	1,7155e-002	52,44	20,9191
PSO-MLP	3,2081e-004	4,0586e-002	1,1798e+000	1,3995e-002	50,81	22,7386
TAEF	1,7673e-004	3,2578e-002	6,4960e-001	7,7672e-003	92,65	54,8204
SHIF	1,0741e-004	2,6501e-002	3,9478e-001	4,7204e-003	78,78	55,2379
DELP	4,4308e-009	1,8428e-004	1,5568e-005	1,9134e-007	100,00	99,9800

Os resultados obtidos nos experimentos realizados com a série FTSE mostram que o desempenho preditivo do modelo DELP para todas as métricas investigadas é superior aos resultados encontrados na literatura. É possível verificar que a métrica THEIL obtida pelo modelo proposto foi bem próxima de zero, mostrando que o dilema do passeio aleatório foi superado novamente. A Tabela 6.12 mostra os ganhos percentuais (do conjunto de teste) do modelo DELP proposto com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF.

Tabela 6.12 Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série FTSE.

	MSE	MAPE	THEIL	ARV	POCID	EF
Ganho (%) DELP / ARIMA	100,00	99,61	100,00	100,00	109,25	433,94
Ganho (%) DELP / MLP	100,00	99,58	100,00	100,00	86,36	385,83
Ganho (%) DELP / MGA-MLP	100,00	99,57	100,00	100,00	90,69	377,94
Ganho (%) DELP / PSO-MLP	100,00	99,55	100,00	100,00	96,81	339,69
Ganho (%) DELP / TAEF	100,00	99,43	100,00	100,00	7,93	82,38
Ganho (%) DELP / SHIF	100,00	99,30	100,00	100,00	26,94	81,00

De acordo com a Tabela 6.12, verificamos um desempenho preditivo bastante superior do modelo proposto em relação aos modelos investigados (obtendo um ganho percentual superior a 86% para todas as métricas, exceto a métrica POCID nos modelos TAEF e SHIF, onde obtivemos um ganho percentual em torno de 7% e 26%, respectivamente). A análise de desempenho global mostrou que o modelo proposto obteve um ganho percentual superior a 81% em relação a todos modelos investigados.

As Figuras 6.7 e 6.8 apresentam um gráfico comparativo entre os valores reais da série (linha sólida) e os valores preditos pelo modelo proposto (linha tracejada) para os últimos cem e dez pontos, respectivamente, do conjunto de teste. Vale mencionar que a previsão está sobreposta aos valores reais da série, o que comprova que o característico atraso de um passo da previsão não ocorreu e o dilema do passei aleatório foi mais uma vez superado.

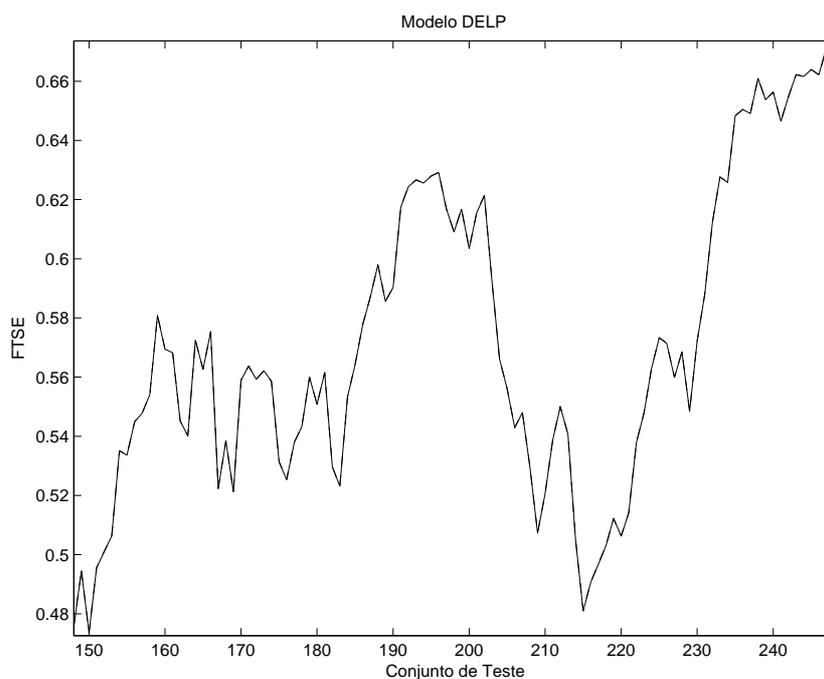


Figura 6.7 Previsão da série FTSE gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).

6.2.5 Preço das Ações do Bradesco PN (BBDC4)

Para a série do Preço das Ações do Bradesco PN (BBDC4), os modelos utilizados que obtiveram o melhor desempenho foram: i) modelo ARIMA com a configuração (2,1,1), ii) modelo MLP com a configuração 3-5-1 e arquitetura definida pela Equação (3.44), iii) o modelo MGA-MLP definiu os retardos temporais (1, 2, 5, 7, 9 e 10) e a rede MLP com a configuração 6-3-1 e arquitetura definida pela Equação (3.47), iv) o modelo PSO-MLP escolheu os retardos temporais (1, 4, 7 e 8) e a rede MLP com a configuração 4-1-1 e arquitetura definida pela Equação (3.47), v) o modelo TAEF determinou os retardos temporais (4, 5, 6, 7 e 8) e a rede MLP com a configuração 5-3-1, arquitetura definida pela Equação (3.47) e o algoritmo *levenberg marquardt* para treinamento da rede MLP, e vi) o modelo SHIF determinou os retardos temporais (2, 4 e 8) e a rede MLP com a configuração 3-10-1, arquitetura definida pela Equação (3.46) e o algoritmo *levenberg marquardt* para treinamento da rede MLP.

A Tabela 6.13 apresenta todos os experimentos realizados com o modelo DELP (utilizando o conjunto de treinamento) variando o valor de μ e σ , onde foram calculadas todas métricas,

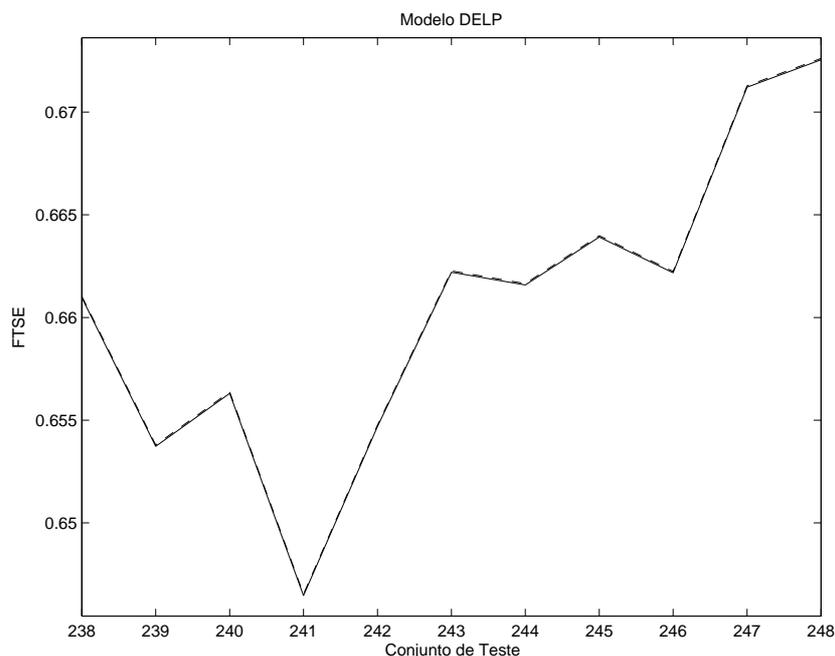


Figura 6.8 Previsão da série FTSE gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).

bem como as estatísticas MEAN, STD e CI para análise do desempenho de previsão médio do modelo proposto.

De acordo com a Tabela 6.13, pode-se verificar que os melhores valores de μ e σ para a série BBDC4 foram 0,01 e 0,5, respectivamente, uma vez que obtivemos o melhor desempenho de previsão em termos da função de avaliação global (EF). Portanto, assumindo essa suposição, escolheu-se o modelo DELP(0,01; 0,5) com a maior métrica EF no conjunto de validação, dentre os dez modelos criados neste experimento, para representar o modelo proposto. Neste modelo, o termo de combinação λ determinado foi 0,0051, significando que o DELP utilizou 99,49% da componente linear e 0,51% da componente não-linear, confirmando a hipótese (através da análise do *lagplot* apresentada no Capítulo 4) que séries temporais financeiras são geradas por componentes lineares dominantes e componentes não-lineares sub-dominantes. Vale mencionar que o termo de combinação da componente não-linear θ foi 0,6047, isto é, utilização de 60,47% do operador de dilatação e 39,53% do operador de erosão. A Tabela 6.14 apresenta os resultados (referentes ao conjunto de teste) com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP.

É possível verificar, de acordo com a Tabela 6.15, que para a série BBDC4 o modelo proposto obteve desempenho superior aos resultados encontrados na literatura. Mais uma vez a métrica THEIL foi bastante próxima de zero, o que indica a superação do dilema do passeio aleatório. A Tabela 6.15 mostra os ganhos percentuais (do conjunto de teste) do modelo DELP proposto com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF.

É possível notar, de acordo com a Tabela 6.15, que o modelo proposto obteve um desempe-

Tabela 6.13 Resultados de todos os experimentos utilizando o modelo proposto para a série BBDC4 (conjunto de treinamento)

DELP ($\mu; \sigma$)	Estatística	Métricas					
		MSE	MAPE	THEIL	ARV	POCID	EF
(0,001; 0,005)	MEAN	1,1449e-006	3,0424e-003	5,3318e-003	1,3890e-004	100,00	99,1641
	STD	1,5523e-006	3,0608e-003	7,2288e-003	1,8832e-004	0,00	1,0154
	CI	$\pm 1,7911e-006$	$\pm 3,5316e-003$	$\pm 8,3407e-003$	$\pm 2,1728e-004$	$\pm 0,00$	$\pm 1,1716$
(0,001; 0,05)	MEAN	4,2303e-006	5,1997e-003	1,9700e-002	5,1319e-004	100,00	97,6691
	STD	8,2380e-006	6,6006e-003	3,8363e-002	9,9937e-004	0,00	4,1298
	CI	$\pm 9,5051e-006$	$\pm 7,6158e-003$	$\pm 4,4263e-002$	$\pm 1,1531e-003$	$\pm 0,00$	$\pm 4,7650$
(0,001; 0,5)	MEAN	5,1339e-006	6,2854e-003	2,3908e-002	6,2281e-004	100,00	97,1674
	STD	8,6126e-006	6,6713e-003	4,0107e-002	1,0448e-003	0,00	4,2558
	CI	$\pm 9,9373e-006$	$\pm 7,6974e-003$	$\pm 4,6276e-002$	$\pm 1,2055e-003$	$\pm 0,00$	$\pm 4,9103$
(0,01; 0,005)	MEAN	3,3935e-008	6,2901e-004	1,5803e-004	4,1167e-006	100,00	99,9210
	STD	2,9153e-008	3,5503e-004	1,3576e-004	3,5366e-006	0,00	0,0493
	CI	$\pm 3,3637e-008$	$\pm 4,0964e-004$	$\pm 1,5664e-004$	$\pm 4,0806e-006$	$\pm 0,00$	$\pm 0,0569$
(0,01; 0,05)	MEAN	1,1178e-007	1,0882e-003	5,2054e-004	1,3560e-005	100,00	99,8382
	STD	1,4452e-007	7,5104e-004	6,7301e-004	1,7532e-005	0,00	0,1427
	CI	$\pm 1,6675e-007$	$\pm 8,6656e-004$	$\pm 7,7653e-004$	$\pm 2,0229e-005$	$\pm 0,00$	$\pm 0,1647$
(0,01; 0,5)	MEAN	2,7994e-008	6,1936e-004	1,3036e-004	3,3960e-006	100,00	99,9247
	STD	1,7557e-008	1,8012e-004	8,1757e-005	2,1298e-006	0,00	0,0263
	CI	$\pm 2,0257e-008$	$\pm 2,0783e-004$	$\pm 9,4333e-005$	$\pm 2,4574e-006$	$\pm 0,00$	$\pm 0,0304$
(0,1; 0,005)	MEAN	1,8912e-006	5,2560e-003	8,8069e-003	2,2943e-004	100,00	98,5908
	STD	1,6867e-007	2,3386e-004	7,8548e-004	2,0462e-005	0,00	0,1011
	CI	$\pm 1,9462e-007$	$\pm 2,6983e-004$	$\pm 9,0630e-004$	$\pm 2,3610e-005$	$\pm 0,00$	$\pm 0,1166$
(0,1; 0,05)	MEAN	1,5790e-006	4,5022e-003	7,3532e-003	1,9156e-004	100,00	98,8123
	STD	8,6473e-007	1,8816e-003	4,0269e-003	1,0490e-004	0,00	0,5903
	CI	$\pm 9,9774e-007$	$\pm 2,1710e-003$	$\pm 4,6462e-003$	$\pm 1,2104e-004$	$\pm 0,00$	$\pm 0,6811$
(0,1; 0,5)	MEAN	1,6902e-006	4,8428e-003	7,8711e-003	2,0505e-004	100,00	98,7261
	STD	7,0737e-007	1,2630e-003	3,2941e-003	8,5812e-005	0,00	0,4547
	CI	$\pm 8,1617e-007$	$\pm 1,4573e-003$	$\pm 3,8007e-003$	$\pm 9,9011e-005$	$\pm 0,00$	$\pm 0,5246$

Tabela 6.14 Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série BBDC4.

Modelo	MSE	MAPE	THEIL	ARV	POCID	EF
ARIMA	2,9680e-004	5,1816e-002	1,3879e+000	3,5589e-002	49,40	19,9539
MLP	2,1635e-004	4,1128e-002	1,0071e+000	2,7329e-002	52,85	25,4588
MGA-MLP	2,2219e-004	4,2066e-002	1,0342e+000	2,8067e-002	54,47	25,8821
PSO-MLP	2,1556e-004	4,1212e-002	1,0034e+000	2,7230e-002	53,66	25,8963
TAEF	3,3265e-005	2,2351e-002	1,5434e-001	4,1106e-003	96,34	81,5874
SHIF	3,2884e-005	2,2227e-002	1,5257e-001	4,0635e-003	96,34	81,7218
DELP	1,3772e-008	4,4888e-004	6,4133e-005	1,6707e-006	100,00	99,9486

no preditivo bastante superior aos modelos MGA-MLP, PSO-MLP, MLP e ARIMA (obtendo um ganho percentual superior a 83% para todas as métricas) e superior aos modelos TAEF e SHIF (alcançando um ganho percentual superior a 97% para as métricas MSE, MAPE, THEIL e ARV, e em torno de 3% para a métrica POCID). Em relação ao desempenho preditivo global, verificamos que o modelo proposto obteve um ganho percentual superior a 285% em relação aos modelos ARIMA, MLP, MGA-MLP e PSO-MLP, e em torno de 22% em relação aos modelos TAEF e SHIF.

As Figuras 6.9 e 6.10 apresentam um gráfico comparativo entre os valores reais da série (linha sólida) e os valores preditos pelo modelo proposto (linha tracejada) para os últimos cem

Tabela 6.15 Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA–MLP, PSO–MLP, TAEF e SHIF para a série BBDC4.

	MSE	MAPE	THEIL	ARV	POCID	EF
Ganho (%) DELP / ARIMA	100,00	99,13	100,00	100,00	102,43	400,90
Ganho (%) DELP / MLP	99,99	98,91	99,99	99,99	89,21	292,59
Ganho (%) DELP / MGA–MLP	99,99	98,93	99,99	99,99	83,59	286,17
Ganho (%) DELP / PSO–MLP	99,99	98,91	99,99	99,99	86,36	285,96
Ganho (%) DELP / TAEF	99,96	97,99	99,96	99,96	3,80	22,50
Ganho (%) DELP / SHIF	99,96	97,98	99,96	99,96	3,80	22,30

e dez pontos, respectivamente, do conjunto de teste. Vale mencionar que a previsão está quase sobreposta aos valores reais da série, o que comprova que o característico atraso de um passo da previsão não ocorreu e o dilema do passei aleatório foi mais uma vez superado.

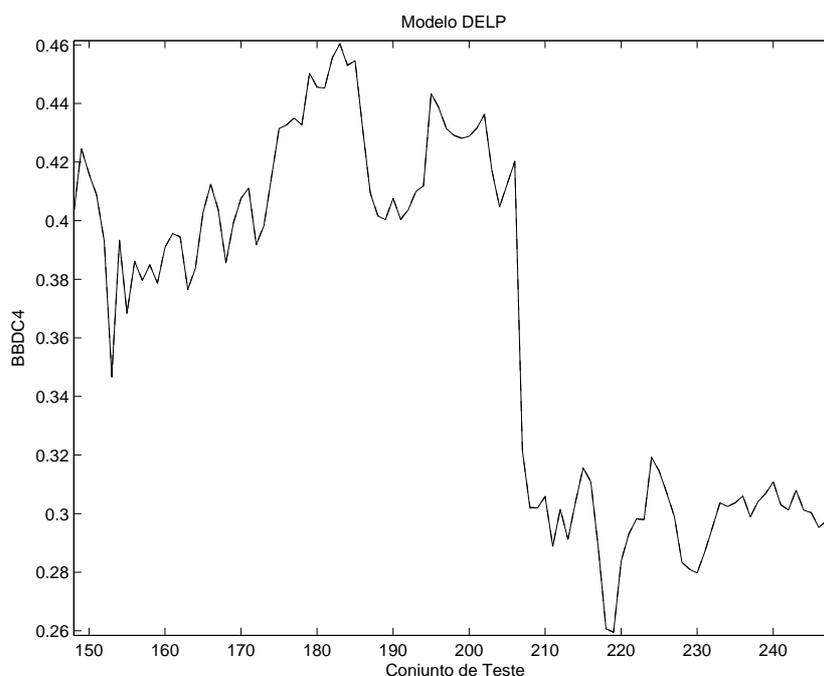


Figura 6.9 Previsão da série BBDC4 gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).

6.2.6 Preço das Ações da GOL PN (GOLL4)

Para a série do Preço das Ações da GOL PN (GOLL4), os modelos utilizados que obtiveram o melhor desempenho foram: i) modelo ARIMA com a configuração (2,1,1), ii) modelo MLP com a configuração 5-25-1 e arquitetura definida pela Equação (3.46), iii) o modelo MGA–MLP definiu os retardos temporais (1, 2, 3, 4, 7 e 10) e a rede MLP com a configuração 6-7-1 e arquitetura definida pela Equação (3.46), iv) o modelo PSO–MLP escolheu os retardos tempo-

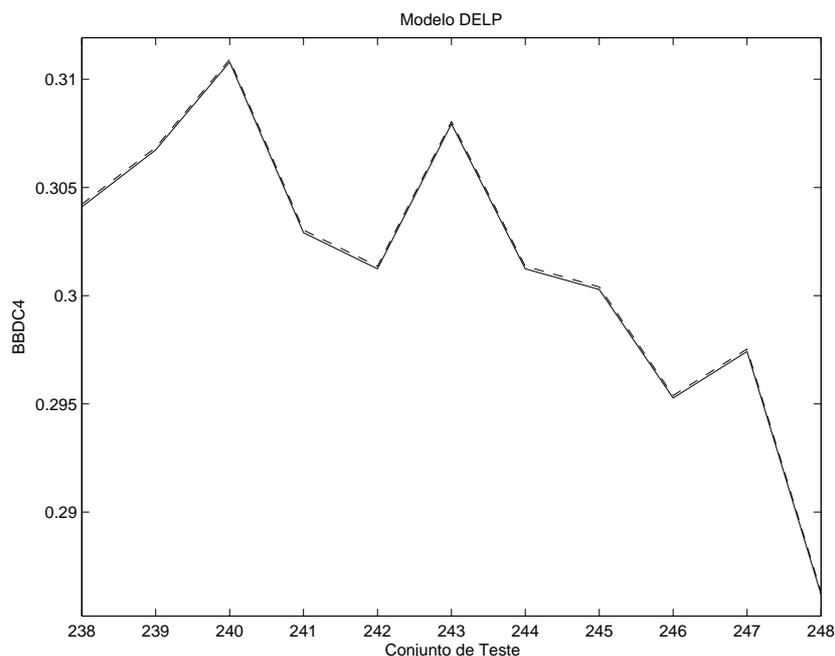


Figura 6.10 Previsão da série BBDC4 gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).

rais (1, 3, 4, 5 e 9) e a rede MLP com a configuração 5-2-1 e arquitetura definida pela Equação (3.47), v) o modelo TAEF determinou os retardos temporais (2, 3, 4 e 8) e a rede MLP com a configuração 4-10-1, arquitetura definida pela Equação (3.46) e o algoritmo *levenberg marquardt* para treinamento da rede MLP, e vi) o modelo SHIF determinou os retardos temporais (2, 4, 8, 9 e 10) e a rede MLP com a configuração 5-9-1, arquitetura definida pela Equação (3.44) e o algoritmo *levenberg marquardt* para treinamento da rede MLP.

A Tabela 6.16 apresenta todos os experimentos realizados com o modelo DELP (utilizando o conjunto de treinamento) variando o valor de μ e σ , onde foram calculadas todas métricas, bem como as estatísticas MEAN, STD e CI para análise do desempenho de previsão médio do modelo proposto.

De acordo com a Tabela 6.16, pode-se verificar que os melhores valores de μ e σ para a série GOLL4 foram 0,01 e 0,05, respectivamente, uma vez que obtivemos o melhor desempenho de previsão em termos da função de avaliação global (EF). Portanto, assumindo essa suposição, escolheu-se o modelo DELP(0,01; 0,05) com a maior métrica EF no conjunto de validação, dentre os dez modelos criados neste experimento, para representar o modelo proposto. Neste modelo, o termo de combinação λ determinado foi 0,0339, significando que o DELP utilizou 96,61% da componente linear e 3,39% da componente não-linear, confirmando a hipótese (através da análise do *lagplot* apresentada no Capítulo 4) que séries temporais financeiras são geradas por componentes lineares dominantes e componentes não-lineares sub-dominantes. Vale mencionar que o termo de combinação da componente não-linear θ foi 0,4926, isto é, utilização de 49,26% do operador de dilatação e 50,74% do operador de erosão. A Tabela 6.17 apresenta os resultados (referentes ao conjunto de teste) com todas as métricas de desempenho

Tabela 6.16 Resultados de todos os experimentos utilizando o modelo proposto para a série GOLL4 (conjunto de treinamento)

DELPH ($\mu; \sigma$)	Estatística	Métricas					
		MSE	MAPE	THEIL	ARV	POCID	EF
(0,001; 0,005)	MEAN	1,2703e-006	2,7506e-002	2,4471e-002	1,6813e-004	100,00	95,3245
	STD	2,1443e-006	2,4874e-002	4,1309e-002	2,8382e-004	0,00	5,5664
	CI	$\pm 2,4741e-006$	$\pm 2,8700e-002$	$\pm 4,7663e-002$	$\pm 3,2748e-004$	$\pm 0,00$	$\pm 6,4226$
(0,001; 0,05)	MEAN	6,6074e-006	7,1243e-002	1,2729e-001	8,7455e-004	100,00	84,6604
	STD	6,6013e-006	4,2345e-002	1,2717e-001	8,7374e-004	0,00	11,3900
	CI	$\pm 7,6167e-006$	$\pm 4,8858e-002$	$\pm 1,4673e-001$	$\pm 1,0081e-003$	$\pm 0,00$	$\pm 13,1419$
(0,001; 0,5)	MEAN	2,2058e-005	1,1274e-001	4,2493e-001	2,9196e-003	100,00	73,0567
	STD	2,6786e-005	1,0620e-001	5,1601e-001	3,5453e-003	0,00	26,0863
	CI	$\pm 3,0906e-005$	$\pm 1,2253e-001$	$\pm 5,9537e-001$	$\pm 4,0906e-003$	$\pm 0,00$	$\pm 30,0986$
(0,01; 0,005)	MEAN	3,3175e-007	1,7892e-002	6,3910e-003	4,3910e-005	100,00	97,6269
	STD	1,0286e-007	2,9055e-003	1,9816e-003	1,3615e-005	0,00	0,4678
	CI	$\pm 1,1869e-007$	$\pm 3,3524e-003$	$\pm 2,2864e-003$	$\pm 1,5709e-005$	$\pm 0,00$	$\pm 0,5397$
(0,01; 0,05)	MEAN	3,2118e-007	1,6297e-002	6,1873e-003	4,2511e-005	100,00	97,8103
	STD	2,7859e-007	7,9743e-003	5,3668e-003	3,6874e-005	0,00	1,2738
	CI	$\pm 3,2144e-007$	$\pm 9,2009e-003$	$\pm 6,1923e-003$	$\pm 4,2545e-005$	$\pm 0,00$	$\pm 1,4698$
(0,01; 0,5)	MEAN	3,1229e-007	1,7296e-002	6,0161e-003	4,1335e-005	100,00	97,7201
	STD	1,0961e-007	3,2698e-003	2,1116e-003	1,4508e-005	0,00	0,5164
	CI	$\pm 1,2647e-007$	$\pm 3,7727e-003$	$\pm 2,4364e-003$	$\pm 1,6740e-005$	$\pm 0,00$	$\pm 0,5958$
(0,1; 0,005)	MEAN	1,1689e-006	3,0710e-002	2,2518e-002	1,5471e-004	100,00	95,0136
	STD	9,4895e-007	1,6147e-002	1,8281e-002	1,2560e-004	0,00	3,1163
	CI	$\pm 1,0949e-006$	$\pm 1,8630e-002$	$\pm 2,1093e-002$	$\pm 1,4492e-004$	$\pm 0,00$	$\pm 3,5956$
(0,1; 0,05)	MEAN	9,6870e-007	2,1729e-002	1,8661e-002	1,2822e-004	100,00	96,2799
	STD	1,3146e-006	2,4553e-002	2,5324e-002	1,7400e-004	0,00	4,5327
	CI	$\pm 1,5168e-006$	$\pm 2,8330e-002$	$\pm 2,9219e-002$	$\pm 2,0076e-004$	$\pm 0,00$	$\pm 5,2299$
(0,1; 0,5)	MEAN	1,3934e-006	3,4578e-002	2,6842e-002	1,8442e-004	100,00	94,2841
	STD	1,1060e-006	1,4881e-002	2,1306e-002	1,4638e-004	0,00	3,1887
	CI	$\pm 1,2761e-006$	$\pm 1,7170e-002$	$\pm 2,4583e-002$	$\pm 1,6890e-004$	$\pm 0,00$	$\pm 3,6792$

para os modelos ARIMA, MLP, MGA–MLP, PSO–MLP, TAEF, SHIF e DELP.

Tabela 6.17 Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA–MLP, PSO–MLP, TAEF, SHIF e DELP) para a série GOLL4.

	MSE	MAPE	THEIL	ARV	POCID	EF
ARIMA	6,4697e-005	2,5908e-001	1,2478e+000	8,6500e-003	50,81	20,1991
MLP	6,5549e-005	1,1030e-001	1,2643e+000	8,7639e-003	50,81	21,3197
MGA–MLP	7,2319e-005	1,0395e-001	1,4038e+000	9,5511e-003	53,82	21,3776
PSO–MLP	6,1195e-005	1,9343e-001	1,1803e+000	8,1819e-003	51,63	21,6737
TAEF	4,2795e-005	1,8234e-001	8,2207e-001	5,7110e-003	93,88	46,7013
SHIF	2,2095e-005	1,3794e-001	4,2443e-001	2,9486e-003	93,88	59,9724
DELP	6,5466e-008	8,0313e-003	1,2612e-003	8,6650e-006	100,00	99,0785

Os resultados obtidos nos experimentos realizados com a série GOLL4 mostram que o desempenho preditivo do modelo DELP para todas as métricas investigadas é superior aos resultados encontrados na literatura. É possível verificar que a métrica THEIL obtida pelo modelo proposto foi bem próxima de zero, mostrando que o dilema do passeio aleatório foi novamente superado. A Tabela 6.18 mostra os ganhos percentuais (do conjunto de teste) do modelo DELP proposto com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA–MLP, PSO–MLP, TAEF e SHIF.

Tabela 6.18 Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA–MLP, PSO–MLP, TAEF e SHIF para a série GOLL4.

	MSE	MAPE	THEIL	ARV	POCID	EF
Ganho (%) DELP / ARIMA	99,90	96,90	99,90	99,90	96,81	390,51
Ganho (%) DELP / MLP	99,90	92,72	99,90	99,90	96,81	364,73
Ganho (%) DELP / MGA–MLP	99,91	92,27	99,91	99,91	85,80	363,47
Ganho (%) DELP / PSO–MLP	99,89	95,85	99,89	99,89	93,69	357,14
Ganho (%) DELP / TAEF	99,85	95,60	99,85	99,85	6,52	112,15
Ganho (%) DELP / SHIF	99,70	94,18	99,70	99,71	6,52	65,21

Pode-se verificar, a partir da Tabela 6.18, que o modelo proposto obteve um desempenho preditivo bastante superior aos resultados reportados na literatura (obtendo um ganho percentual superior a 85% para todas as métricas, exceto a métrica POCID nos modelos TAEF e SHIF, obtendo um ganho percentual em torno de 6%). Avaliando o modelo proposto em termos de desempenho preditivo global, pode-se verificar que obtivemos um ganho percentual superior a 357% em relação aos modelos ARIMA, MLP, MGA–MLP e PSO–MLP, e em torno de 112% e 65% em relação aos modelos TAEF e SHIF, respectivamente.

As Figuras 6.11 e 6.12 apresentam um gráfico comparativo entre os valores reais da série (linha sólida) e os valores preditos pelo modelo proposto (linha tracejada) para os últimos cem e dez pontos, respectivamente, do conjunto de teste. Vale mencionar que a previsão está bastante próxima aos valores reais da série, o que comprova que o característico atraso de um passo da previsão não ocorreu e o dilema do passei aleatório foi mais uma vez superado.

6.2.7 Preço das Ações do Itaú Unibanco PN (ITUB4)

Para a série do Preço das Ações do Itaú Unibanco PN (ITUB4), os modelos utilizados que obtiveram o melhor desempenho foram: i) modelo ARIMA com a configuração (2,1,1), ii) modelo MLP com a configuração 1-10-1 e arquitetura definida pela Equação (3.44), iii) o modelo MGA–MLP definiu os retardos temporais (1, 2, 3, 5, 6 e 7) e a rede MLP com a configuração 6-8-1 e arquitetura definida pela Equação (3.46), iv) o modelo PSO–MLP escolheu os retardos temporais (1, 3, 4, 6, 8, e 10) e a rede MLP com a configuração 6-4-1 e arquitetura definida pela Equação (3.46), v) o modelo TAEF determinou os retardos temporais (5, 8 e 9) e a rede MLP com a configuração 3-6-1, arquitetura definida pela Equação (3.46) e o algoritmo *one step secant conjugate gradient* para treinamento da rede MLP, e vi) o modelo SHIF determinou os retardos temporais (2, 5, 7, 8 e 9) e a rede MLP com a configuração 5-10-1, arquitetura definida pela Equação (3.44) e o algoritmo *resilient backpropagation* para treinamento da rede MLP.

A Tabela 6.19 apresenta todos os experimentos realizados com o modelo DELP (utilizando o conjunto de treinamento) variando o valor de μ e σ , onde foram calculadas todas métricas, bem como as estatísticas MEAN, STD e CI para análise do desempenho de previsão médio do modelo proposto.

De acordo com a Tabela 6.19, pode-se verificar que os melhores valores de μ e σ para a série ITUB4 foram 0, 1 e 0,005, respectivamente, uma vez que obtivemos o melhor desempenho de previsão em termos da função de avaliação global (EF). Portanto, assumindo essa suposi-

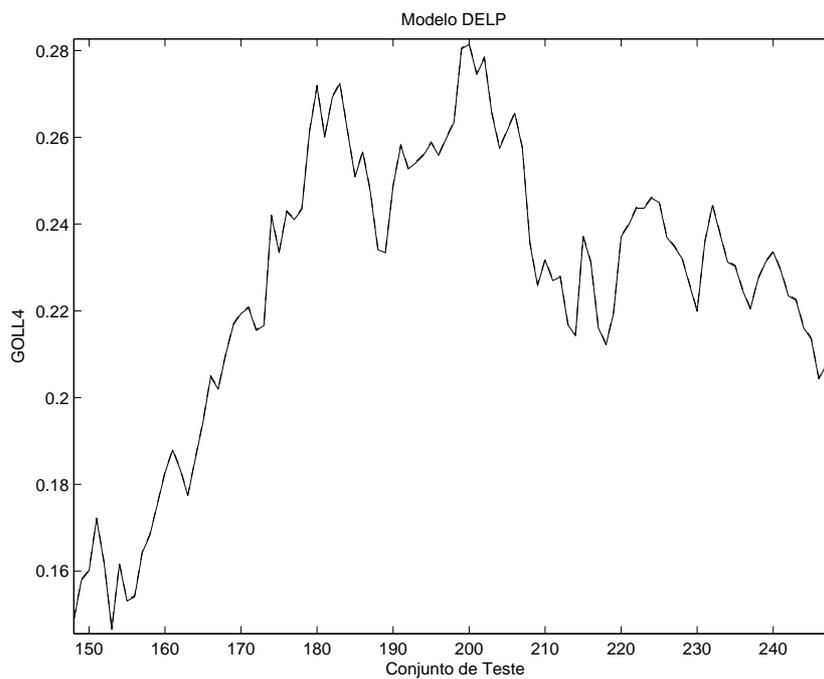


Figura 6.11 Previsão da série GOLL4 gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).

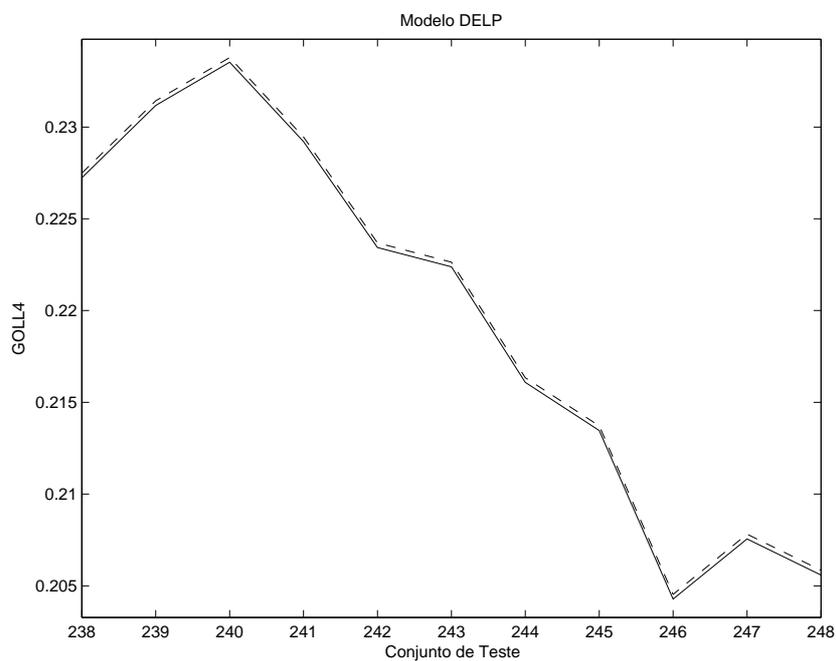


Figura 6.12 Previsão da série GOLL4 gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).

Tabela 6.19 Resultados de todos os experimentos utilizando o modelo proposto para a série ITUB4 (conjunto de treinamento)

DELP ($\mu; \sigma$)	Estatística	Métricas					
		MSE	MAPE	THEIL	ARV	POCID	EF
(0,001; 0,005)	MEAN	3,9208e-006	3,2156e-003	8,5041e-003	3,0497e-004	100,00	98,8338
	STD	6,2542e-006	3,2028e-003	1,3565e-002	4,8647e-004	0,00	1,6484
	CI	$\pm 7,2162e-006$	$\pm 3,6954e-003$	$\pm 1,5652e-002$	$\pm 5,6129e-004$	$\pm 0,00$	$\pm 1,9019$
(0,001; 0,05)	MEAN	8,3634e-006	5,5281e-003	1,8140e-002	6,5052e-004	100,00	97,6596
	STD	8,1967e-006	3,3544e-003	1,7779e-002	6,3756e-004	0,00	2,0395
	CI	$\pm 9,4575e-006$	$\pm 3,8704e-003$	$\pm 2,0513e-002$	$\pm 7,3563e-004$	$\pm 0,00$	$\pm 2,3532$
(0,001; 0,5)	MEAN	5,0859e-006	4,6122e-003	1,1031e-002	3,9559e-004	100,00	98,4286
	STD	3,6538e-006	1,8661e-003	7,9251e-003	2,8420e-004	0,00	0,9700
	CI	$\pm 4,2158e-006$	$\pm 2,1531e-003$	$\pm 9,1440e-003$	$\pm 3,2791e-004$	$\pm 0,00$	$\pm 1,1192$
(0,01; 0,005)	MEAN	1,1487e-006	2,3065e-003	2,4915e-003	8,9348e-005	100,00	99,5136
	STD	3,5991e-007	3,7735e-004	7,8063e-004	2,7994e-005	0,00	0,1175
	CI	$\pm 4,1526e-007$	$\pm 4,3539e-004$	$\pm 9,0070e-004$	$\pm 3,2300e-005$	$\pm 0,00$	$\pm 0,1355$
(0,01; 0,05)	MEAN	9,9413e-007	2,1451e-003	2,1563e-003	7,7326e-005	100,00	99,5640
	STD	2,9393e-007	3,5560e-004	6,3753e-004	2,2862e-005	0,00	0,1008
	CI	$\pm 3,3914e-007$	$\pm 4,1029e-004$	$\pm 7,3559e-004$	$\pm 2,6379e-005$	$\pm 0,00$	$\pm 0,1163$
(0,01; 0,5)	MEAN	2,7135e-006	3,1759e-003	5,8856e-003	2,1106e-004	100,00	99,0875
	STD	3,2643e-006	1,8538e-003	7,0803e-003	2,5391e-004	0,00	0,8912
	CI	$\pm 3,7664e-006$	$\pm 2,1389e-003$	$\pm 8,1693e-003$	$\pm 2,9296e-004$	$\pm 0,00$	$\pm 1,0283$
(0,1; 0,005)	MEAN	3,3305e-008	3,3585e-004	7,2239e-005	2,5906e-006	100,00	99,9590
	STD	3,0473e-008	2,3650e-004	6,6097e-005	2,3703e-006	0,00	0,0304
	CI	$\pm 3,5161e-008$	$\pm 2,7288e-004$	$\pm 7,6263e-005$	$\pm 2,7349e-006$	$\pm 0,00$	$\pm 0,0351$
(0,1; 0,05)	MEAN	1,3528e-007	4,4739e-004	2,9342e-004	1,0522e-005	100,00	99,9251
	STD	2,8879e-007	7,4142e-004	6,2638e-004	2,2463e-005	0,00	0,1383
	CI	$\pm 3,3321e-007$	$\pm 8,5546e-004$	$\pm 7,2273e-004$	$\pm 2,5918e-005$	$\pm 0,00$	$\pm 0,1596$
(0,1; 0,5)	MEAN	5,0925e-007	8,4248e-004	1,1046e-003	3,9611e-005	100,00	99,8029
	STD	1,1143e-006	1,4574e-003	2,4170e-003	8,6677e-005	0,00	0,3923
	CI	$\pm 1,2857e-006$	$\pm 1,6815e-003$	$\pm 2,7888e-003$	$\pm 1,0001e-004$	$\pm 0,00$	$\pm 0,4527$

ção, escolheu-se o modelo DELP(0,1; 0,005) com a maior métrica EF no conjunto de validação, dentre os dez modelos criados neste experimento, para representar o modelo proposto. Neste modelo, o termo de combinação λ determinado foi 0,0196, significando que o DELP utilizou 98,04% da componente linear e 1,96% da componente não-linear, confirmando a hipótese (através da análise do *lagplot* apresentada no Capítulo 4) que séries temporais financeiras são geradas por componentes lineares dominantes e componentes não-lineares sub-dominantes. Vale mencionar que o termo de combinação da componente não-linear θ foi 0,5983, isto é, utilização de 59,83% do operador de dilatação e 40,17% do operador de erosão. A Tabela 6.20 apresenta os resultados (referentes ao conjunto de teste) com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP.

Os resultados obtidos nos experimentos realizados com a série ITUB4 mostram que o desempenho preditivo do modelo DELP para todas as métricas investigadas é superior aos resultados encontrados na literatura. É possível verificar que a métrica THEIL obtida pelo modelo proposto foi bem próxima de zero, mostrando que o dilema do passeio aleatório foi superado. A Tabela 6.21 mostra os ganhos percentuais (do conjunto de teste) do modelo DELP proposto com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF.

De acordo com a Tabela 6.21, pode-se verificar que o modelo proposto obteve desempenho preditivo bastante superior aos resultados reportados na literatura (obtendo um ganho percentual superior a 76% para todas as métricas, exceto a métrica POCID nos modelos TAEF e

Tabela 6.20 Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série ITUB4.

	MSE	MAPE	THEIL	ARV	POCID	EF
ARIMA	7,6529e-004	4,8775e-002	1,6675e+000	5,8657e-002	55,42	19,9665
MLP	4,5796e-004	3,4674e-002	1,0349e+000	3,7220e-002	54,88	26,0428
MGA-MLP	4,6689e-004	3,5529e-002	1,0550e+000	3,7946e-002	56,50	26,5403
PSO-MLP	4,5972e-004	3,5410e-002	1,0389e+000	3,7363e-002	55,28	26,1753
TAEF	1,9221e-005	7,8527e-003	4,1563e-002	1,5251e-003	96,75	92,0568
SHIF	1,9248e-005	7,8564e-003	4,1623e-002	1,5273e-003	96,75	92,0510
DELP	1,5322e-009	8,5133e-005	3,3233e-006	1,1918e-007	100,00	99,9911

Tabela 6.21 Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série ITUB4.

	MSE	MAPE	THEIL	ARV	POCID	EF
Ganho (%) DELP / ARIMA	100,00	99,83	100,00	100,00	80,44	400,79
Ganho (%) DELP / MLP	100,00	99,75	100,00	100,00	82,22	283,95
Ganho (%) DELP / MGA-MLP	100,00	99,76	100,00	100,00	76,99	276,75
Ganho (%) DELP / PSO-MLP	100,00	99,76	100,00	100,00	80,90	282,01
Ganho (%) DELP / TAEF	99,99	98,92	99,99	99,99	3,36	8,62
Ganho (%) DELP / SHIF	99,99	98,92	99,99	99,99	3,36	8,63

SHIF, que obteve um ganho percentual em torno de 3%). Em termos globais, pode-se verificar que o modelo proposto obteve um ganho percentual superior a 276% em relação aos modelos ARIMA, MLP, MGA-MLP e PSO-MLP, e em torno de 8% em relação aos modelos TAEF e SHIF.

As Figuras 6.13 e 6.14 apresentam um gráfico comparativo entre os valores reais da série (linha sólida) e os valores preditos pelo modelo proposto (linha tracejada) para os últimos cem e dez pontos, respectivamente, do conjunto de teste. Vale mencionar que a previsão está sobreposta aos valores reais da série, o que comprova que o característico atraso de um passo da previsão não ocorreu e o dilema do passei aleatório foi mais uma vez superado.

6.2.8 Preço das Ações da Petrobras PN (PETR4)

Para a série do Preço das Ações da Petrobras PN (PETR4), os modelos utilizados que obtiveram o melhor desempenho foram: i) modelo ARIMA com a configuração (2,1,1), ii) modelo MLP com a configuração 10-5-1 e arquitetura definida pela Equação (3.46), iii) o modelo MGA-MLP definiu os retardos temporais (1, 2, 3, 4, 5, 6, 8 e 10) e a rede MLP com a configuração 8-4-1 e arquitetura definida pela Equação (3.47), iv) o modelo PSO-MLP escolheu os retardos temporais (1, 7 e 8) e a rede MLP com a configuração 3-8-1 e arquitetura definida pela Equação (3.44), v) o modelo TAEF determinou os retardos temporais (2, 3, 4, 5, 6, 9 e 10) e a rede MLP com a configuração 7-6-1, arquitetura definida pela Equação (3.46) e o algoritmo *levenberg marquardt* para treinamento da rede MLP, e vi) o modelo SHIF determinou os retardos tempo-

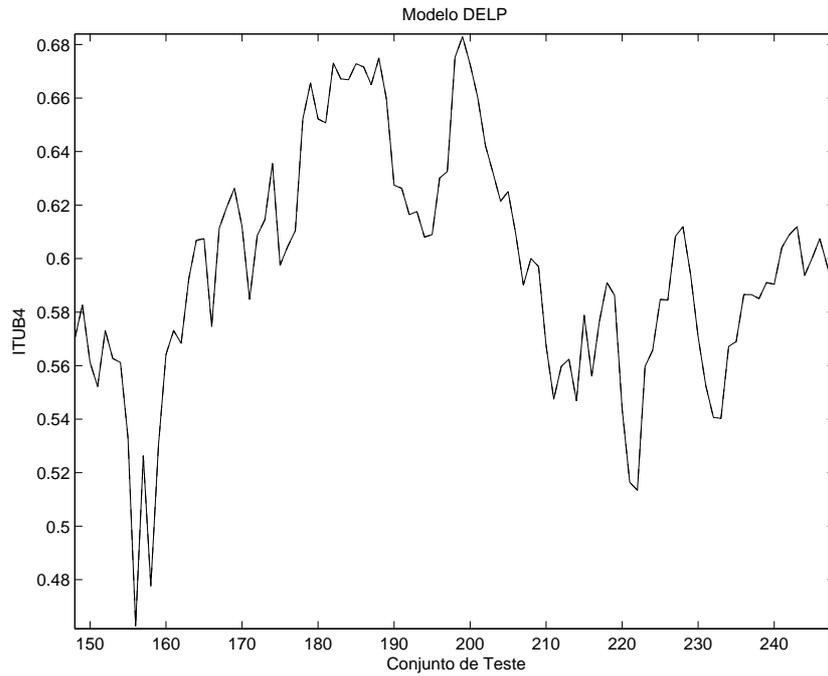


Figura 6.13 Previsão da série ITUB4 gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).

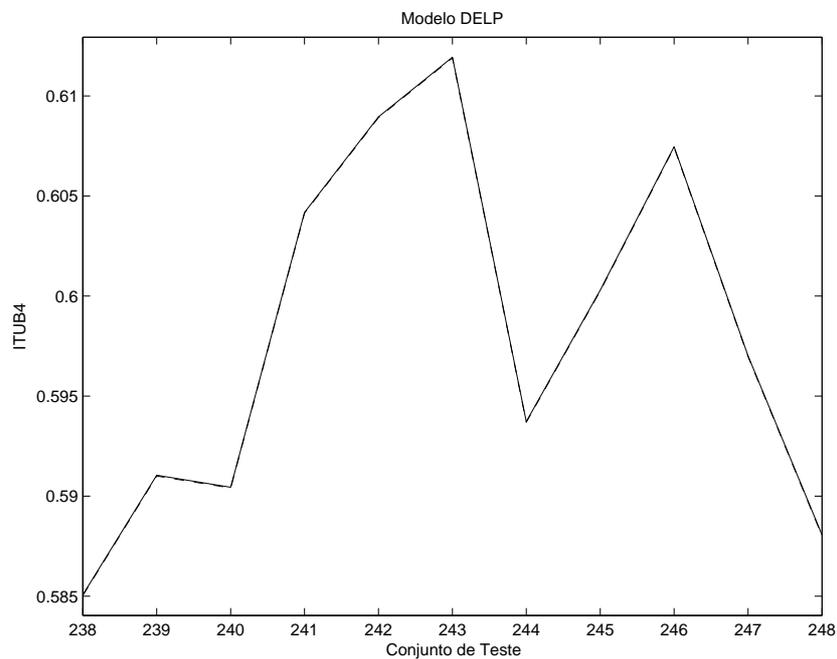


Figura 6.14 Previsão da série ITUB4 gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).

rais (2, 3, 4, 5, 6, 7, 8 e 9) e a rede MLP com a configuração 8-8-1, arquitetura definida pela Equação (3.44) e o algoritmo *resilient backpropagation* para treinamento da rede MLP.

A Tabela 6.22 apresenta todos os experimentos realizados com o modelo DELP (utilizando o conjunto de treinamento) variando o valor de μ e σ , onde foram calculadas todas métricas, bem como as estatísticas MEAN, STD e CI para análise do desempenho de previsão médio do modelo proposto.

Tabela 6.22 Resultados de todos os experimentos utilizando o modelo proposto para a série PETR4 (conjunto de treinamento)

DELP ($\mu; \sigma$)	Estatística	Métricas					
		MSE	MAPE	THEIL	ARV	POCID	EF
(0,001; 0,005)	MEAN	1,6446e-006	4,2372e-003	2,4013e-002	1,1584e-003	100,00	97,2418
	STD	2,1645e-006	4,0681e-003	3,1603e-002	1,5245e-003	0,00	3,4278
	CI	$\pm 2,4974e-006$	$\pm 4,6938e-003$	$\pm 3,6464e-002$	$\pm 1,7590e-003$	$\pm 0,00$	$\pm 3,9551$
(0,001; 0,05)	MEAN	9,6455e-006	8,4431e-003	1,4083e-001	6,7938e-003	100,00	90,4170
	STD	1,9748e-005	1,1814e-002	2,8833e-001	1,3909e-002	0,00	18,0423
	CI	$\pm 2,2785e-005$	$\pm 1,3631e-002$	$\pm 3,3268e-001$	$\pm 1,6049e-002$	$\pm 0,00$	$\pm 20,8174$
(0,001; 0,5)	MEAN	1,4881e-006	5,2570e-003	2,1727e-002	1,0481e-003	100,00	97,2777
	STD	4,6025e-007	8,5705e-004	6,7199e-003	3,2417e-004	0,00	0,7488
	CI	$\pm 5,3104e-007$	$\pm 9,8887e-004$	$\pm 7,7535e-003$	$\pm 3,7403e-004$	$\pm 0,00$	$\pm 0,8639$
(0,01; 0,005)	MEAN	8,6434e-008	1,2480e-003	1,2620e-003	6,0879e-005	100,00	99,7436
	STD	4,5778e-008	3,1983e-004	6,6839e-004	3,2243e-005	0,00	0,1014
	CI	$\pm 5,2819e-008$	$\pm 3,6902e-004$	$\pm 7,7120e-004$	$\pm 3,7203e-005$	$\pm 0,00$	$\pm 0,1170$
(0,01; 0,05)	MEAN	1,0432e-007	1,3632e-003	1,5231e-003	7,3475e-005	100,00	99,7050
	STD	5,8562e-008	3,8787e-004	8,5505e-004	4,1248e-005	0,00	0,1274
	CI	$\pm 6,7569e-008$	$\pm 4,4753e-004$	$\pm 9,8656e-004$	$\pm 4,7592e-005$	$\pm 0,00$	$\pm 0,1470$
(0,01; 0,5)	MEAN	8,8929e-008	1,2689e-003	1,2984e-003	6,2637e-005	100,00	99,7378
	STD	4,3263e-008	3,0935e-004	6,3167e-004	3,0472e-005	0,00	0,0966
	CI	$\pm 4,9917e-008$	$\pm 3,5693e-004$	$\pm 7,2883e-004$	$\pm 3,5159e-005$	$\pm 0,00$	$\pm 0,1114$
(0,1; 0,005)	MEAN	1,0270e-008	4,0115e-004	1,4995e-004	7,2336e-006	100,00	99,9442
	STD	6,3713e-009	2,0575e-004	9,3025e-005	4,4876e-006	0,00	0,0302
	CI	$\pm 7,3512e-009$	$\pm 2,3740e-004$	$\pm 1,0733e-004$	$\pm 5,1778e-006$	$\pm 0,00$	$\pm 0,0348$
(0,1; 0,05)	MEAN	8,6850e-009	3,6883e-004	1,2681e-004	6,1172e-006	100,00	99,9499
	STD	5,3884e-009	1,8936e-004	7,8675e-005	3,7953e-006	0,00	0,0271
	CI	$\pm 6,2172e-009$	$\pm 2,1849e-004$	$\pm 9,0776e-005$	$\pm 4,3791e-006$	$\pm 0,00$	$\pm 0,0312$
(0,1; 0,5)	MEAN	3,5589e-008	7,7914e-004	5,1963e-004	2,5067e-005	100,00	99,8678
	STD	2,9453e-008	2,9141e-004	4,3003e-004	2,0745e-005	0,00	0,0739
	CI	$\pm 3,3983e-008$	$\pm 3,3624e-004$	$\pm 4,9618e-004$	$\pm 2,3936e-005$	$\pm 0,00$	$\pm 0,0853$

De acordo com a Tabela 6.22, pode-se verificar que os melhores valores de μ e σ para a série PETR4 foram 0,1 e 0,05, respectivamente, uma vez que obtivemos o melhor desempenho de previsão em termos da função de avaliação global (EF). Portanto, assumindo essa suposição, escolheu-se o modelo DELP(0,1; 0,05) com a maior métrica EF no conjunto de validação, dentre os dez modelos criados neste experimento, para representar o modelo proposto. Neste modelo, o termo de combinação λ determinado foi 0,0147, significando que o DELP utilizou 98,53% da componente linear e 1,47% da componente não-linear, confirmando a hipótese (através da análise do *lagplot* apresentada no Capítulo 4) que séries temporais financeiras são geradas por componentes lineares dominantes e componentes não-lineares sub-dominantes. Vale mencionar que o termo de combinação da componente não-linear θ foi 0,5619, isto é, utilização de 56,19% do operador de dilatação e 43,81% do operador de erosão. A Tabela 6.23 apresenta os resultados (referentes ao conjunto de teste) com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP.

Os resultados obtidos nos experimentos realizados com a série PETR4, ilustrados na Tabela

Tabela 6.23 Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série PETR4.

	MSE	MAPE	THEIL	ARV	POCID	EF
ARIMA	9,2550e-005	3,3043e-002	1,3952e+000	6,6391e-002	50,41	20,2054
MLP	1,0152e-004	3,6045e-002	1,4776e+000	7,0669e-002	53,41	20,6678
MGA-MLP	7,2165e-005	2,8422e-002	1,0879e+000	5,1768e-002	50,81	23,4362
PSO-MLP	7,2304e-005	2,8364e-002	1,0900e+000	5,1867e-002	51,22	23,6005
TAEF	5,6890e-006	9,3961e-003	8,5488e-002	4,0756e-003	95,10	86,5378
SHIF	2,9687e-006	6,4348e-003	4,4610e-002	2,1267e-003	95,10	90,3004
DELP	7,1251e-009	3,6761e-004	1,0403e-004	5,0185e-006	100,00	99,9524

6.23, mostram que o desempenho preditivo do modelo DELP para todas as métricas investigadas é superior aos resultados encontrados na literatura. Note que a métrica THEIL obtida pelo modelo proposto foi bem próxima de zero, mostrando que o dilema do passeio aleatório foi mis uma vez superado. A Tabela 6.24 mostra os ganhos percentuais (do conjunto de teste) do modelo DELP proposto com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF.

Tabela 6.24 Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série PETR4.

	MSE	MAPE	THEIL	ARV	POCID	EF
Ganho (%) DELP / ARIMA	99,99	98,89	99,99	99,99	98,37	394,68
Ganho (%) DELP / MLP	99,99	98,98	99,99	99,99	87,23	383,61
Ganho (%) DELP / MGA-MLP	99,99	98,71	99,99	99,99	96,81	326,49
Ganho (%) DELP / PSO-MLP	99,99	98,70	99,99	99,99	95,24	323,52
Ganho (%) DELP / TAEF	99,87	96,09	99,88	99,88	5,15	15,50
Ganho (%) DELP / SHIF	99,76	94,29	99,77	99,76	5,15	10,69

Pode-se verificar, de acordo com a Tabela 6.24, que o modelo proposto obteve desempenho preditivo bastante superior aos resultados reportados na literatura (obtendo um ganho percentual superior a 87% para todas as métricas, exceto a métrica POCID nos modelos TAEF e SHIF, obtendo um ganho percentual em torno de 5%). Também, pode-se verificar, em termos globais, que o modelo proposto obteve um ganho percentual superior a 323% em relação aos modelos ARIMA, MLP, MGA-MLP e PSO-MLP, e superior a 10% em relação aos modelos TAEF e SHIF.

As Figuras 6.15 e 6.16 apresentam um gráfico comparativo entre os valores reais da série (linha sólida) e os valores preditos pelo modelo proposto (linha tracejada) para os últimos cem e dez pontos, respectivamente, do conjunto de teste. Vale mencionar que a previsão está sobreposta aos valores reais da série, o que comprova que o característico atraso de um passo da previsão não ocorreu e o dilema do passeio aleatório foi mais uma vez superado.

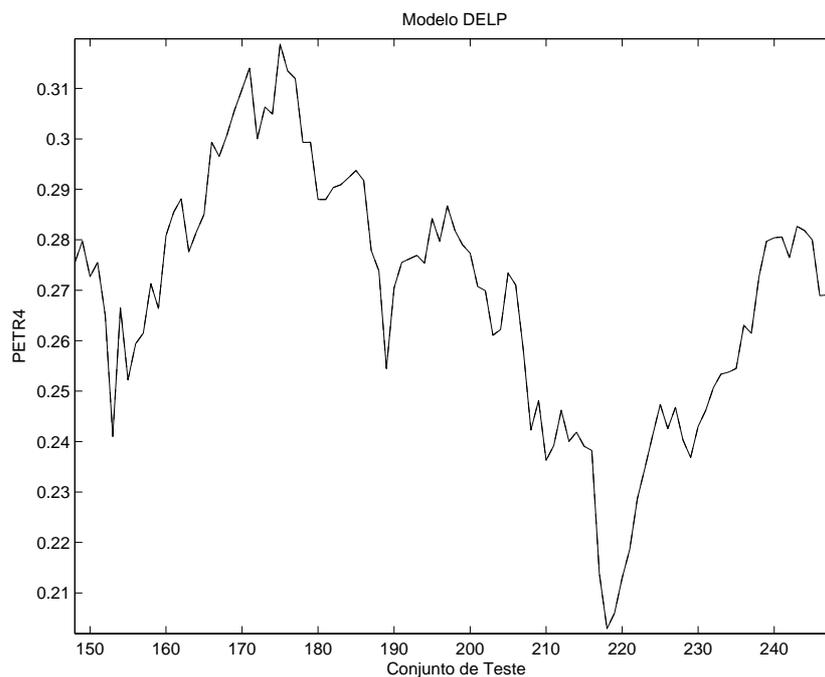


Figura 6.15 Previsão da série PETR4 gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).

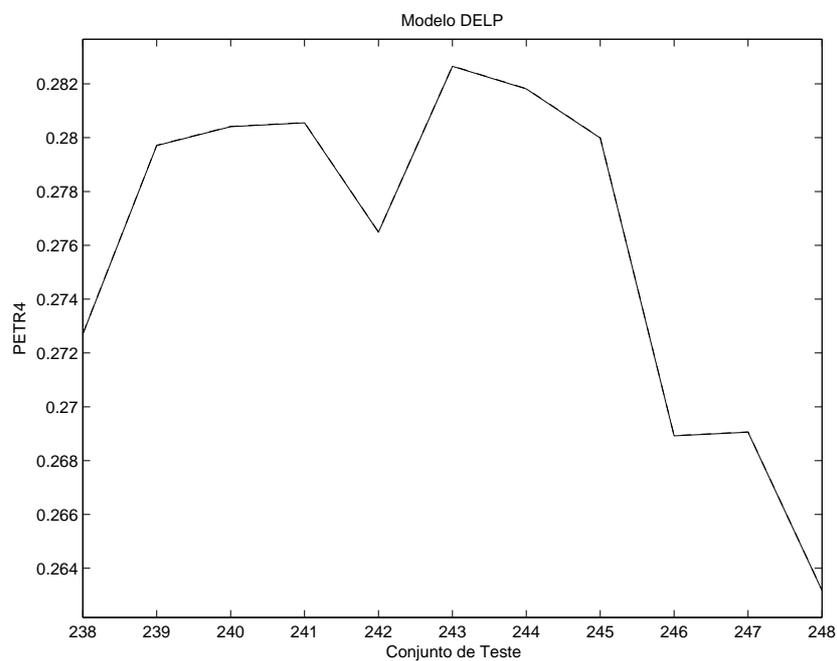


Figura 6.16 Previsão da série PETR4 gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).

6.2.9 Preço das Ações da Usiminas PNA (USIM5)

Para a série do Preço das Ações da Usiminas PNA (USIM5), os modelos utilizados que obtiveram o melhor desempenho foram: i) modelo ARIMA com a configuração (2,1,1), ii) modelo MLP com a configuração 1 – 5 – 1 e arquitetura definida pela Equação (3.46), iii) o modelo MGA–MLP definiu os retardos temporais (1, 3, 4, 6 e 7) e a rede MLP com a configuração 5-1-1 e arquitetura definida pela Equação (3.46), iv) o modelo PSO–MLP escolheu os retardos temporais (1, 3, 5, 6 e 8) e a rede MLP com a configuração 5-2-1 e arquitetura definida pela Equação (3.46), v) o modelo TAEF determinou os retardos temporais (2, 6 e 8) e a rede MLP com a configuração 3-2-1, arquitetura definida pela Equação (3.46) e o algoritmo *levenberg marquardt* para treinamento da rede MLP, e vi) o modelo SHIF determinou os retardos temporais (2, 5, 6 e 7) e a rede MLP com a configuração 4-1-1, arquitetura definida pela Equação (3.46) e o algoritmo *one step secant conjugate gradient* para treinamento da rede MLP.

A Tabela 6.25 apresenta todos os experimentos realizados com o modelo DELP (utilizando o conjunto de treinamento) variando o valor de μ e σ , onde foram calculadas todas métricas, bem como as estatísticas MEAN, STD e CI para análise do desempenho de previsão médio do modelo proposto.

Tabela 6.25 Resultados de todos os experimentos utilizando o modelo proposto para a série USIM5 (conjunto de treinamento)

DELP ($\mu; \sigma$)	Estatística	Métricas					
		MSE	MAPE	THEIL	ARV	POCID	EF
(0,001; 0,005)	MEAN	5,3444e-006	1,0011e-002	7,1434e-002	1,4158e-003	100,00	93,1082
	STD	7,9335e-006	8,2717e-003	1,0604e-001	2,1017e-003	0,00	8,8673
	CI	$\pm 9,1538e-006$	$\pm 9,5440e-003$	$\pm 1,2235e-001$	$\pm 2,4250e-003$	$\pm 0,00$	$\pm 10,2312$
(0,001; 0,05)	MEAN	2,9464e-006	7,3334e-003	3,9382e-002	7,8056e-004	100,00	95,6997
	STD	3,8642e-006	6,2894e-003	5,1650e-002	1,0237e-003	0,00	5,2073
	CI	$\pm 4,4586e-006$	$\pm 7,2568e-003$	$\pm 5,9594e-002$	$\pm 1,1812e-003$	$\pm 0,00$	$\pm 6,0082$
(0,001; 0,5)	MEAN	7,4886e-006	1,3238e-002	1,0009e-001	1,9838e-003	100,00	90,1293
	STD	5,9831e-006	7,2347e-003	7,9970e-002	1,5850e-003	0,00	7,3675
	CI	$\pm 6,9033e-006$	$\pm 8,3475e-003$	$\pm 9,2270e-002$	$\pm 1,8288e-003$	$\pm 0,00$	$\pm 8,5008$
(0,01; 0,005)	MEAN	7,5996e-007	4,5872e-003	1,0158e-002	2,0133e-004	100,00	98,5294
	STD	2,9998e-007	1,1131e-003	4,0096e-003	7,9471e-005	0,00	0,5080
	CI	$\pm 3,4613e-007$	$\pm 1,2843e-003$	$\pm 4,6264e-003$	$\pm 9,1694e-005$	$\pm 0,00$	$\pm 0,5861$
(0,01; 0,05)	MEAN	1,1376e-006	5,6270e-003	1,5206e-002	3,0138e-004	100,00	97,9361
	STD	5,5969e-007	1,2843e-003	7,4808e-003	1,4827e-004	0,00	0,8464
	CI	$\pm 6,4577e-007$	$\pm 1,4818e-003$	$\pm 8,6315e-003$	$\pm 1,7108e-004$	$\pm 0,00$	$\pm 0,9766$
(0,01; 0,5)	MEAN	1,7955e-006	6,6628e-003	2,3998e-002	4,7565e-004	100,00	97,0374
	STD	1,8647e-006	3,0949e-003	2,4924e-002	4,9400e-004	0,00	2,5876
	CI	$\pm 2,1516e-006$	$\pm 3,5710e-003$	$\pm 2,8758e-002$	$\pm 5,6998e-004$	$\pm 0,00$	$\pm 2,9856$
(0,1; 0,005)	MEAN	2,0417e-006	7,6043e-003	2,7290e-002	5,4088e-004	100,00	96,5846
	STD	6,2320e-007	1,3084e-003	8,3298e-003	1,6510e-004	0,00	0,9232
	CI	$\pm 7,1906e-007$	$\pm 1,5096e-003$	$\pm 9,6110e-003$	$\pm 1,9049e-004$	$\pm 0,00$	$\pm 1,0652$
(0,1; 0,05)	MEAN	2,0263e-006	7,5703e-003	2,7083e-002	5,3679e-004	100,00	96,6076
	STD	6,3069e-007	1,3403e-003	8,4298e-003	1,6708e-004	0,00	0,9369
	CI	$\pm 7,2769e-007$	$\pm 1,5464e-003$	$\pm 9,7264e-003$	$\pm 1,9278e-004$	$\pm 0,00$	$\pm 1,0810$
(0,1; 0,5)	MEAN	2,1550e-006	7,8708e-003	2,8804e-002	5,7089e-004	100,00	96,4121
	STD	4,1830e-007	8,1345e-004	5,5911e-003	1,1082e-004	0,00	6,6093
	CI	$\pm 4,8264e-007$	$\pm 9,3857e-004$	$\pm 6,4511e-003$	$\pm 1,2786e-004$	$\pm 0,00$	$\pm 0,7031$

De acordo com a Tabela 6.25, pode-se verificar que os melhores valores de μ e σ para a série USIM5 foram 0,01 e 0,005, respectivamente, uma vez que obtivemos o melhor desempenho de previsão em termos da função de avaliação global (EF). Portanto, assumindo essa suposição, escolheu-se o modelo DELP(0,01; 0,005) com a maior métrica EF no conjunto de validação,

dentre os dez modelos criados neste experimento, para representar o modelo proposto. Neste modelo, o termo de combinação λ determinado foi 0,0111, significando que o DELP utilizou 98,89% da componente linear e 1,11% da componente não-linear, confirmando a hipótese (através da análise do *lagplot* apresentada no Capítulo 4) que séries temporais financeiras são geradas por componentes lineares dominantes e componentes não-lineares sub-dominantes. Vale mencionar que o termo de combinação da componente não-linear θ foi 0,3116, isto é, utilização de 31,16% do operador de dilatação e 68,84% do operador de erosão. A Tabela 6.26 apresenta os resultados (referentes ao conjunto de teste) com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP.

Tabela 6.26 Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série USIM5.

	MSE	MAPE	THEIL	ARV	POCID	EF
ARIMA	1,2208e-004	4,8637e-002	1,6290e+000	3,1230e-002	54,22	20,0136
MLP	7,5881e-005	3,7166e-002	1,0210e+000	2,0934e-002	59,76	28,7407
MGA-MLP	9,8718e-005	4,1555e-002	1,3282e+000	2,7234e-002	59,35	24,7587
PSO-MLP	7,3855e-005	3,6173e-002	9,9371e-001	2,0375e-002	60,16	29,3429
TAEF	1,5872e-005	1,7199e-002	2,1294e-001	4,4115e-003	98,37	79,6778
SHIF	1,3126e-005	1,5656e-002	1,7610e-001	3,6483e-003	98,37	82,2870
DELP	2,3830e-007	2,6285e-003	3,1851e-003	6,3128e-005	100,00	99,4157

Os resultados obtidos nos experimentos realizados com a série USIM5, apresentados na Tabela 6.26, mostram que o desempenho preditivo do modelo DELP para todas as métricas investigadas é superior aos resultados encontrados na literatura. Verificamos novamente, que a métrica THEIL obtida pelo modelo proposto foi bem próxima de zero, mostrando que o dilema do passeio aleatório foi superado. A Tabela 6.27 mostra os ganhos percentuais (do conjunto de teste) do modelo DELP proposto com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF.

Tabela 6.27 Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF para a série USIM5.

	MSE	MAPE	THEIL	ARV	POCID	EF
Ganho (%) DELP / ARIMA	99,80	94,60	99,80	99,80	84,43	396,74
Ganho (%) DELP / MLP	99,69	92,93	99,69	99,70	67,34	245,91
Ganho (%) DELP / MGA-MLP	99,76	93,67	99,76	99,77	68,49	301,54
Ganho (%) DELP / PSO-MLP	99,68	92,73	99,68	99,69	66,22	238,81
Ganho (%) DELP / TAEF	98,50	84,72	98,50	98,57	1,66	24,77
Ganho (%) DELP / SHIF	98,18	83,21	98,19	98,27	1,66	20,82

É possível verificar na Tabela 6.27 que o modelo proposto obteve desempenho preditivo bastante superior aos resultados reportados na literatura (obtendo um ganho percentual superior a 66% para todas as métricas, exceto a métrica POCID nos modelos TAEF e SHIF, obtendo um ganho percentual em torno de 1%). Vale mencionar que, em termos globais, o modelo proposto

obteve um ganho percentual superior a 238% em relação aos modelos ARIMA, MLP, MGA-MLP e PSO-MLP, e superior a 20% em relação aos modelos TAEF e SHIF.

As Figuras 6.17 e 6.18 apresentam um gráfico comparativo entre os valores reais da série (linha sólida) e os valores preditos pelo modelo proposto (linha tracejada) para os últimos cem e dez pontos, respectivamente, do conjunto de teste. Vale mencionar que a previsão está bem próxima aos valores reais da série, o que comprova que o característico atraso de um passo da previsão não ocorreu e o dilema do passei aleatório foi mais uma vez superado.

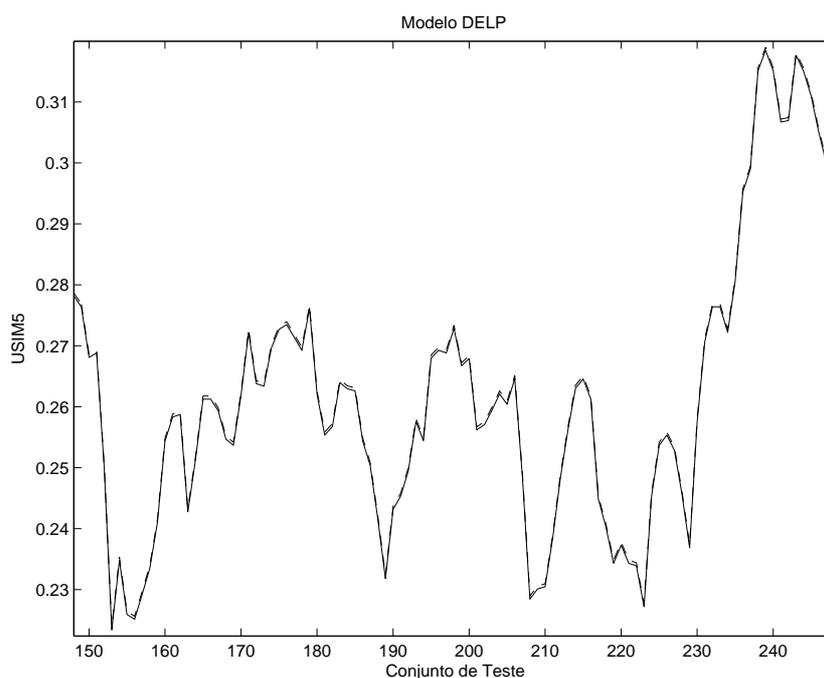


Figura 6.17 Previsão da série USIM5 gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).

6.2.10 Preço das Ações da Vale PNA (VALE5)

Para a série do Preço das Ações da Vale PNA (VALE5), os modelos utilizados que obtiveram o melhor desempenho foram: i) modelo ARIMA com a configuração (5,1,1), ii) modelo MLP com a configuração 5-25-1 e arquitetura definida pela Equação (3.44), iii) o modelo MGA-MLP definiu os retardos temporais (1, 4 e 5) e a rede MLP com a configuração 3-7-1 e arquitetura definida pela Equação (3.44), iv) o modelo PSO-MLP escolheu os retardos temporais (1, 4, 7, 8 e 10) e a rede MLP com a configuração 5-6-1 e arquitetura definida pela Equação (3.46), v) o modelo TAEF determinou os retardos temporais (2, 5, 8 e 10) e a rede MLP com a configuração 4-10-1, arquitetura definida pela Equação (3.44) e o algoritmo *scaled conjugate gradient* para treinamento da rede MLP, e vi) o modelo SHIF determinou os retardos temporais (2, 4 e 6) e a rede MLP com a configuração 3-5-1, arquitetura definida pela Equação (3.44) e o algoritmo *levenberg marquardt* para treinamento da rede MLP.

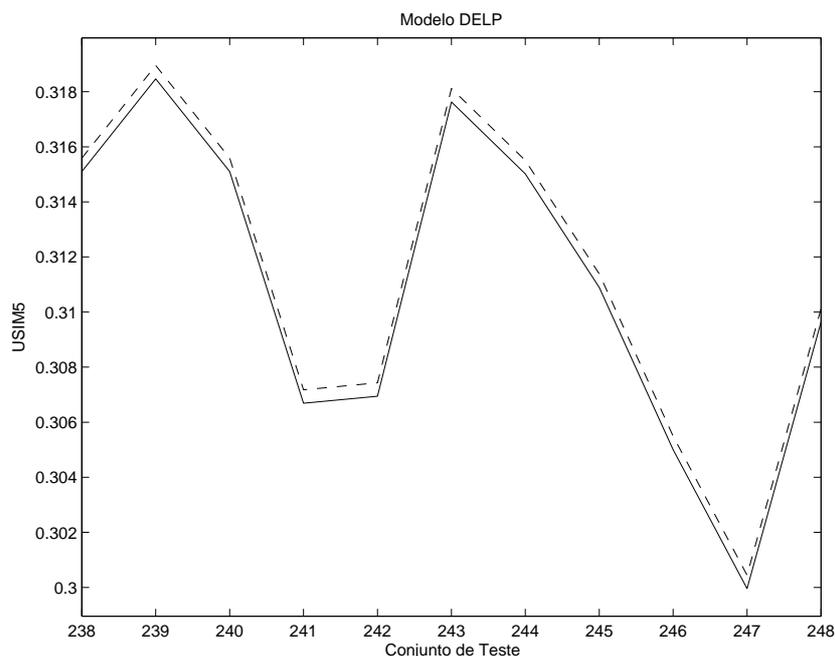


Figura 6.18 Previsão da série USIM5 gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).

A Tabela 6.28 apresenta todos os experimentos realizados com o modelo DELP (utilizando o conjunto de treinamento) variando o valor de μ e σ , onde foram calculadas todas métricas, bem como as estatísticas MEAN, STD e CI para análise do desempenho de previsão médio do modelo proposto.

De acordo com a Tabela 6.28, pode-se verificar que os melhores valores de μ e σ para a série VALE5 foram 0,001 e 0,005, respectivamente, uma vez que obtivemos o melhor desempenho de previsão em termos da função de avaliação global (EF). Portanto, assumindo essa suposição, escolheu-se o modelo DELP(0,001; 0,005) com a maior métrica EF no conjunto de validação, dentre os dez modelos criados neste experimento, para representar o modelo proposto. Neste modelo, o termo de combinação λ determinado foi 0,0190, significando que o DELP utilizou 98,10% da componente linear e 1,90% da componente não-linear, confirmando a hipótese (através da análise do *lagplot* apresentada no Capítulo 4) que séries temporais financeiras são geradas por componentes lineares dominantes e componentes não-lineares sub-dominantes. Vale mencionar que o termo de combinação da componente não-linear θ foi 0,9012, isto é, utilização de 90,12% do operador de dilatação e 9,88% do operador de erosão. A Tabela 6.29 apresenta os resultados (referentes ao conjunto de teste) com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP.

Os resultados obtidos nos experimentos realizados com a série VALE5 mostram que o desempenho preditivo do modelo DELP para todas as métricas investigadas é superior aos resultados encontrados na literatura. Verifica-se que a métrica THEIL obtida pelo modelo proposto foi bem próxima de zero, mostrando novamente que o dilema do passeio aleatório foi superado. A Tabela 6.30 mostra os ganhos percentuais (do conjunto de teste) do modelo DELP proposto

Tabela 6.28 Resultados de todos os experimentos utilizando o modelo proposto para a série VALE5 (conjunto de treinamento)

DELDP ($\mu; \sigma$)	Estatística	Métricas					
		MSE	MAPE	THEIL	ARV	POCID	EF
(0,001; 0,005)	MEAN	2,9559e-006	3,7832e-003	8,1527e-003	1,2612e-004	100,00	98,8128
	STD	2,0736e-006	2,1580e-003	5,7193e-003	8,8473e-005	0,00	0,7778
	CI	$\pm 2,3926e-006$	$\pm 2,4899e-003$	$\pm 6,5990e-003$	$\pm 1,0208e-004$	$\pm 0,00$	$\pm 0,8974$
(0,001; 0,05)	MEAN	4,2169e-006	4,9576e-003	1,1631e-002	1,7992e-004	100,00	98,3540
	STD	2,0424e-006	1,2024e-003	5,6333e-003	8,7144e-005	0,00	0,6674
	CI	$\pm 2,3566e-006$	$\pm 1,3873e-003$	$\pm 6,4998e-003$	$\pm 1,0055e-004$	$\pm 0,00$	$\pm 0,7701$
(0,001; 0,5)	MEAN	6,8060e-006	6,0072e-003	1,8772e-002	2,9039e-004	100,00	97,5737
	STD	4,9286e-006	2,6093e-003	1,3594e-002	2,1028e-004	0,00	1,5616
	CI	$\pm 5,6867e-006$	$\pm 3,0107e-003$	$\pm 1,5685e-002$	$\pm 2,4263e-004$	$\pm 0,00$	$\pm 1,8017$
(0,01; 0,005)	MEAN	1,0039e-005	7,1929e-003	2,7690e-002	4,2835e-004	100,00	96,6661
	STD	1,0732e-005	3,4510e-003	2,9601e-002	4,5791e-004	0,00	2,9999
	CI	$\pm 1,2383e-005$	$\pm 3,9818e-003$	$\pm 3,4154e-002$	$\pm 5,2834e-004$	$\pm 0,00$	$\pm 3,4613$
(0,01; 0,05)	MEAN	4,4934e-006	5,0352e-003	1,2393e-002	1,9172e-004	100,00	98,2724
	STD	2,1040e-006	1,6079e-003	5,8031e-003	8,9769e-005	0,00	0,7313
	CI	$\pm 2,4276e-006$	$\pm 1,8552e-003$	$\pm 6,6956e-003$	$\pm 1,0358e-004$	$\pm 0,00$	$\pm 0,8438$
(0,01; 0,5)	MEAN	5,5927e-006	5,8416e-003	1,5425e-002	2,3862e-004	100,00	97,8942
	STD	1,5224e-007	7,9333e-005	4,1989e-004	6,4954e-006	0,00	0,0485
	CI	$\pm 1,7565e-007$	$\pm 9,1535e-005$	$\pm 4,8447e-004$	$\pm 7,4945e-006$	$\pm 0,00$	$\pm 0,0559$
(0,1; 0,005)	MEAN	1,3123e-005	8,5510e-003	3,6195e-002	5,5991e-004	100,00	95,7192
	STD	8,9911e-006	2,9506e-003	2,4799e-002	3,8362e-004	0,00	2,5377
	CI	$\pm 1,0374e-005$	$\pm 3,4045e-003$	$\pm 2,8613e-002$	$\pm 4,4262e-004$	$\pm 0,00$	$\pm 2,9281$
(0,1; 0,05)	MEAN	1,2471e-005	8,3650e-003	3,4396e-002	5,3208e-004	100,00	95,8987
	STD	8,5931e-006	2,7681e-003	2,3701e-002	3,6664e-004	0,00	2,4073
	CI	$\pm 9,9148e-006$	$\pm 3,1938e-003$	$\pm 2,7346e-002$	$\pm 4,2303e-004$	$\pm 0,00$	$\pm 2,7775$
(0,1; 0,5)	MEAN	1,6404e-005	9,5777e-003	4,5245e-002	6,9991e-004	100,00	94,8025
	STD	9,7812e-006	3,2356e-003	2,6978e-002	4,1733e-004	0,00	2,7635
	CI	$\pm 1,1286e-005$	$\pm 3,7333e-003$	$\pm 3,1127e-002$	$\pm 4,8152e-004$	$\pm 0,00$	$\pm 3,1886$

Tabela 6.29 Resultados experimentais (conjunto de teste) com os melhores modelos (ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF, SHIF e DELP) para a série VALE5.

	MSE	MAPE	THEIL	ARV	POCID	EF
ARIMA	4,9965e-004	4,1664e-002	1,3881e+000	2,0968e-002	49,80	20,3158
MLP	3,8806e-004	3,4891e-002	1,0789e+000	1,6592e-002	49,19	23,0841
MGA-MLP	3,7297e-004	3,4724e-002	1,0370e+000	1,5947e-002	48,37	23,1676
PSO-MLP	4,3964e-004	3,7566e-002	1,2223e+000	1,8797e-002	47,56	20,8683
TAEF	2,3196e-005	8,6518e-003	6,3737e-002	9,9554e-004	90,24	84,0723
SHIF	2,3690e-005	8,7577e-003	6,5092e-002	1,0167e-003	90,24	83,9564
DELP	1,5873e-006	3,1124e-003	4,3781e-003	6,7726e-005	100,00	99,2497

com todas as métricas de desempenho para os modelos ARIMA, MLP, MGA-MLP, PSO-MLP, TAEF e SHIF.

Verifica-se na Tabela 6.30 que o modelo proposto obteve desempenho preditivo bastante superior aos resultados reportados na literatura (obtendo um ganho percentual superior a 91% para todas as métricas, exceto as métricas MAPE e POCID nos modelos TAEF e SHIF, obtendo um ganho percentual superior a em torno de 64% e 10%, respectivamente). Vale mencionar que, em termos globais, o modelo proposto obteve um ganho percentual superior a 328% em relação aos modelos ARIMA, MLP, MGA-MLP e PSO-MLP, e em torno de 18% em relação aos modelos TAEF e SHIF.

Tabela 6.30 Ganhos percentuais (conjunto de teste) do modelo DELP em relação aos modelos ARIMA, MLP, MGA–MLP, PSO–MLP, TAEF e SHIF para a série VALE5.

	MSE	MAPE	THEIL	ARV	POCID	EF
Ganho (%) DELP / ARIMA	99,68	92,53	99,68	99,68	100,80	388,53
Ganho (%) DELP / MLP	99,59	91,08	99,59	99,59	103,29	329,95
Ganho (%) DELP / MGA–MLP	99,57	91,04	99,58	99,58	106,74	328,40
Ganho (%) DELP / PSO–MLP	99,64	91,71	99,64	99,64	110,26	375,60
Ganho (%) DELP / TAEF	93,16	64,03	93,13	93,20	10,82	18,05
Ganho (%) DELP / SHIF	93,30	64,46	93,27	93,34	10,82	18,22

As Figuras 6.19 e 6.20 apresentam um gráfico comparativo entre os valores reais da série (linha sólida) e os valores preditos pelo modelo proposto (linha tracejada) para os últimos cem e dez pontos, respectivamente, do conjunto de teste. Vale mencionar que a previsão está sobreposta aos valores reais da série, o que comprova que o característico atraso de um passo da previsão não ocorreu e o dilema do passei aleatório foi novamente superado.

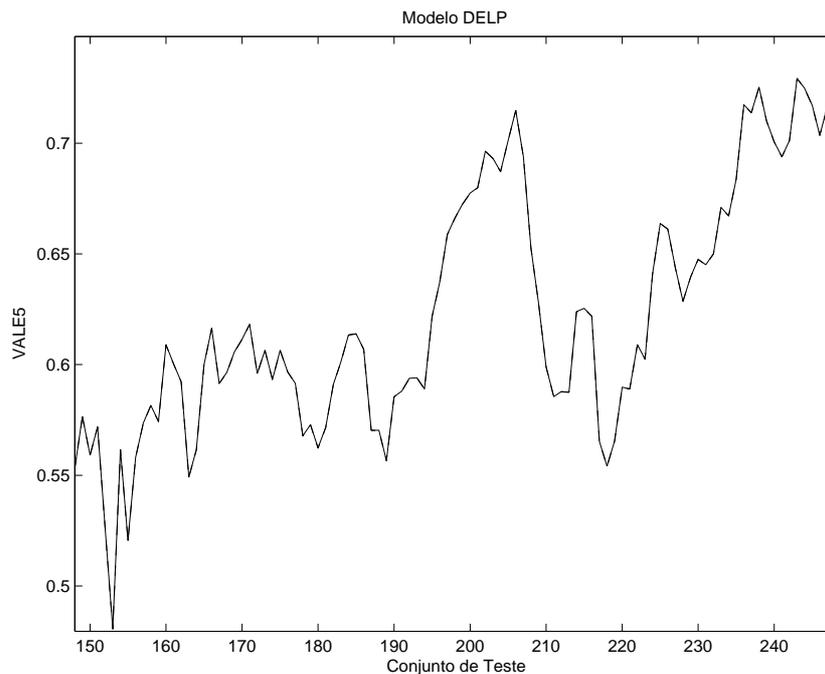


Figura 6.19 Previsão da série VALE5 gerada pelo modelo DELP proposto (últimos cem pontos do conjunto de teste).

6.2.11 Considerações

A partir dos experimentos realizados para as séries temporais financeiras investigadas é possível observar que, em todos os casos, o modelo proposto obteve desempenho preditivo expressivamente superior aos modelos encontrados na literatura, e superando, de forma prática, o dilema

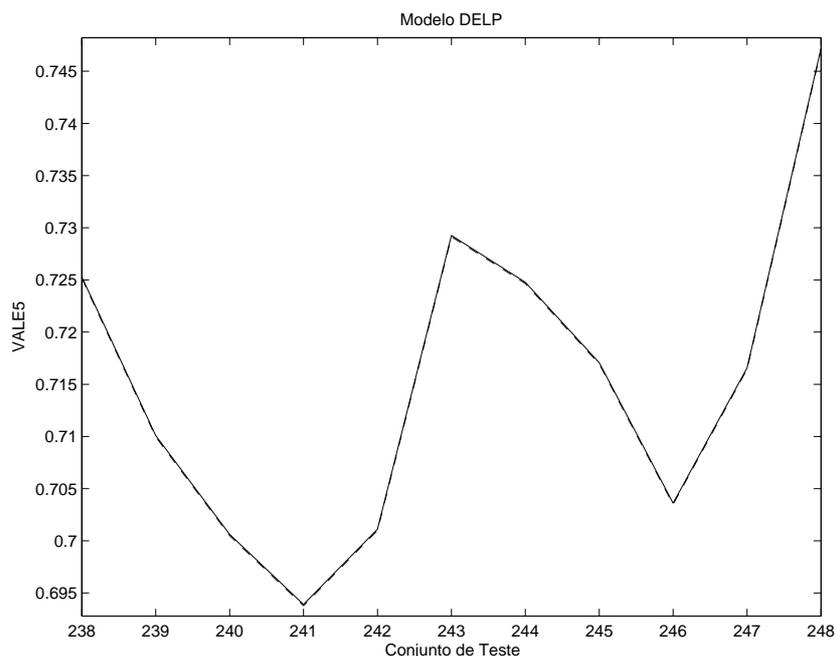


Figura 6.20 Previsão da série VALE5 gerada pelo modelo DELP proposto (últimos dez pontos do conjunto de teste).

do passeio aleatório. Assim, é possível evidenciar que o modelo proposto conseguiu alcançar um desempenho superior no problema de previsão de séries temporais financeiras quando comparado as melhores técnicas de previsão reportadas na literatura (sob as mesmas condições de experimentação), e em especial, na resolução do dilema do passeio aleatório para previsão de séries temporais financeiras. Tal fato nos indica que o modelo proposto é uma opção válida de alto desempenho preditivo que pode eficientemente ser utilizada para previsão financeira.

Também, a partir da análise do termo de combinação λ do DELP, foi confirmada a hipótese apresentada no Capítulo 4, indicando que séries temporais financeiras são geradas por uma combinação de processos lineares dominantes e processos não-lineares sub-dominantes. Para todos os experimentos com séries temporais financeiras o termo de mistura do DELP foi determinado por uma forte componente linear (utilização superior a 95%) e uma componente não-linear complexa sub-dominante (utilização inferior a 5%).

Portanto, encontramos evidências claras que justificam o fato do modelo proposto ter alcançado desempenho preditivo bastante superior como solução do dilema do passeio aleatório para séries financeiras: i) a hipótese, confirmada pela análise do *lagplot* e pelo termo de combinação do DELP, que séries temporais financeiras são geradas por uma combinação de componentes lineares e não-lineares (onde tal relacionamento não-linear complexo pode ser eficientemente estimado por funções morfológicas crescentes), ii) a habilidade do DELP selecionar, através de uma estimativa balanceada, o percentual de utilização das componentes linear e não-linear, e iii) a aplicação do algoritmo de treinamento baseado em gradiente descendente com ajuste de fase temporal.

6.3 Resumo do Capítulo

Neste Capítulo foram discutidos os resultados obtidos pelo modelo proposto e pelas técnicas consolidadas na literatura. Inicialmente foi apresentada a metodologia utilizada para realização dos experimentos com cada técnica em particular. Depois, para cada série temporal, foram apresentados os resultados utilizando cinco métricas bem conhecidas na literatura e uma função de avaliação que mede o desempenho preditivo global de um determinado modelo de previsão. Também, foi apresentada uma análise tendo como base o ganho percentual do modelo proposto em relação a todas as técnicas investigadas, permitindo, em termos percentuais, para cada métrica e para a função de avaliação, determinar o quão melhor foi o modelo proposto. Tendo em vista verificar, graficamente, o desempenho preditivo do modelo, foram apresentados os gráficos resultantes da previsão para os últimos cem e dez pontos do conjunto de teste, de forma a observar que as distorções de fase temporais (comportamento caracterizado por um atraso de um passo da previsão em relação aos valores reais da série) foram corretamente ajustadas. Por fim, foi confirmado, a partir da análise do termo de mistura do DELP que séries financeiras são geradas por uma combinação de processos lineares dominantes e processos não-lineares sub-dominantes, sendo passíveis de previsão, bem como foi constatado que o modelo proposto obteve desempenho bastante superior às técnicas consolidadas na literatura de previsão de séries temporais financeiras.

CAPÍTULO 7

Conclusões

Neste Capítulo são apresentadas as considerações finais sobre o trabalho desenvolvido nesta dissertação. Também, é realizada uma breve discussão sobre o modelo proposto, considerando sua viabilidade em aplicações de previsão de séries temporais financeiras, bem como são discutidas as suas limitações práticas e teóricas. Ao final, são apresentadas algumas sugestões para trabalhos futuros, tendo em vista contribuir para a evolução do modelo proposto e para um melhor entendimento do problema de previsão de séries temporais financeiras.

7.1 Resumo da Dissertação

Neste trabalho, o problema de previsão de séries temporais foi abordado com o estudo de várias técnicas inspiradas na estatística e inteligência computacional (*computational intelligence*, CI). Os modelos estatísticos, lineares e não lineares, são os mais popularmente encontrados na literatura de previsão de séries temporais. Neste contexto, destacam-se os modelos autoregressivos integrados de médias móveis (*autoregressive integrated moving average*, ARIMA), que são os mais difundidos e utilizados em aplicações de previsão financeira.

Entretanto, devido a linearidade dos modelos ARIMA é possível identificar uma limitação prática de sua utilização, uma vez que os fenômenos temporais financeiros são comumente não lineares. Analisando os modelos estatísticos não lineares, pode-se verificar uma limitação prática de sua utilização, uma vez que a não linearidade de tais modelos não implica em um ganho significativo, em termos de desempenho preditivo, quando comparado ao desempenho de modelos estatísticos lineares. Neste contexto, uma alternativa promissora surgiu a partir do desenvolvimento de modelos baseados em CI para previsão de séries temporais.

Dentre estes modelos, foram analisados aqueles mais comumente encontrados em aplicações de séries financeiras, bem como os modelos que na literatura obtiveram os melhores resultados preditivos, que foram os modelos baseados em redes neurais do tipo *perceptron* multicamadas (*multilayer perceptron*, MLP). Note que, a previsão de uma série temporal consiste basicamente na tentativa de mapear informações passadas em informações futuras. Portanto, como as redes do tipo MLP possuem uma capacidade intrínseca de mapeamento de informação, estas são consideradas as mais comumente utilizadas para o propósito de previsão financeira. No entanto, as redes MLP requerem a definição de um conjunto de parâmetros bastante difícil de se determinar. Além disso, no caso particular do problema de previsão séries temporais financeiras utilizando redes MLP, um elemento crucial para previsão é a definição dos retardos temporais para representar a série.

A partir deste ponto, foram investigados o algoritmo genético modificado (*modified gene-*

tic algorithm, MGA) e o otimizador de enxame de partículas (*particle swarm optimizer*, PSO) para configuração e ajuste automático de todos os parâmetros das redes MLP, gerando sistemas híbridos (MGA–MLP e PSO–MLP) para previsão de séries temporais financeiras. No entanto, como o espaço de busca dos parâmetros das redes MLP é bastante complexo, ambas as técnicas podem não ser capazes de ajustar e treinar, simultaneamente e eficientemente, este tipo particular de rede neural. Portanto, foram investigados sistemas híbridos otimizados, o método *time-delay added evolutionary forecasting* (TAEF) e o método *swarm-based hybrid intelligent forecasting* (SHIF), que utilizam o MGA e o PSO, respectivamente, para ajuste das redes MLP, e utilizam técnicas baseadas em gradiente descendente para treiná-las. Estes modelos híbridos otimizados apresentaram resultados promissores e atualmente são consideradas as melhores técnicas de previsão encontradas na literatura.

Entretanto, mesmo com as técnicas sofisticadas e metodologias híbridas que têm sido recentemente propostas na literatura para solucionar o problema de previsão de séries temporais, uma limitação surge de todos estes modelos de previsão no caso particular de séries temporais financeiras, conhecido como dilema do passeio aleatório. Este dilema ocorre quando se tenta prever qualquer série temporal financeira utilizando um modelo arbitrário de previsão, onde conseqüentemente as previsões geradas apresentam um característico atraso de um passo a frente em relação aos valores reais da série temporal, isto é, possuem uma distorção de fase temporal na reconstrução do espaço de fase de um determinado fenômeno temporal financeiro. A prova de tal afirmação foi desenvolvida no Capítulo 2.

Devido a este fato, alguns pesquisadores argumentaram que séries temporais financeiras são geradas por processos aleatórios e não podem ser previstas. Entretanto, a justificativa para tal afirmação se baseia no fato que nenhum modelo de previsão proposto na literatura é capaz de prever estas séries sem o atraso de um passo da previsão em relação aos valores reais da série. Portanto, utilizou-se um conjunto relevante de séries temporais financeiras comumente utilizadas na literatura na tentativa de investigar melhor este tipo de série: Índice da Bolsa de Valores de São Paulo, Índice *Dow Jones Industrial Average*, Índice *National Association of Securities Dealers Automated Quotation*, Índice *Financial Times and London Stock Exchange 100*, Preço das Ações do Bradesco PN, Preço das Ações da Gol PN, Preço das Ações do Itaú Unibanco PN, Preço das Ações da Petrobras PN, Preço das Ações da Usiminas PNA e Preço das Ações da Vale PNA. A análise dos *lagplots* destas séries realizada no Capítulo 4 nos fornece a demonstração prática que séries temporais financeiras são geradas por componentes não lineares complexas subdominantes e uma componente linear dominante. Este argumento nos fornece as indicações necessárias para defender a não aleatoriedade destas séries.

Nesta análise realizada no Capítulo 4, verifica-se que as séries financeiras investigadas não são geradas por processos aleatórios, uma vez que foi encontrada a presença de relacionamentos não lineares subdominantes complexos embutidos em relacionamentos lineares dominantes. Se tais séries fossem unicamente aleatórias não seria possível identificar quaisquer relacionamentos não lineares em retardos temporais de alta ordem, isto é, seria possível identificar apenas um aumento na variância nos pontos destes gráficos, uma vez que em um modelo de passeio aleatório a variância aumenta linearmente com o aumento da ordem dos retardos temporais.

Assim, pode-se concluir que séries financeiras têm um comportamento bastante similar a séries de passeio aleatório em retardos de baixa ordem. No entanto, com o aumento da ordem

dos retardos temporais, tem-se claras evidências de relacionamentos não lineares complexos presentes nestas séries. Neste sentido, pode-se notar que este tipo de série não é gerado por um processo aleatório, mas por uma combinação de processos lineares dominantes e processos não lineares subdominantes. Note que o relacionamento linear dominante é sensível aos eventos em um passado bastante recente, o que leva a um comportamento similar a um processo aleatório devido a fortes inferências especulativas no mercado financeiro. Também, é possível verificar que o relacionamento não linear subdominante é responsável por eventos de médio/longo prazo, o que leva a uma percepção correta do fenômeno temporal sem a inferência especulativa (aqui pode-se observar o comportamento de médio/longo prazo da série, sendo este responsável pelo aparecimento destas estruturas não lineares bem definidas nos *lagplots*). Também, outra observação interessante apresentada nesta análise é que tais componentes podem ser eficientemente aproximadas em termos de uma combinação balanceada entre operadores lineares de resposta finita ao impulso (componente linear) e operadores não lineares crescentes (componente não linear). A justificativa da estimativa da componente não linear utilizando operadores crescentes é devido ao fato que a análise destas séries enfatizando o critério de médio/longo prazo revelou que o comportamento do mercado financeiro é naturalmente crescente. Por fim, foi apresentado um conjunto de métricas com características diversas (POCID, MSE, MAPE, THEIL e ARV) e uma função de avaliação (EF) para medir o desempenho preditivo global dos modelos investigados.

Definidas as principais motivações deste trabalho, foi proposta uma classe de *perceptrons* híbridos baseado nos princípios da morfologia matemática (*mathematical morphology*, MM) no contexto de teoria de reticulados (*lattice theory*) para superar o dilema do passeio aleatório (*random walk dilemma*, RWD) no problema de previsão de séries temporais financeiras. A MM sobre o contexto de teoria de reticulados foi escolhida para fundamentar o modelo proposto devido à sua natureza inerentemente crescente. Vale mencionar que a justificativa teórica da possibilidade de construção do futuro se fundamenta no trabalho de Takens, que mostrou, matematicamente, que é possível reproduzir qualquer fenômeno temporal não aleatório no tempo futuro a partir de suas observações no tempo passado, isto é, a partir da escolha de uma certa dimensionalidade mínima necessária para uma correta caracterização do sistema gerador deste fenômeno.

O modelo proposto foi chamado de *perceptron* de dilatação-erosão-linear (*dilation-erosion-linear perceptron*, DELP), uma vez que este consiste de uma combinação linear entre operadores não lineares (do tipo morfológicos no contexto de teoria de reticulados) e um operador linear (do tipo resposta finita ao impulso). A ideia de propor este modelo é fundamentada pela análise das séries temporais financeiras realizada no Capítulo 4. Assim, o DELP foi projetado de forma a conter toda informação necessária para se prever séries temporais financeiras, uma vez que sabe-se a priori os tipos de relacionamento que governam esse tipo particular de séries temporais (relacionamento linear dominante e um relacionamento não linear subdominante com comportamento crescente).

Para projetar o DELP (processo de aprendizagem), foi proposto um método de gradiente descendente utilizando ideias do algoritmo de retropropagação do erro (*back propagation*, BP) e uma abordagem sistemática para superar o problema da não-diferenciabilidade das operações morfológicas de dilatação e erosão, uma vez que estes operadores não são diferenciáveis de

forma usual. Também, no processo de aprendizagem do DELP, foi incluída uma etapa adicional, definida como procedimento de correção de fase (*phase fix procedure*, PFP), para ajustar distorções de fase temporais que ocorrem na reconstrução do espaço de fase de fenômenos financeiros.

Para se estabelecer um nível de referência para o desempenho preditivo do modelo proposto, foram inicialmente realizados experimentos com os modelos ARIMA, haja vista que estes são modelos lineares comumente utilizados em aplicações no mercado financeiro. Posteriormente, foram realizados vários experimentos com redes MLPs, onde os ajustes e configurações aplicados foram escolhidos através de exaustivos experimentos preliminares com o objetivo de otimizar o modelo. Depois, foram realizados os experimentos com os sistemas híbridos MGA-MLP e PSO-MLP e os sistemas híbridos otimizados TAEF e SHIF. Por fim, foram realizados os experimentos com o modelo proposto.

Analisando os resultados alcançados, pode-se verificar que as redes MLP obtiveram um desempenho preditivo superior aos modelos ARIMA. No entanto, mesmo aplicando o procedimento de correção de fase (que ajusta distorções de fase temporais na representação das séries financeiras), ambos os modelos seguem o mesmo comportamento de um modelo de passeio aleatório (como pode ser verificado pelo valor da métrica THEIL em torno de 1). Observando este comportamento, pode-se assumir que este resultado é um reflexo da escolha incorreta dos parâmetros do modelo de previsão (sendo determinados a partir de procedimentos manuais e não otimizados).

Quando se observa os resultados obtidos pelos sistemas híbridos MGA-MLP e PSO-MLP, verifica-se que estes obtiveram desempenho levemente superior aos modelos MLP e ARIMA. No entanto, mais uma vez, mesmo aplicando o procedimento de correção de fase, ambos os modelos também seguem o mesmo comportamento de um modelo de passeio aleatório (THEIL com o valor próximo de 1). Este comportamento vem provavelmente do fato do espaço de busca para os parâmetros da rede MLP constitui um universo extremamente vasto e de dimensionalidade alta, necessitando de uma grande quantidade de tempo para um correto treinamento e ajuste, simultâneo e automático, da rede MLP. Mesmo assim, estes modelos híbridos têm a vantagem de não necessitarem da interação de um especialista humano para construção do modelo final de previsão.

Uma explicação plausível para a incapacidade de ajuste das distorções de fase temporais destes modelos de previsão é que o procedimento de correção de fase depende da complexidade da informação contida na série temporal e a habilidade de definir precisamente os melhores parâmetros do modelo de previsão para estimar os valores reais da série, ou seja, o sucesso do procedimento de correção de fase é fortemente dependente do ajuste acurado dos parâmetros do modelo de previsão, bem como do próprio modelo utilizado para o propósito de previsão.

Ao observar os resultados alcançados com os modelos híbridos otimizados (TAEF e SHIF), verificou-se um desempenho preditivo superior aos modelos consolidados na literatura (THEIL com valor inferior a 1), onde o procedimento de correção de fase se mostrou capacitado para o ajuste da distorção de fase temporal, o que evidencia que alguns modelos de previsão só conseguem superar o problema quando são corretamente ajustados com técnicas de busca globais combinadas com técnicas de busca locais para refinamento dos parâmetros globais.

Observando os resultados obtidos nos experimentos realizados, pode-se notar que o modelo

proposto conseguiu superar, em termos de qualidade de previsão, os resultados alcançados pelos modelos consolidados na literatura de previsão de séries financeiras, considerando todas as métricas, individualmente, bem como considerando a função de avaliação global. Pode-se notar que, devido a natureza linear e não linear do DELP (característica inerente a séries financeiras), e a inclusão do procedimento de correção de fase do algoritmo de treinamento do modelo proposto, este foi capaz de ajustar mais precisamente a distorção de fase temporal de séries temporais financeiras em relação ao ajuste alcançado pelos modelos TAEF e SHIF. Em outras palavras, o modelo proposto foi capaz de superar o dilema do passeio aleatório para previsão de séries temporais financeiras.

Uma explicação mais detalhada de tal comportamento é que o modelo proposto foi construído com características similares as que regem o fenômeno gerador de séries temporais financeiras, o que reduz o espaço de busca por um modelo ótimo de previsão. Tal fato associado a inclusão do procedimento de correção de fase no algoritmo de treinamento do DELP leva a um ajuste mais preciso das distorções de fase temporais, uma vez que a complexidade da informação contida nas séries financeiras é mapeada diretamente nas equações do DELP.

Vale mencionar que, em todas as séries temporais analisadas, verifica-se que o modelo proposto utilizou em torno de 98% da componente linear e em torno de 2% da componente não linear. Como o modelo proposto é capaz de selecionar o percentual de utilização de cada operador em particular, pode-se verificar que o mesmo estimou tais séries a partir de uma combinação balanceada de operadores lineares dominantes e operadores não lineares subdominantes. Tal fato confirma a análise dos *lagplots* que séries financeiras são geradas por uma combinação de relacionamentos lineares dominantes com relacionamentos não lineares subdominantes, uma vez que o modelo proposto é visto como um mapeamento direto dos relacionamentos contidos em séries temporais financeiras.

Também, pode-se observar que o modelo DELP não teve nenhuma similaridade com um modelo de passeio aleatório, uma vez em nenhuma série temporal analisada ocorreu o característico atraso de um passo em relação aos valores reais da série. Também, vale mencionar que o primeiro retardo temporal não foi selecionado para prever qualquer série com o modelo proposto, uma vez que uma estrutura de passeio aleatória é necessária para se utilizar o procedimento de correção de fase, pois a chave desta etapa é a previsão de dois passos para o ajuste do atraso de um passo. Note que, uma das principais vantagens do modelo proposto, além de seu desempenho preditivo (sem a utilização de técnicas evolucionárias e/ou híbridas em seu processo de aprendizagem) bem mais acurado quando comparado as melhores técnicas de previsão reportadas na literatura, é sua estrutura computacional bastante simples e a sua habilidade de estimar, de forma balanceada, as componentes linear e não linear, reproduzindo fielmente o fenômeno gerador de séries temporais financeiras.

Desta forma, o DELP apresenta-se como um modelo de previsão viável, em termos práticos e científicos, para solucionar o dilema do passeio aleatório para previsão de séries temporais financeiras.

7.2 Principais Contribuições

As principais contribuições deste trabalho são apresentadas a seguir:

- Investigação de modelos estatísticos lineares e não lineares, bem como modelos baseados em inteligência artificial, aplicados para previsão de séries temporais financeiras;
- Identificação, a partir da análise dos *lagplots*, que fenômenos temporais financeiros são gerados por uma combinação de componentes lineares dominantes e componentes não lineares complexas sub-dominantes, sendo esta a justificativa da possibilidade de se prever eficientemente séries financeiras;
- Proposta de uma classe de *perceptrons* híbridos baseado nos princípios da MM no contexto de teoria de reticulados para superar o dilema do passeio aleatório no problema de previsão de séries temporais financeiras:
 - Demonstração da possibilidade de se estimar a componente não linear de séries temporais financeiras utilizando os operadores morfológicos crescentes;
 - Definição formal do *perceptron* de dilatação-erosão-linear (*dilation-erosion-linear perceptron*, DELP);
 - Desenvolvimento de uma abordagem sistemática para superar o problema da não-diferenciabilidade das operações morfológicas que serão utilizadas no algoritmo de treinamento do DELP;
 - Desenvolvimento do processo de aprendizagem do DELP utilizando o método de gradiente descendente e ideias do algoritmo de retropropagação do erro
 - Inclusão do procedimento de correção de fase no processo de aprendizagem do DELP visando o ajuste de distorções de fase temporais que ocorrem na reconstrução do espaço de fase de fenômenos temporais financeiros (comportamento caracterizado por um atraso de um passo a frente da previsão em relação aos valores reais da série – “dilema do passeio aleatório”);
- Realização de uma análise experimental extensiva com o modelo proposto e outros modelos apresentados na literatura;
- Investigação do desempenho do modelo proposto utilizando um conjunto relevante de séries temporais financeiras a fim de demonstrar o comportamento do modelo proposto como solução viável para o dilema do passeio aleatório.

7.3 Principais Limitações da Dissertação

Embora o modelo proposto neste trabalho tenha alcançado desempenho preditivo bastante superior aos modelos consolidados na literatura, ainda existem algumas questões que devem ser investigadas.

Uma delas é a formalização das propriedades do modelo proposto visando determinar as suas limitações práticas e teóricas, em termos de desempenho preditivo, em outras séries temporais com componentes de tendências, sazonalidades, impulsos, passos, ruídos e outras não linearidades.

Outra questão importante é a justificativa do motivo pelo qual alguns modelos de previsão não conseguem ajustar as distorções de fase temporais, mesmo aplicando o procedimento de correção de fase. Uma explicação teórica de tal fenômeno ainda é desconhecida.

Por fim, um estudo formal, em termos de custo computacional, deve ser realizado. Entretanto, na prática, o modelo proposto necessita de um tempo em torno de trinta minutos para construção de um modelo robusto de previsão, utilizando um computador pessoal executando o sistema operacional Windows 7 64bits e com o processador Intel Core i5 (*clock* 2,66GHz dual core com 4 threads e 3MB de cache L3) e memória DDR3 de 6GB. Considerando uma aplicação prática no mercado de ações, o modelo proposto seria viável, uma vez que existe a possibilidade de construir o modelo de previsão antes do fechamento do pregão para fornecimento de informação para o processo de tomada de decisão de compra e venda de ações. Entretanto, quando a carteira de ações aumenta, isto é, a quantidade de séries temporais aumenta, o método proposto necessita de um poder computacional alto para que todas as previsões sejam geradas em tempo hábil.

7.4 Trabalhos Futuros

A formalização teórica e uma investigação mais detalhada sobre as possíveis limitações do modelo proposto, do processo de aprendizagem e da abordagem utilizada para superar o problema da não-diferenciabilidade dos operadores morfológicos deve ser realizada. Também, um estudo particular sobre a complexidade computacional do modelo proposto deve ser realizada para estabelecer sua avaliação completa em termos de custo-desempenho. De acordo com esta investigação, será possível determinar o tempo necessário para se construir um modelo ótimo de previsão.

A investigação de sistemas híbridos clássicos e híbridos otimizados utilizando o modelo proposto como base para geração das previsões deve ser feita, uma vez que um ponto importante a se considerar é a otimização do processo de determinação dos retardos temporais relevantes e parâmetros iniciais utilizados no modelo proposto.

Estudos adicionais, focando o risco e retorno financeiro, devem ser feitos para determinar os benefícios econômicos, para investidores no mercado de ações, com a utilização do modelo proposto. Também, deve ser feita a inclusão do modelo proposto em um sistema de tomada de decisão para o mercado de ações, tendo em vista a investigação do desempenho prático do DELP para determinação de pontos de compra e venda em uma carteira real de ações.

Por fim, uma questão interessante deste trabalho é o ajuste de distorções de fase para solucionar o dilema do passeio aleatório na previsão de séries temporais financeiras. Assumindo uma série como um sistema dinâmico contido em um espaço de fase, os valores observáveis da série perdem o termo de fase dos pontos da trajetória, o que sugere o desenvolvimento de modelos de previsão que preservem a correlação entre os observáveis e o sistema original. Assim, como estudos futuros é interessante se investigar os princípios de sistemas quânticos, e aplicá-las aos operadores morfológicos sob o contexto de teoria de reticulados, gerando modelos de *perceptrons* morfológicos quântico-inspirados, de forma ajustar naturalmente distorções de fase em fenômenos temporais financeiros.

Referências Bibliográficas

- [1] M. M. Ali and A. Torn. Population set-based global optimization algorithms: some modifications and numerical studies. *Comput. Oper. Res.*, 31(10):1703–1725, 2004.
- [2] P. Angelov. Supplementary crossover operator for genetic algorithms based on the centre-of-gravity paradigm. *Control and Cybernetics*, 30(2):159–176, 2001.
- [3] B. De Baets, E. Kerre, and M. Gupta. The fundamentals of fuzzy mathematical morphology, part 1: basic concepts. *International Journal of General Systems*, 23:155–171, 1994.
- [4] G. J. F. Banon and J. Barrera. Decomposition of mappings between complete lattices by mathematical morphology, part 1. general lattices. *Signal Processing*, 30(3):299–327, February 1993.
- [5] R. Battiti. One step secant conjugate gradient. *Neural Computation*, 4:141–166, 1992.
- [6] C. Berzuini and C. Larizza. A unified approach for modeling longitudinal and failure time data, with application in medical monitoring. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18:109–123, 1996.
- [7] G. Birkhoff. *Lattice Theory*. American Mathematical Society, Providence, 3rd edition, 1993.
- [8] G. Birkhoff and J. Lipson. Heterogeneous algebras. *Journal of Combinatorial Theory*, 8:115–133, 1970.
- [9] T. Bollerslev. Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, 31:307–328, 1986.
- [10] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel. *Time Series Analysis: Forecasting and Control*. Prentice Hall, New Jersey, third edition, 1994.
- [11] U. Braga-Neto and J. Goutsias. Supremal multiscale signal analysis. *SIAM Journal of Mathematical Analysis*, 36(1):94–120, 2004.
- [12] J. Brest, S. Greiner, B. Bošković, M. Mernik, and V. Žumer. Self-Adapting Control Parameters in Differential Evolution: A Comparative Study on Numerical Benchmark Problems. *IEEE Transactions on Evolutionary Computation*, 10(6):646–657, 2006.

- [13] B. Carré. An algebra for network routing problems. *J. Inst. Math. Appl.*, 7:273–294, 1971.
- [14] M. P. Clements, P. H. Franses, and N. R. Swanson. Forecasting economic and financial time-series with non-linear models. *International Journal of Forecasting*, 20:169–183, 2004.
- [15] M. P. Clements and D. F. Hendry. On the limitations of comparing mean square forecast errors. *Journal of Forecasting*, 12(8):617–637, Dec. 1993.
- [16] M. Crottel, B. Girard, Y. Girard, M. Mangeas, and C. Muller. Neural modeling for time series: a statistical stepwise method for weight elimination. *IEEE Transaction on Neural Networks*, 6(6):1355–1364, 1995.
- [17] R. Cuninghame-Green. *Minimax Algebra: Lecture Notes in Economics and Mathematical Systems 166*. Springer-Verlag, New York, 1979.
- [18] R. Cuninghame-Green. Minimax algebra and applications. In P. Hawkes, editor, *Advances in Imaging and Electron Physics*, volume 90, pages 1–121. Academic Press, New York, NY, 1995.
- [19] A. V. A. da Cruz, M. M. B. R. Vellasco, and M. A. C. Pacheco. Quantum-inspired evolutionary algorithm for numerical optimization. In *Proceedings of the IEEE Congress on Evolutionary Computation*, Vancouver, Canada, 2006.
- [20] C. Darwin. *The origin of species*. United King, 1859.
- [21] J. L. Davidson. Template learning in morphological neural nets. In *Image Algebra and Morphological Image Processing II*, volume 1568 of *Proceedings of SPIE*, pages 176–187, July 1991.
- [22] J. L. Davidson. Simulated annealing and morphological neural networks. *Proc. SPIE*, 1769:119–127, July 1992.
- [23] J. L. Davidson and G. X. Ritter. A theory of morphological neural networks. *Proc. SPIE*, 1215:378–388, July 1990.
- [24] J. L. Davidson and A. Talukder. Template identification using simulated annealing in morphology neural networks. In *Proceedings of 2nd Annual Midwest Electro-Technology Conference*, pages 64–67, Ames, IA, April 1993.
- [25] R. de A. Araújo. Swarm-based hybrid intelligent forecasting method for financial time series prediction. *Learning and Nonlinear Models*, 5(2):137–154, 2007.
- [26] R. de A. Araújo. Hybrid intelligent methodology to design translation invariant morphological operators for brazilian stock market prediction. *Neural Networks*, 23(10):1238–1251, 2010.

- [27] R. de A. Araújo. A hybrid intelligent morphological approach for stock market forecasting. *Neural Processing Letters*, 31(3):195–217, 2010.
- [28] R. de A. Araújo. A quantum-inspired evolutionary hybrid intelligent approach for stock market prediction. *International Journal of Intelligent Computing and Cybernetics*, 3:24–54, 2010.
- [29] R. de A. Araújo. Swarm-based translation-invariant morphological method for financial time series forecasting. *Information Sciences*, 180(24):4784–4805, 2010.
- [30] R. de A. Araújo. Translation invariant morphological time-lag added evolutionary forecasting method for stock market prediction. *Expert Systems with Applications*, 38(3):2835–2848, 2011.
- [31] R. de A. Araújo, A. L. I. de Oliveira, and S. C. B. Soares. A morphological-rank-linear approach for software development cost estimation. In *IEEE International Conference on Tools with Artificial Intelligence*. IEEE, 2009.
- [32] R. de A. Araújo, A. L. I. de Oliveira, and S. C. B. Soares. A covariance matrix adaptation based evolutionary methodology for phase adjustment in financial time series forecasting. In *ACM Genetic and Evolutionary Computation Conference*. ACM, 2010.
- [33] R. de A. Araújo, A. L. I. de Oliveira, and S. C. B. Soares. Hybrid evolutionary quantum inspired method to adjust time phase distortions in financial time series. In *ACM Symposium On Applied Computing*. ACM, 2010.
- [34] R. de A. Araújo, A. L. I. de Oliveira, and S. C. B. Soares. Hybrid intelligent design of morphological-rank-linear perceptrons for software development cost estimation. In *IEEE International Conference on Tools with Artificial Intelligence*. IEEE, 2010.
- [35] R. de A. Araújo, A. L. I. de Oliveira, and S. C. B. Soares. Overcoming the random walk dilemma using a covariance matrix adaptation evolutionary method. In *IEEE Conference on Systems, Man and Cybernetics*. IEEE, 2010.
- [36] R. de A. Araújo, A. L. I. de Oliveira, and S. C. B. Soares. A quantum-inspired hybrid methodology for financial time series prediction. In *IEEE International Joint Conference on Neural Networks*. IEEE, 2010.
- [37] R. de A. Araújo, R. P. de Sousa, and T. A. E. Ferreira. An intelligent hybrid approach for designing increasing translation invariant morphological operators for time series forecasting. In *ISNN (2)*, volume 4492 PART II of *Lecture Notes in Computer Science*, pages 602–611. Springer-Verlag, 2007.
- [38] R. de A. Araújo and T. A. E. Ferreira. An intelligent hybrid morphological-rank-linear method for financial time series prediction. *Neurocomputing*, 72(10-12):2507–2524, 2009.

- [39] R. de A. Araújo and T. A. E. Ferreira. A morphological-rank-linear evolutionary method for stock market prediction. *Information Sciences*, 2010. . In Press.
- [40] R. de A. Araújo, A. R. L. Junior, and T. A. E. Ferreira. Morphological-rank-linear time-lag added evolutionary forecasting method for financial time series forecasting. In *IEEE Congress on Evolutionary Computation*, pages 1340–1347. IEEE, 2008.
- [41] R. de A. Araújo, A. R. L. Junior, and T. A. E. Ferreira. A quantum-inspired intelligent hybrid method for stock market forecasting. In *IEEE Congress on Evolutionary Computation*. IEEE, 2008.
- [42] R. de A. Araújo, F. Madeiro, R. P. de Sousa, and L. F. C. Pessoa. Modular morphological neural network training via adaptive genetic algorithm for designing translation invariant operators. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 873–876, May 2006.
- [43] R. de A. Araújo, F. Madeiro, R. P. de Sousa, L. F. C. Pessoa, and T. A. E. Ferreira. An evolutionary morphological approach for financial time series forecasting. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 2467–2474, 2006.
- [44] R. de A. Araújo, F. Madeiro, T. A. F. Ferreira, R. P. de Sousa, and L. F. C. Pessoa. Improved evolutionary hybrid method for designing morphological operators. In *Proceedings of the IEEE International Conference on Image Processing*, pages 2417–2420, 2006.
- [45] R. de A. Araújo, A. L. I. Oliveira, and S. C. B. Soares. A shift-invariant morphological system for software development cost estimation. *Expert Systems with Applications*, 38(4):4162–4168, 2011.
- [46] R. de A. Araújo, G. C. Vasconcelos, and T. A. E. Ferreira. An evolutionary morphological-rank-linear approach for time series prediction. In Dipti Srinivasan and Lipo Wang, editors, *2007 IEEE Congress on Evolutionary Computation*, pages 4321–4328, Singapore, 25-28 September 2007. IEEE Computational Intelligence Society, IEEE Press.
- [47] R. de A. Araújo, G. C. Vasconcelos, and T. A. E. Ferreira. Hybrid differential evolutionary system for financial time series forecasting. In Dipti Srinivasan and Lipo Wang, editors, *2007 IEEE Congress on Evolutionary Computation*, Singapore, 25-28 September 2007. IEEE Computational Intelligence Society, IEEE Press.
- [48] B. De Baets. Fuzzy morphology: A logical approach. In B. M. Ayyub and M. M. Gupta, editors, *Uncertainty Analysis in Engineering and Science: Fuzzy Logic, Statistics, and Neural Network Approach*, pages 53–67. Kluwer Academic Publishers, Norwell, 1997.
- [49] P. S. G. de Mattos Neto. *Perturbative Theory within Hybrid Intelligent Systems for Time Series Prediction*. PhD thesis, Federal University of Pernambuco, 2008.
- [50] R. Dedekind. *Gesammelte Mathematische Werke*. Math. Annalen, Braunschweig, 1987.

- [51] T. Q. Deng and H. J. A. M. Heijmans. Grey-scale morphology based on fuzzy logic. *Journal of Mathematical Imaging and Vision*, 16(2):155–171, Mar. 2002.
- [52] M. E. R. do V. Mesquita. *Fundamentals and Applications of Fuzzy Morphological Associative Memories*. PhD thesis, State University of Campinas, 2007.
- [53] E. R. Dougherty and R. P. Loce. *Efficient Design Strategies for the Optimal Binary Digital Morphological Filter: Probabilities, Constraints, and Structuring-Element Libraries*. Marcel Decker, Inc., New York, 1993.
- [54] R. C. Eberhart and X. Hu. Human tremor analysis using particle swarm optimization. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 1927–1930, Washington, USA, 1999.
- [55] R. C. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *Proceedings of the International Symposium on Micro Machine and Human Science*, pages 39–43, Nagoya, Japan, 1995.
- [56] A. E. Eiben and J. E. Smith. *Introduction to Evolutionary Computing*. Springer–Verlag, 2003.
- [57] R. F. Engle. Autoregressive conditional heteroskedasticity with estimates of the variance of uk onflation. *Econometrica*, 50:987–1008, 1982.
- [58] S. E. Fahlman. An empirical study of learning speed in backpropagation networks. Technical report, Carnegie Mellon University, 1988.
- [59] S. E. Fahlman. Faster-learning variations of back-propagation: An empirical study. In *Proceedings of the 1998 Connectionist Models Summer School*, pages 38–51, 1989.
- [60] T. A. E. Ferreira. *A new Intelligent Hybrid Methodology for Time Serie Forecasting*. PhD thesis, Federal University of Pernambuco, 2006.
- [61] T. A. E. Ferreira, G. C. Vasconcelos, and P. J. L. Adeodato. A new intelligent system methodology for time series forecasting with artificial neural networks. In *Neural Processing Letters*, volume 28, pages 113–129, 2008.
- [62] S. Franklin. *Artificial Minds*. MIT Press, 2001.
- [63] P. D. Gader, M. A. Khabou, and A. Koldobsky. Morphological regularization neural networks. *Pattern Recognition, Special Issue on Mathematical Morphology and Its Applications*, 33(6):935–945, June 2000.
- [64] M. Gen and R. Cheng. *Genetic Algorithms and Engineering Design*. John Wiley and sons, New York, 1997.
- [65] B. Giffler. Mathematical solution of production planning and scheduling problems. Tech. rep., IBM ASDD, 1960.

- [66] D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
- [67] S. Goonatilake and S. Khebbal. *Intelligence Hybrid Systems*. John Wiley & Son, 1995.
- [68] M. Graña, J. Gallego, F. J. Torrealdea, and A. D'Anjou. On the application of associative morphological memories to hyperspectral image analysis. *Lecture Notes in Computer Science*, 2687:567–574, 2003.
- [69] G. A. Grätzer. *Lattice Theory: First Concepts and Distributive Lattices*. W. H. Freeman, San Francisco, CA, 1971.
- [70] H. Hadwiger. *Vorlesungen über Inhalt, Oberfläche und Isoperimetrie*. Springer-Verlag, Berlin, 1957.
- [71] M. Hagan and M. Menhaj. Training feedforward networks with the marquardt algorithm. *IEEE Transactions on Neural Networks*, 5(6):989–993, November 1994.
- [72] N. Hansen. The CMA evolution strategy: a comparing review. In J.A. Lozano, P. Larranaga, I. Inza, and E. Bengoetxea, editors, *Towards a new evolutionary computation. Advances on estimation of distribution algorithms*, pages 75–102. Springer, 2006.
- [73] N. R. Harvey and S. Marshall. The use of genetic algorithms in morphological filter design. *Signal Processing Image Communication*, 8:55–71, 1996.
- [74] S. Haykin. *Neural networks: A comprehensive foundation*. Prentice Hall, New Jersey, 1998.
- [75] S. Haykin. *Neural Networks and Learning Machines*. McMaster University, Canada, 2007.
- [76] H. J. A. M. Heijmans. *Morphological Image Operators*. Academic Press, New York, NY, 1994.
- [77] J. H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Michigan, 1975.
- [78] J. H. Holland. Genetic algorithms. *Scientific American*, pages 66–72, july 1992.
- [79] R. A. Jacobs. Increased rates of convergence through learning rate adaptation. *Neural Networks*, 1(4):295–308, 1998.
- [80] V. G. Kaburlasos and V. Petridis. Fuzzy lattice neurocomputing (FLN) models. *Neural Networks*, 13(10):1145–1170, 2000.
- [81] H. Kantz and T. Schreiber. *Nonlinear Time Series analysis*. Cambridge University Press, New York, NY, USA, second edition, 2003.

- [82] M. A. Khabou, P. D. Gader, and J. M. Keller. LADAR target detection using morphological shared-weight neural networks. *Machine Vision and Applications*, 11(6):300–305, May 2000.
- [83] C. Kim. Segmenting a low-depth-of-field image using morphological filters and region merging. *IEEE Transactions on Image Processing*, 14(10):1503–1511, 2005.
- [84] M. Kim and V. Pavlovic. Discriminative learning for dynamic state prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31:1847–1861, 2009.
- [85] F. H. F. Leung, H. K. Lam, S. H. Ling, and P. K. S. Tam. Tuning of the structure and parameters of the neural network using an improved genetic algorithm. *IEEE Transactions on Neural Networks*, 14(1):79–88, January 2003.
- [86] Y. W. Leung and Y. Wang. An orthogonal genetic algorithm with quantization for global numerical optimization. *IEEE Trans. Evolutionary Computation*, 5(1):41–53, 2001.
- [87] J. Liu and J. Lampinen. A fuzzy adaptive differential evolution algorithm. *Soft Comput.*, 9(6):448–462, 2005.
- [88] B. G. Malkiel. *A Random Walk Down Wall Street, Completely Revised and Updated Edition*. W. W. Norton & Company, April 2003.
- [89] P. Maragos. A representation theory for morphological image and signal processing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11:586–599, 1989.
- [90] P. Maragos. Lattice image processing: A unification of morphological and fuzzy algebraic systems. *Journal of Mathematical Imaging and Vision*, 22(2-3):333–353, May 2005.
- [91] G. Matheron. *Elements Pour une Theorie des Milieux Poreux*. Masson, Paris, 1967.
- [92] G. Matheron. *Théorie des Ensembles Aléatoires*. Ecole des Mines, Paris, 1967.
- [93] G. Matheron. *Random Sets and Integral Geometry*. Wiley, New York, 1975.
- [94] T. C. Mills. *The Econometric Modeling of Financial Time Series*. Cambridge University Press, Cambridge, 2003.
- [95] H. Minkowski. *Gesammelte Abhandlungen*. Teubner Verlag, Leipzig-Berlin, 1911.
- [96] M. Mitchell. *A Introduction to Genetic Algorithms*. MIT Press, Canbridge, 1999.
- [97] M. F. Moller. A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, 6:525–533, 1993.
- [98] J. Nin and V. Torra. Towards the evaluation of time series protection methods. *Information Sciences*, 179(11):1663–1677, 2009.

- [99] T. Ozaki. *Nonlinear Time Series Models and Dynamical Systems*, volume 5 of *HandBook of Statistics*. Noth-Holland, Amsterdam, 1985.
- [100] D. C. Pack, M. A. El-Sharkawi, and R. J. Marks II. An adaptively trained neural network. *IEEE Transactions on Neural Networks*, 2(3):334–345, 1991.
- [101] D. C. Park. Structure optimization of bilinear recurrent neural networks and its application to ethernet network traffic prediction. *Information Sciences*, 2010. . In Press.
- [102] D. B. Percival and A. T. Walden. *Spectral Analysis for Physical Applications – Multitaper and Conventional Univariate Techniques*. Cambridge University Press, New York, 1998.
- [103] M. Perus. Neuro-quantum parallelism in brain-mind and computers. *Informatika (Slovenia)*, 20(2), 1996.
- [104] L. F. C. Pessoa and P. Maragos. Morphological rank neural networks and their adaptive optimal design for image processing. In *Proc. of the IEEE Intl Conference on Acoustics, Speech and Signal Processing*, pages 3398–3401, Atlanta, 1996.
- [105] L. F. C. Pessoa and P. Maragos. MRL-filters: A general class of nonlinear systems and their optimal design for image processing. *IEEE Transactions on Image Processing*, 7:966–978, 1998.
- [106] L. F. C. Pessoa and P. Maragos. Neural networks with hybrid morphological rank linear nodes: a unifying framework with applications to handwritten character recognition. *Pattern Recognition*, 33:945–960, 2000.
- [107] V. Petridis and V. G. Kaburlasos. Fuzzy lattice neural network (FLNN): a hybrid model for learning. *IEEE Transactions on Neural Networks*, 9(5):877–890, sep 1998.
- [108] D. T. Pham and D. Karaboga. *Intelligent Optimization Techniques: Genetic Algorithms, Tabu Search, Simulated Annealing and Neural Networks*. Springer, New York, 2000.
- [109] H. Pi and C. Peterson. Finding the embedding dimension and variable dependences in time series. *Neural Computation*, 6:509–520, 1994.
- [110] L. Prechelt. Proben1: A set of neural network benchmark problems and benchmarking rules. Technical Report 21/94, 1994.
- [111] M. B. Priestley. *Non-Linear and Non-Stationary Time Series Analysis*. Academic Press, 1988.
- [112] B. Raducanu, M. Graña, and X. F. Albizuri. Morphological scale spaces and associative morphological memories: Results on robustness and practical applications. *Journal of Mathematical Imaging and Vision*, 19(2):113–131, 2003.

- [113] B. Raducanu, M. Graña, and P. Sussner. Advances in mobile robot self-localization using morphological neural networks. In *Proceedings of IFAC 2001*, Cheju Island, South Korea, May 2001.
- [114] B. Raducanu, M. Graña, and P. Sussner. Morphological neural networks for vision based self-localization. In *Proceedings of the International Conference on Robotics and Automation*, Seoul, South Korea, May 2001.
- [115] A. Rahimi, B. Recht, and T. Darrell. Learning to transform time series with a few examples. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29:1759–1775, 2007.
- [116] T. S. Rao and M. M. Gabr. *Introduction to Bispectral Analysis and Bilinear Time Series Models*, volume 24 of *Lecture Notes in Statistics*. Springer, Berlin, 1984.
- [117] M. Riedmiller and H. Braun. A direct adaptive method for faster backpropagation learning: The rprop algorithm. In *Proceedings of the IEEE Int. Conf. on Neural Networks (ICNN)*, pages 586–591, San Francisco, 1993.
- [118] G. X. Ritter, L. Iancu, and M. S. Schmalz. A new auto-associative memory based on lattice algebra. *Lecture Notes in Computer Science*, 3287:148–155, 2004.
- [119] G. X. Ritter, D. Li, and J. N. Wilson. Image algebra and its relationship to neural networks. In *Technical Symposium Southeast on Optics, Electro-Optics, and Sensors*, Proceedings of SPIE, pages 90–101, Orlando, FL, mar 1989.
- [120] G. X. Ritter and P. Sussner. An introduction to morphological neural networks. In *Proceedings of the 13th International Conference on Pattern Recognition*, pages 709–717, Vienna, Austria, 1996.
- [121] G. X. Ritter and P. Sussner. Associative memories based on lattice algebra. In *Computational Cybernetics and Simulation*, Orlando, Florida, 1997. 1997 IEEE International Conference on Systems, Man, and Cybernetics.
- [122] G. X. Ritter and P. Sussner. Morphological perceptrons. In *ISAS'97, Intelligent Systems and Semiotics*, Gaithersburg, Maryland, 1997.
- [123] G. X. Ritter and G. Urcid. Lattice algebra approach to single-neuron computation. *IEEE Transactions on Neural Network*, 14(2):282–295, March 2003.
- [124] G. X. Ritter, G. Urcid, and M. S. Schmalz. Autonomous single-pass endmember approximation using lattice auto-associative memories. *Neurocomputing*, 72(10–12):2101–2110, 2009.
- [125] G. X. Ritter and J. N. Wilson. *Handbook of Computer Vision Algorithms in Image Algebra*. CRC Press, Boca Raton, 2 edition, 2001.

- [126] G. X. Ritter, J. N. Wilson, and J. L. Davidson. Image algebra: An overview. *Computer Vision, Graphics, and Image Processing*, 49(3):297–331, March 1990.
- [127] C. Ronse. Why mathematical morphology needs complete lattices. *Signal Processing*, 21(2):129–154, October 1990.
- [128] D. E. Rumelhart and J. L. McClelland. *Parallel Distributed Processing, Explorations in the Microstructure of Cognition*, volume 1 & 2. MIT Press, 1987.
- [129] P. Salembier. Adaptive rank order based filters. *Signal Process.*, 27(1):1–25, 1992.
- [130] P. Salembier. Structuring element adaptation for morphological filters. *Journal for Visual Communication and Image Representation*, 3(2):115–136, June 1992.
- [131] R. Savit and M. Green. Time series and dependent variables. *Physica D*, 50:95–116, 1991.
- [132] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, London, 1982.
- [133] J. Serra. *Image Analysis and Mathematical Morphology, Volume 2: Theoretical Advances*. Academic Press, New York, 1988.
- [134] Y. Shi and R. C. Eberhart. A modified particle swarm optimizer. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 69–73, Anchorage, AK, 1998.
- [135] Y. Shi and R. C. Eberhart. Empirical study of particle swarm optimization. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 1945–1950, Washington, USA, 1999.
- [136] R. Sitte and J. Sitte. Neural networks approach to the random walk dilemma of financial time series. *Applied Intelligence*, 16(3):163–171, May 2002.
- [137] A. Sobania, J. Evans, and O. Paul. Morphological corner detector using paired triangular structuring elements. *Pattern Recognition*, 38(7):1087–1098, 2005.
- [138] P. Soille. *Morphological Image Analysis*. Springer Verlag, Berlin, 1999.
- [139] R. P. Sousa. *Design of translation invariant operators via neural network training*. PhD thesis, UFPB, Campina Grande, Brazil, 2000.
- [140] R. P. Sousa, J. M. Carvalho, F. M. Assis, and L. F. C. Pessoa. Designing translation invariant operations via neural network training. In *Proc. of the IEEE Intl Conference on Image Processing*, Vancouver, Canada, 2000.
- [141] S. R. Sternberg. Parallel architecture for image processing. In *Proceedings of the Third International IEEE Compsac*, Chicago, 1979.

- [142] S. R. Sternberg. Overview of image algebra and related issues. In S. Levialdi, editor, *Integrated Technology for Parallel Image Processing*. Academic Press, London, 1985.
- [143] S. R. Sternberg. Grayscale morphology. *Computer Vision, Graphics and Image Processing*, 35:333–355, 1986.
- [144] R. Storn. System design by constraint adaptation and differential evolution. *IEEE Trans. on Evolutionary Computation*, 3(1):22–34, 1999.
- [145] J. Sun, Q. Zhang, and E. P. K. Tsang. De/eda: a new evolutionary algorithm for global optimization. *Inf. Sci. Inf. Comput. Sci.*, 169(3-4):249–262, 2005.
- [146] P. Sussner. *Matrix decomposition in minimax algebra and applications in image processing*. PhD thesis, University of Florida, 1996.
- [147] P. Sussner. Morphological perceptron learning. In *Proceedings of the IEEE International Symposium on Intelligent Control*, pages 477–482, Gaithersburg, MD, 1998.
- [148] P. Sussner and E. L. Esmi. Constructive morphological neural networks: some theoretical aspects and experimental results in classification. In J. Kacprzyk, editor, *Constructive Neural Networks*, Studies in Computational Intelligence, pages 123–144. Springer Verlag, Heidelberg, Germany, 2009.
- [149] P. Sussner and E. L. Esmi. An introduction to morphological perceptrons with competitive learning. In *Proceedings of the International Joint Conference on Neural Networks*, pages 3024–3031, Atlanta, GA, 2009.
- [150] P. Sussner and E. L. Esmi. Morphological perceptrons with competitive learning: Lattice-theoretical framework and constructive learning algorithm. *Information Sciences*, 181(10):1929–1950, 2011.
- [151] P. Sussner, R. Miyasaki, and M. E. Valle. An introduction to parameterized IFAM models with applications in prediction. In *Proceedings of IFSA-EUSFLAT 2009*, pages 3024–3031, Lisbon, Portugal, 2009.
- [152] P. Sussner and M. E. Valle. A brief account of the relations between gray-scale mathematical morphologies. In *Proceedings of SIBGRAPI 2005*, pages 79–86, Natal, RN, Brazil, 2005.
- [153] P. Sussner and M. E. Valle. Morphological and certain fuzzy morphological associative memories for classification and prediction. In V. G. Kaburlassos and G. X. Ritter, editors, *Computational Intelligence Based on Lattice Theory*, volume 67, pages 149 – 173. Springer Verlag, Heidelberg, Germany, 2007.
- [154] P. Sussner and M. E. Valle. Classification of fuzzy mathematical morphologies based on concepts of inclusion measure and duality. *Journal of Mathematical Imaging and Vision*, 32(2):139–159, 2008.

- [155] F. Takens. Detecting strange attractor in turbulence. In A. Dold and B. Eckmann, editors, *Dynamical Systems and Turbulence*, volume 898 of *Lecture Notes in Mathematics*, pages 366–381, New York, 1980. Springer-Verlag.
- [156] N. Tanaka, H. Okamoto, and M. Naito. Estimating the active dimension of the dynamics in a time series based on a information criterion. *Physica D*, 158:19–31, 2001.
- [157] I. Terol-Villalobos and J. Mendiola-Santibañez. Transformations with reconstruction criteria: Image segmentation and filtering. *Computational Imaging and Vision*, 30:75–84, september 2005.
- [158] J. T. Tsai, T. K. Liu, and J. H. Chou. Hybrid taguchi-genetic algorithm for global numerical optimization. *IEEE Trans. Evolutionary Computation*, 8(4):365–377, 2004.
- [159] Z. Tu and Y. Lu. A Robust Stochastic Genetic Algorithm (StGA) for Global Numerical Optimization. *IEEE Transactions on Evolutionary Computation*, 8(5):456–470, 2004.
- [160] M. E. Valle and P. Sussner. A general framework for fuzzy morphological associative memories. *Fuzzy Sets and Systems*, 159(7):747–768, 2008.
- [161] F. Vandenbergh and A. P. Engelbrecht. Cooperative learning in neural networks using particle swarm optimizers. In *South African Comput. J.*, volume 26, pages 84–90, 2000.
- [162] F. Vandenbergh and A. P. Engelbrecht. A cooperative approach to particle swarm optimization. *IEEE Trans. Evolutionary Computation*, 8(3):225–239, 2004.
- [163] D. Vargas-Vázquez, J. Crespo, and V. Maojo. Morphological image reconstruction with criterion from labelled markers. *Discrete Geometry for Computer Imagery – Lecture Notes in Computer Science*, 2886:475–484, 2003.
- [164] J. Vesterstroem and R. Thomsen. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. In *Proceedings of the IEEE Congress on Evolutionary Computation*, volume 2, pages 1980–1987, 2004.
- [165] G. Zhang, B. E. Patuwo, and M. Y. Hu. Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting*, 14:35–62, 1998.
- [166] G. P. Zhang and D. M. Kline. Quarterly time-series forecasting with neural networks. *IEEE Transactions on Neural Networks*, 18(6):1800–1814, Nov. 2007.
- [167] U. Zimmermann. *Linear and Combinatorial Optimization of Ordered Algebraic Structures*. North-Holland, Amsterdam, 1981.