



Pós-Graduação em Ciência da Computação

**METODOLOGIA PARA SELEÇÃO DE  
CARACTERÍSTICAS E DO CONJUNTO DE  
TREINAMENTO PARA PREVISÃO DE SÉRIES  
TEMPORAIS**

**Por**

***Gustavo Henrique Tavares Ribeiro***

**Dissertação de Mestrado**



Universidade Federal de Pernambuco  
posgraduacao@cin.ufpe.br  
[www.cin.ufpe.br/~posgraduacao](http://www.cin.ufpe.br/~posgraduacao)

RECIFE, AGOSTO/2012



UNIVERSIDADE FEDERAL DE PERNAMBUCO  
CENTRO DE INFORMÁTICA  
PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

GUSTAVO HENRIQUE TAVARES RIBEIRO

METODOLOGIA PARA SELEÇÃO DE CARACTERÍSTICAS E DO CONJUNTO DE  
TREINAMENTO PARA PREVISÃO DE SÉRIES TEMPORAIS

*ESTE TRABALHO FOI APRESENTADO À PÓS-GRADUAÇÃO EM  
CIÊNCIA DA COMPUTAÇÃO DO CENTRO DE INFORMÁTICA DA  
UNIVERSIDADE FEDERAL DE PERNAMBUCO COMO REQUISITO  
PARCIAL PARA OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIA DA  
COMPUTAÇÃO.*

Orientador: Prof. Dr. George Darmiton da Cunha  
Cavalcanti  
Co-orientador: Prof. Dr. Tsang Ing Ren

RECIFE, AGOSTO/2012

**Catálogo na fonte**  
**Bibliotecária Jane Souto Maior, CRB4-571**

**Ribeiro, Gustavo Henrique Tavares Ribeiro**

**Metodologia para seleção de características e do conjunto de treinamento para previsão de séries temporais. / Gustavo Henrique Tavares Ribeiro. - Recife: O Autor, 2012.**

**xiii, 79 folhas: fig., tab.**

**Orientador: George Darmiton da Cunha Cavalcanti.**

**Dissertação (mestrado) - Universidade Federal de Pernambuco. CIn, Ciência da Computação, 2012.**

**Inclui bibliografia.**

**1. Ciência da Computação. 2. Estatística. 3. Séries temporais. I. Cavalcanti, George Darmiton da Cunha Cavalcanti (orientador). II. Título.**

**004**

**CDD (23. ed.)**

**MEI2013 – 068**

Dissertação de Mestrado apresentada por **Gustavo Henrique Tavares Ribeiro** à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, sob o título “**Metodologia para seleção de Características e do conjunto de Treinamento para Previsão de Séries Temporais**” orientada pelo **Prof. George Darmiton da Cunha Cavalcanti** e aprovada pela Banca Examinadora formada pelos professores:

---

Prof. Adriano Lorena Inácio Oliveira  
Centro de Informática / UFPE

---

Prof. Tiago Alessandro Espinola Ferreira  
Departamento de Estatística e Informática / UFRPE

---

Prof. George Damirton da Cunha Cavalcanti  
Centro de Informática / UFPE

Visto e permitida a impressão.  
Recife, 22 de agosto de 2012

---

**Prof. Nelson Souto Rosa**

Coordenador da Pós-Graduação em Ciência da Computação do  
Centro de Informática da Universidade Federal de Pernambuco.

*Dedico este trabalho à minha Família, em especial à minha mãe e ao meu pai pela educação que recebi, à minha irmã por todo o apoio ao longo dos estudos realizados e à minha noiva pela compreensão e apoio em todos os momentos.*

## AGRADECIMENTOS

- Agradeço aos meus pais, Arisberto Ribeiro da Silva e Maria do Carmo Tavares Ribeiro, pela educação que me foi dada ao longo da vida que me possibilitou dar mais esse passo na vida acadêmica e profissional.
- A minha irmã, Maria Carla Tavares Ribeiro, pelo palavras de apoio e pelo esforço realizado para me ajudar nos momentos necessários.
- A minha noiva, Juliana Cristina de Araújo, companheira de todas as horas, ombro amigo. Pelos momentos de alegria, amor, descontração, paciência e principalmente pelo apoio incondicional em todos os momentos.
- Ao meu orientador, Prof. Dr. George Darmiton da Cunha Cavalcanti e ao meu Co-orientador, Prof. Dr. Tsang Ing Ren, pelas orientações sempre relevantes, pelo aprendizado, entusiasmo, apoio e paciência.
- Ao aluno de Doutorado do Centro de Informática (Cin), Paulo Salgado Gomes de Mattos Neto, pela contribuição em parte dos trabalhos desenvolvidos nesta dissertação.

*Se queres prever o futuro, estuda o passado.*

—551 A.C 479 A.C (Confúcio)

## RESUMO

Séries Temporais são dados que podem ser coletados em diversas situações do cotidiano, preços diários de uma ação financeira e o índice pluviométrico mensal de uma cidade são bons exemplos desses dados. A previsão de valores futuros de uma série temporal é uma área de pesquisa muito útil como suporte à decisões. Para executar essa tarefa, são necessários um conjunto de dados históricos e um modelo de previsão.

As redes neurais artificiais são modelos poderosos e têm sido utilizados em pesquisas que envolvem previsão. Apesar de já serem utilizadas na resolução de inúmeros problemas, configurar uma rede neural nem sempre é uma atividade simples. Esse fato se deve ao número de graus de liberdade que existem para sua configuração. Por isso, muitas pesquisas propõem metodologias para otimizar os parâmetros da rede de acordo com a série alvo da previsão. No entanto, uma tarefa mais importante é a seleção das características de entrada da rede.

Este trabalho reporta uma metodologia de previsão que tem como foco a seleção de características para previsão de séries temporais. Para realizar essa tarefa, foi utilizado um algoritmo de otimização até então não testado para essa finalidade. Sabe-se que em geral, esses algoritmos são custosos, foi utilizada uma abordagem de seleção do conjunto de treinamento através de análise de similaridade entre janelas da série que ajuda na redução do esforço computacional.

Sabendo da importância de uma seleção correta das características, diversos trabalhos propõem metodologias de seleção. Dentre essas, podemos destacar os filtros, wrappers, a combinação dessas últimas e algoritmos de otimização, tais como: algoritmos genéticos, otimização baseada em enxames de partículas e evolução diferencial.

Esta dissertação descreve uma metodologia capaz de realizar a previsão de séries originárias de diferentes contextos. A metodologia é composta por três etapas: o tratamento das séries, a seleção do conjunto de treinamento através de uma abordagem inovadora e a seleção das características por meio de um algoritmo de otimização. O seu desempenho foi avaliado em oito séries temporais e os modelos utilizados para as previsões foram as redes neurais do tipo *Multilayer Perceptron* e a *Support Vector Regression*.

Portanto, é proposta uma metodologia híbrida, inovadora, que não exige análise de especialista para fazer suas previsões e pode ser totalmente automatizada. Além disso, mostrou-se capaz de realizar previsões de diferentes séries temporais com precisão, alcançando resultados superiores aos reportados na literatura.

**Palavras-chave:** Séries Temporais, seleção de características, seleção do conjunto de treinamento.

## ABSTRACT

An Time Series is a data that can be collected in various situations, daily prices of stock market and monthly pluviometric index of a city are good examples of such data. The time series forecasting is a very useful area of research for decisions support. To perform this task its necessary a set of historical data and a prediction model.

The artificial neural network models are powerful and have been used in research involving prediction. Despite already being used in solving many problems, configure a neural network is not always a simple activity. This is due to the number of freedom degrees that exist for your configuration. Therefore, many researches propose methodologies to optimize the network parameters. However, a more important task is selection of the input of the network.

This paper reports a forecasting methodology that focuses on feature selection for time series forecasting. To perform this task, were used an optimization algorithm that has not been used for this purpose. It is known that in general, these algorithms are expensive, to reduce the computational effort were used a selection approach of the training set using similarity analysis between windows series.

Knowing the importance of a correct feature selection, several studies proposes approaches of feature selection. Among these, can highlight the filters, wrappers, the combination of filters and wrappers, optimization algorithms such as genetic algorithms, optimization based on particle swarm and differential evolution.

This paper describes a methodology of forecasting that is capable to deal with time series of different contexts. The methodology consists of three steps: the preprocessing of time series, the train set selection through an innovative approach and feature selection through an optimization algorithm. Its performance was evaluated in eight series and the prediction models were the *Multilayer Perceptron* neural network and the *Support Vector Regression*.

Therefore, were proposed a hybrid methodology, innovative, that no requiring expert analysis to make their forecastings and can be fully automated. Furthermore, this approach produced accurated forecastings, achieving better results than previous researchs.

**Keywords:** Time series, prediction models, feature selection, train set selection.

# SUMÁRIO

<b>Capítulo 1—Introdução</b>	1
1.1 Motivação . . . . .	1
1.2 Objetivos . . . . .	3
1.3 Estrutura da dissertação . . . . .	4
<b>Capítulo 2—Séries Temporais</b>	5
2.1 Conceitos básicos . . . . .	5
2.1.1 Estacionariedade, tendência e sazonalidade . . . . .	6
2.2 Previsão de Séries Temporais . . . . .	8
2.3 Modelos para previsão de séries temporais . . . . .	9
2.3.1 Redes Neurais Artificiais . . . . .	9
2.3.2 Máquinas de Vetor de Suporte para Regressão . . . . .	11
2.4 Tratamentos dos dados . . . . .	11
2.4.1 Transformação de Box-Cox . . . . .	12
2.4.2 Diferenciação da série . . . . .	12
2.5 Comentários Finais . . . . .	13
<b>Capítulo 3—Seleção de Características para séries temporais</b>	14
3.1 O problema . . . . .	14
3.2 Técnicas para seleção de características em séries temporais . . . . .	15
3.2.1 Filtros . . . . .	15
3.2.2 Wrappers . . . . .	16
3.3 Frankenstein’s PSO . . . . .	17
3.4 Comentários Finais . . . . .	22
<b>Capítulo 4—Seleção do conjunto de treinamento</b>	23
4.1 Dynamic Time Warping . . . . .	23
4.1.1 Fast Dynamic Time Warping . . . . .	26
4.2 Comentários Finais . . . . .	27
<b>Capítulo 5—Método Proposto</b>	29
5.1 Arquitetura Utilizada . . . . .	29
5.1.1 Seleção do conjunto de treinamento . . . . .	29
5.1.2 Tratamento dos dados . . . . .	31

5.1.3	Seleção de características . . . . .	31
5.1.4	Métricas para avaliação das previsões . . . . .	32
5.2	Comentários Finais . . . . .	33
<b>Capítulo 6—Experimentos</b>		<b>34</b>
6.1	Etapas de um experimento . . . . .	34
6.2	Séries temporais utilizadas . . . . .	36
6.3	Resultados . . . . .	42
6.3.1	Experimentos com a série ESTSP1 . . . . .	44
6.3.2	Experimentos com a série ESTSP2 . . . . .	46
6.3.3	Experimentos com a série Pollution . . . . .	50
6.3.4	Experimentos com a série Paper . . . . .	53
6.3.5	Experimentos com a série PETR4 . . . . .	58
6.3.6	Experimentos com a série VALE5 . . . . .	61
6.3.7	Experimentos com a série PCO . . . . .	64
6.3.8	Experimentos com a série Umidade . . . . .	67
6.4	Testes de Hipótese . . . . .	70
6.5	Comentários Finais . . . . .	72
<b>Capítulo 7—Conclusão</b>		<b>73</b>
7.1	Contribuições . . . . .	74
7.2	Limitações do trabalho . . . . .	74
7.3	Trabalhos Futuros . . . . .	75

## LISTA DE FIGURAS

1.1	Algumas áreas nas quais a previsão de séries temporais pode ser útil . . .	1
2.1	Série de Temperaturas médias mensais da cidade de Cananéia de 1976 a 1985 . . . . .	5
2.2	Série de fechamento da ação da Petrobrás de 2007 a 2012 . . . . .	6
2.3	Comportamento de uma série estacionária . . . . .	6
2.4	Consumo mensal de energia no Espírito Santo de 1968 a 1979 . . . . .	7
2.5	Índice pluviométrico semanal da cidade de Lavras em Minas Gerais de 1966 a 1997 . . . . .	7
2.6	Observações de uma série temporal com previsões de origem $t$ e horizontes de previsão iguais a um, dois e h. . . . .	8
2.7	Rede MLP tendo como entrada valores passados da série e a saída é um valor futuro. Arquitetura composta por uma camada de entrada, uma camada intermediária e uma camada de saída . . . . .	10
2.8	Série Não estacionária x Série Estacionária . . . . .	13
3.1	Funcionamento de um wrapper . . . . .	16
3.2	Procedimento de poda aplicado numa topologia de partículas . . . . .	19
3.3	Fluxo de execução do FPSO . . . . .	20
4.1	Similaridade através da distância Euclidiana . . . . .	26
4.2	Similaridade através do DTW . . . . .	26
4.3	Quatro diferentes resoluções avaliadas durante execução do FastDTW . . . . .	28
5.1	Arquitetura da metodologia de previsão . . . . .	29
5.2	Divisão empregada em cada série temporal para seleção do conjunto de treinamento . . . . .	30
5.3	Detalhamento sobre a montagem do conjunto de treinamento . . . . .	30
6.1	Abordagem de escolha da janela para o <i>FastDTW</i> . . . . .	35
6.2	Formatação dos experimentos . . . . .	36
6.3	Representação gráfica da série ESTSP1 . . . . .	37
6.4	Representação gráfica da série ESTSP2 . . . . .	37
6.5	Representação gráfica da série Pollution . . . . .	37
6.6	Representação gráfica da série Paper . . . . .	38
6.7	Representação gráfica da série PETR4 . . . . .	38
6.8	Representação gráfica da série VALE5 . . . . .	39
6.9	Representação gráfica da série PCO . . . . .	39

6.10	Representação gráfica da série umidade . . . . .	40
6.11	Estrutura básica da metodologia proposta . . . . .	43
6.12	Média $\times$ Desvio para a série ESTSP1 . . . . .	44
6.13	Gráfico da média amostral para a série ESTSP1 . . . . .	45
6.14	Gráfico da autocovariância amostral para a série ESTSP1 . . . . .	45
6.15	Número de Padrões de treinamento antes e depois da seleção na Série ESTSP1 . . . . .	47
6.16	Gráfico previsto $\times$ real para a série ESTSP1 . . . . .	47
6.17	Gráfico desvio $\times$ média na série ESTSP2 . . . . .	48
6.18	Gráfico da média amostral para a série ESTSP2 . . . . .	49
6.19	Gráfico da autocovariância amostral para a série ESTSP2 . . . . .	49
6.20	Número de Padrões de treinamento antes e depois da seleção na Série ESTSP2 . . . . .	51
6.21	Gráfico previsto $\times$ real para série ESTSP2 . . . . .	51
6.22	Gráfico desvio $\times$ média na série Pollution . . . . .	51
6.23	Gráfico da autocovariância amostral para a série Pollution após transformação logarítmica . . . . .	52
6.24	Gráfico da média amostral para a série Pollution após transformação logarítmica	53
6.25	Número de Padrões de treinamento antes e depois da seleção na Série Pollution . . . . .	54
6.26	Gráfico previsto $\times$ real para a série Pollution . . . . .	54
6.27	Gráfico desvio $\times$ média na série Paper . . . . .	55
6.28	Gráfico da média amostral para a série Paper após transformação logarítmica	55
6.29	Gráfico da autocovariância amostral para a série Paper após transformação logarítmica . . . . .	56
6.30	Número de Padrões de treinamento antes e depois da seleção na Série Paper	57
6.31	Gráfico previsto $\times$ real para a série Paper . . . . .	58
6.32	Gráfico média $\times$ desvio para a série PETR4 . . . . .	58
6.33	Gráfico da média amostral para a série PETR4 . . . . .	59
6.34	Gráfico da autocovariância amostral para a série PETR4 . . . . .	60
6.35	Número de Padrões de treinamento antes e depois da seleção na Série PETR4	61
6.36	Gráfico previsto $\times$ real na série PETR4 . . . . .	61
6.37	Gráfico média $\times$ desvio para a série VALE5 . . . . .	62
6.38	Gráfico da média amostral para a série VALE5 . . . . .	62
6.39	Gráfico da autocovariância amostral para a série VALE5 . . . . .	63
6.40	Número de Padrões de treinamento antes e depois da seleção na Série VALE5	64
6.41	Gráfico previsto $\times$ real para a série VALE5 . . . . .	64
6.42	Gráfico média $\times$ desvio para a série PCO . . . . .	65
6.43	Gráfico da média amostral para a série PCO após a transformação logarítmica	65
6.44	Gráfico da autocorrelação amostral para a série PCO após a transformação logarítmica . . . . .	66
6.45	Número de Padrões de treinamento antes e depois da seleção na Série PCO	67
6.46	Gráfico previsto $\times$ real para a série Poluição . . . . .	67

6.47	Gráfico média $\times$ desvio para a série Umidade . . . . .	68
6.48	Gráfico da média amostral para série Umidade . . . . .	68
6.49	Gráfico da autocorrelação amostral para a série Umidade . . . . .	69
6.50	Número de Padrões de treinamento antes e depois da seleção na Série Umidade . . . . .	70
6.51	Gráfico previsto $\times$ real para a série Umidade . . . . .	71

## LISTA DE TABELAS

4.1	Tabela $H_{16,16}$ criada pelo DTW . . . . .	25
6.1	Propriedades das séries utilizadas neste trabalho . . . . .	40
6.2	Parâmetros dos modelos de previsão e os valores que foram investigados nesta dissertação . . . . .	41
6.3	Parâmetros do FPSO . . . . .	41
6.4	Tamanhos dos subconjuntos de dados usados no DTW para cada série temporal . . . . .	42
6.5	Legenda das metodologias . . . . .	42
6.6	Resultados para a série ESTSP1 . . . . .	46
6.7	Variância da média amostral e da autocovariância para a série ESTSP2 após transformação logarítmica . . . . .	48
6.8	Resultados para a série ESTSP2 . . . . .	50
6.9	Variância da média amostral e da autocovariância para a série Pollution após transformação logarítmica . . . . .	52
6.10	Resultados para a série Pollution . . . . .	53
6.11	Variância da média amostral e da autocovariância para a série Paper após transformação logarítmica . . . . .	56
6.12	Resultados para a série Paper . . . . .	57
6.13	Variância da média amostral e da autocovariância para a série PETR4 . . . . .	59
6.14	Resultados para a série PETR4 . . . . .	60
6.15	Variância da média amostral e da autocovariância para a série VALE5 . . . . .	63
6.16	Resultados para a série VALE5 . . . . .	63
6.17	Variância da média amostral e da autocovariância para a série PCO . . . . .	66
6.18	Resultados para a série PCO . . . . .	66
6.19	Variância da média amostral e da autocovariância para a série Umidade . . . . .	69
6.20	Resultados para a série Umidade . . . . .	69
6.21	Resultados dos testes de hipótese . . . . .	72

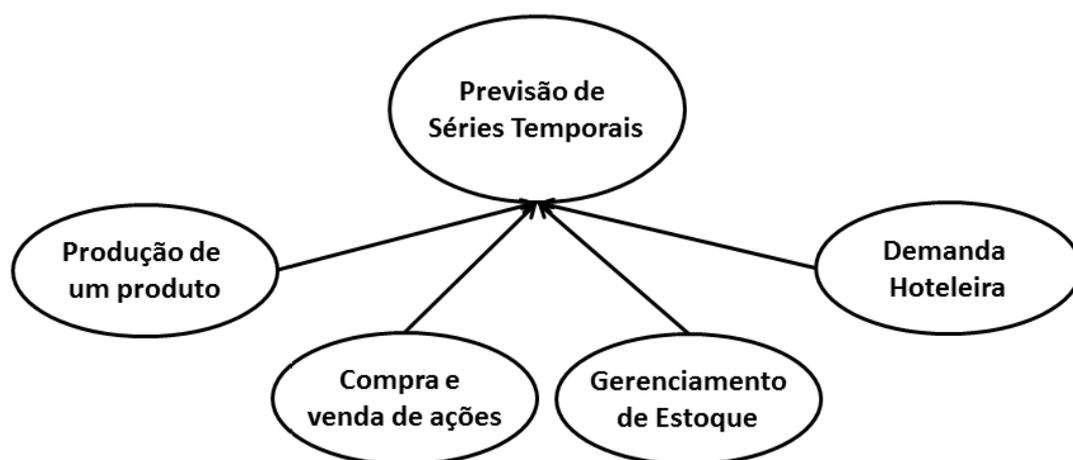
# CAPÍTULO 1

## INTRODUÇÃO

Este capítulo apresenta a motivação para o desenvolvimento desta dissertação. Cita situações nas quais a previsão de séries temporais é útil, além de formas de realizar essa tarefa que foram utilizadas nos últimos anos e apresenta os objetivos deste trabalho.

### 1.1 MOTIVAÇÃO

Uma série temporal é um conjunto de dados proveniente de um evento qualquer e que são ordenadas no tempo. A previsão de séries temporais é alvo de inúmeras pesquisas, pois trata-se de uma atividade que pode ser muito usada para dar suporte às decisões que dependem dos valores previstos. Várias situações podem ser exemplificadas para mostrar como essa tarefa pode ser importante. Por exemplo, é útil prever: o consumo médio de etanol de um país para planejar a produção deste produto; os valores futuros de ações para auxiliar investidores em suas operações de compra e venda; a quantidade de carros vendidos mensalmente para auxiliar o gerenciamento do estoque de uma montadora; o número de turistas que visitarão uma cidade para que a rede hoteleira se organize para atender a demanda. A Figura 1.1 mostra algumas áreas nas quais a previsão pode ser de grande utilidade.



**Figura 1.1** Algumas áreas nas quais a previsão de séries temporais pode ser útil

Basicamente, para fazer uma previsão são necessários dados históricos da série e um modelo de previsão. Esse último utiliza os dados históricos para realizar a previsão. Muitos modelos de previsão já foram investigados, podem ser citados os tradicionais auto-regressivo (AR) [1, 2], médias móveis (MA) [1, 2] e os auto-regressivos integrados e de médias móveis (ARIMA) [1, 2]. Merecem destaque também as técnicas de computação

inteligente, tais como redes neurais artificiais (RNA) [3] e máquinas de vetores de suporte (SVM) [3, 4].

As redes neurais artificiais são modelos que apresentam um grande potencial para resolver problemas tanto de classificação quanto de regressão [3, 5]. Apesar disso, ainda deixam de ser utilizadas em algumas aplicações práticas para previsão de séries temporais, principalmente quando não se sabe nada sobre a origem das séries temporais alvo das previsões, como acontece nas séries utilizadas em competições [6]. Esse fato pode ser causado pela dificuldade de configurar tais modelos, pois existem muitos parâmetros que devem ser ajustados para a obtenção de previsões corretas. Sendo assim, é importante configurar corretamente os parâmetros da RNA para a obtenção de modelos que produzam previsões ajustadas [5, 3, 6, 7].

Muitos trabalhos na área de previsão de séries temporais se dedicam à investigação de parâmetros ótimos para os modelos [8, 6, 9]. Outros trabalhos fazem combinação [10, 11, 12] e seleção dos melhores modelos [13]. Ao realizar previsão de séries temporais com redes neurais, um aspecto muito importante e que merece atenção é a seleção correta das entradas para a rede [14, 15]. Em séries temporais, as entradas são conhecidas como *lags* ou características.

A seleção de características para séries temporais pode ser realizada de várias formas, dentre as quais merecem destaque: seleção através de experimentos, filtros [6, 16], técnicas *wrapper* [16, 6, 17, 18] e algoritmos de otimização [19, 8, 20, 9].

Neste trabalho foi utilizado o algoritmo *Frankenstein's Particle Swarm Optimization* (FPSO) para seleção de características. Ele pertence à família de algoritmos baseados em enxame de partículas e é composto por partes de outros algoritmos. Vale salientar que não foram encontrados trabalhos que o utilizaram anteriormente para seleção de características em séries temporais. Além disso, o FPSO apresenta melhor desempenho na tarefa de otimização de funções complexas em comparação com outros algoritmos da sua família [21] e na tarefa de seleção de características em comparação com o PSO original [19]. A vantagem de utilizar um algoritmo de otimização para essa tarefa é que ele pode encontrar um subconjunto ótimo de características sem a necessidade de testar todos os subconjuntos possíveis. Além disso, algoritmos de otimização não fazem qualquer suposição a respeito do comportamento da série alvo da previsão. Uma desvantagem desses algoritmos é que eles possuem alto custo computacional, pois quanto maior a série temporal mais difícil é a seleção das características. Para reduzir essa limitação, um procedimento para seleção do conjunto de treinamento foi utilizado. Essa tarefa ajuda na convergência do FPSO e dos modelos de previsão empregados.

A seleção do conjunto de treinamento foi realizada através da análise de similaridade entre janelas de dados. Para isso, foi utilizada uma variação do algoritmo *Dynamic Time Warping* (DTW) [22, 23] que é chamada de *Fast Dynamic Time Warping* (*Fast-DTW*) [24]. Esse algoritmo é capaz de encontrar resultados similares ao do DTW, mas realizando menos operações.

As previsões foram feitas com duas técnicas de computação inteligente, *Support Vector Regression* (SVR) e *Multilayer Perceptron* (MLP). As duas técnicas foram escolhidas por terem sido capazes de produzir bons resultados em publicações anteriores a este trabalho [25, 26, 19, 16].

Existem inúmeras métricas para avaliar o desempenho de modelos de previsão, cada uma com sua particularidade. Neste trabalho quatro métricas foram utilizadas, o MSE (*Mean Squared Error*), o RMSE (*Root Mean Squared Error*), o MAPE (*Mean Absolute Percentage Error*) e a estatística U de THEIL. Todas essas métricas foram escolhidas para fins de comparação com resultados reportados por outros trabalhos.

## 1.2 OBJETIVOS

Este trabalho possui três objetivos, são eles:

- Investigar o desempenho do algoritmo de otimização FPSO na seleção de características para previsão de séries temporais;
- Propor uma abordagem de seleção do conjunto de treinamento, visando a redução do esforço computacional gasto pelo sistema de previsão.
- Propor um sistema híbrido composto pelos seguintes procedimentos: tratamento das séries, seleção do conjunto de treinamento e seleção de características. Neste trabalho o conjunto de características foi considerado como as entradas dos modelos de previsão, tais entradas compõem um padrão que também é formado pela saída desejada. Já o conjunto de treinamento é um conjunto de padrões que são formados por suas características e suas saídas desejadas.

Existem inúmeros trabalhos que investigam técnicas de seleção de características. A corretude da seleção depende do algoritmo empregado na seleção, da série temporal e do modelo utilizado na previsão. O algoritmo empregado na seleção vai avaliar quais as características devem ser descartadas bem como as que devem ser selecionadas e deve analisar se existe relação de dependência entre elas. É desejável que essas tarefas sejam realizadas com o menor número de operações possível. Cada série temporal tem um comportamento particular, por isso deve-se analisá-la bem de forma a garantir uma seleção de características mais correta possível. Algoritmos de seleção que utilizam os modelos de previsão para analisar a importância de cada característica são denominados de *Wrappers* e a seleção depende do modelo de previsão utilizado, por isso deve-se analisar o comportamento da série para escolher o modelo de previsão utilizado pelo algoritmo de seleção. Sendo assim, pode-se concluir que é importante investigar novas abordagens de seleção de características. Nesta dissertação foi investigado o desempenho de um algoritmo de otimização baseado em enxame de partículas denominado de *Frankenstein's Particle Swarm Optimization*(FPSO) [21].

A proposta do FPSO é ser melhor que os outros algoritmos da sua família [21]. Resultados publicados já concluíram que seu desempenho é superior aos principais algoritmos baseados em enxames de partículas [21, 19]. A utilização deste algoritmo para a seleção de *lags* é pioneira.

Sabe-se que a definição dos parâmetros dos modelos de previsão é muito importante, mas o foco desse trabalho é selecionar as características e o conjunto de treinamento para melhorar as previsões das séries investigadas. Por isso, para cada parâmetro foi

definido um conjunto de valores possíveis. A partir disso foram realizados experimentos com esses valores e definidos os que produziram melhores resultados para o conjunto de treinamento.

O conjunto de treinamento influencia muito no desempenho do modelo para alcançar previsões corretas. Não se tem notícia de uma abordagem de seleção do conjunto de treinamento para previsão de séries temporais. Por isso, neste trabalho é proposta uma metodologia simples de seleção do conjunto de treinamento combinada com o *FastDTW*.

### 1.3 ESTRUTURA DA DISSERTAÇÃO

Esta dissertação está organizada em sete capítulos. Neste capítulo foi mostrada uma breve introdução sobre previsão de séries temporais, os objetivos desta pesquisa e a motivação para realização da mesma.

O Capítulo 2 apresenta noções sobre séries temporais e a importância de prever seus valores futuros, menciona alguns modelos estatísticos de previsão e aborda os dois modelos de computação inteligente usados neste trabalho.

O Capítulo 3 trata de um ponto muito importante na previsão de séries temporais, a seleção de características. São mostradas técnicas mais tradicionais para realizar essa tarefa, suas vantagens e desvantagens. Além disso, expõe alguns algoritmos de otimização baseados em enxames de partículas que foram a base para a criação do FPSO e também são mostrados mais detalhes sobre o FPSO.

O Capítulo 4 apresenta detalhes sobre a técnica *FastDTW*, utilizada na seleção dos conjuntos de treinamento. Além disso, é mostrada a diferença entre o DTW e a distância euclidiana para medição de similaridade entre dois conjuntos de dados.

O Capítulo 5 apresenta a arquitetura da metodologia de previsão utilizada. Cada etapa realizada antes de chegar a previsão é detalhada. Além disso, há uma breve descrição das métricas usadas para medir a qualidade das previsões realizadas.

No Capítulo 6 são apresentados detalhes sobre as séries temporais utilizadas. Este capítulo também descreve como cada experimento foi realizado, os parâmetros utilizados nos modelos de previsão e no FPSO. Os resultados obtidos são comparados com os reportados por outros trabalhos. Por fim, são realizados testes de hipótese para verificar a relevância de se fazer a seleção do conjunto de treinamento.

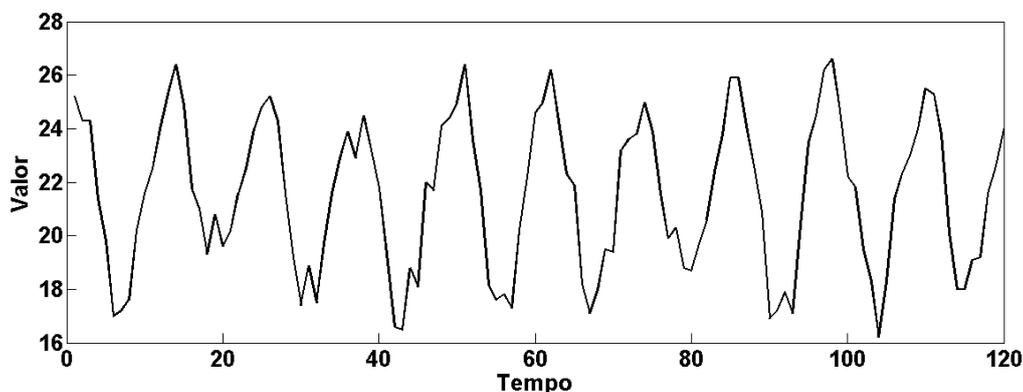
O Capítulo 7 discute as principais contribuições desta dissertação bem como as limitações da metodologia de previsão proposta. Além disso, são listadas possíveis contribuições futuras que podem ser realizadas a partir deste trabalho.

## SÉRIES TEMPORAIS

Este capítulo define o que é uma série temporal, mostra exemplos e define o que é tendência, sazonalidade e estacionariedade. Além disso, expõe dois tratamentos que podem ser realizados nas séries. Outro ponto abordado é a importância de se realizar previsões de séries temporais, citando o que é necessário para sua realização. O capítulo também cita alguns modelos lineares de previsão e disserta um pouco sobre os dois modelos de computação inteligente utilizados nesse trabalho, as redes do tipo *Multilayer Perceptron* (MLP) e as *Support Vector Regression* (SVR).

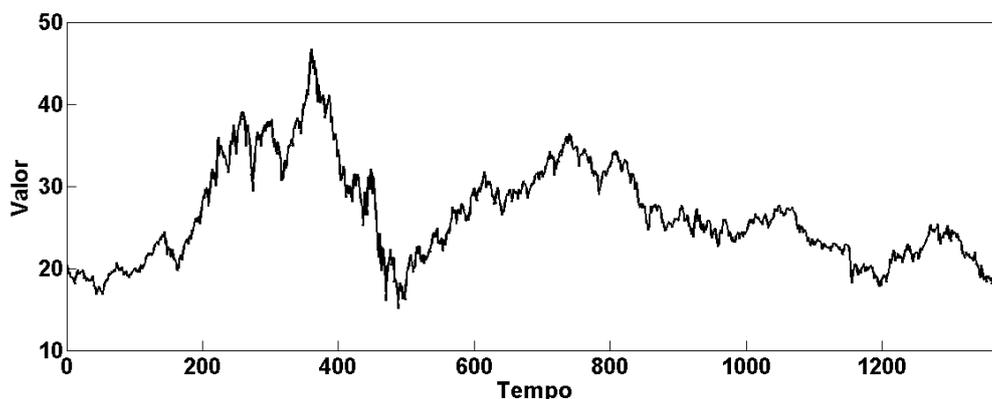
## 2.1 CONCEITOS BÁSICOS

Uma série temporal  $X_t = \{x_t \in \mathbb{R} \mid t = 1, 2, \dots, M\}$  é um conjunto de observações ordenadas no tempo a respeito de um fenômeno qualquer, nesse contexto  $t$  representa o índice temporal e  $M$  é o número de observações que compõem a série. Séries temporais são muito presentes no cotidiano das pessoas. Por exemplo: a temperatura média mensal de uma cidade ao longo de um período, mostrada na Figura 2.1, os níveis de marés ao longo de trinta anos, o índice pluviométrico de uma região durante um ano e cotações de uma ação no mercado financeiro, mostrada na Figura 2.2.



**Figura 2.1** Série de Temperaturas médias mensais da cidade de Cananéia de 1976 a 1985

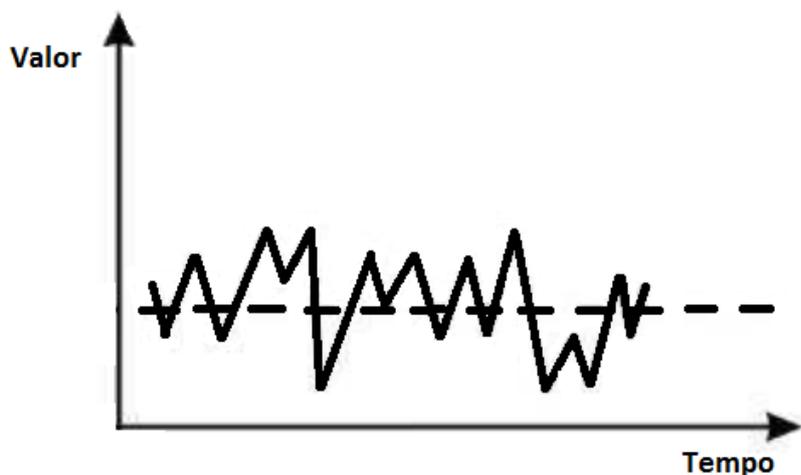
Quanto à classificação, uma série temporal pode ser contínua ou discreta. Em séries contínuas, tais como os níveis de marés, são coletadas amostras em intervalos de tempo iguais para torná-las discretas. Os valores também podem ser coletados em intervalos de tempo diferentes, esse fato depende da disponibilidade dos dados e do foco da pesquisa.



**Figura 2.2** Série de fechamento da ação da Petrobrás de 2007 a 2012

### 2.1.1 Estacionariedade, tendência e sazonalidade

Estacionariedade, tendência e sazonalidade são propriedades que podem existir numa série temporal. Uma série temporal pode ter estacionariedade fraca ou ser estritamente estacionária [1, 27]. Neste trabalho, quando for dito que uma série é estacionária deseje-se referir a uma série com estacionariedade fraca. A estacionariedade é fraca quando os valores da série se desenvolvem aleatoriamente em torno de um valor médio constante [27] como pode ser visto na Figura 2.3.



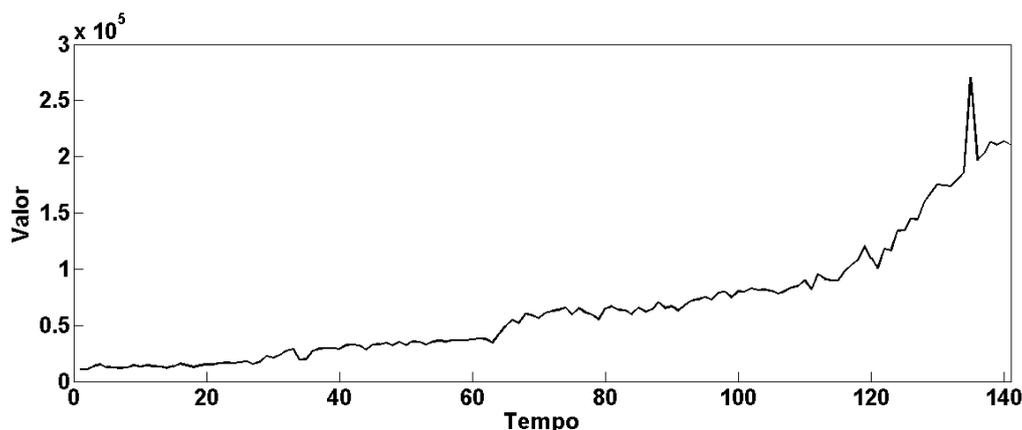
**Figura 2.3** Comportamento de uma série estacionária

Com mais formalidade, uma série é dita fracamente estacionária se sua média e sua autocovariância amostrais independem da amostra selecionada. Para uma série  $X_t$ , teríamos:

$$\begin{cases} \mu_X(t) = E(X_t) \\ \gamma_X(k) = Cov(X_t, X_{t+k}) \end{cases} \quad (2.1)$$

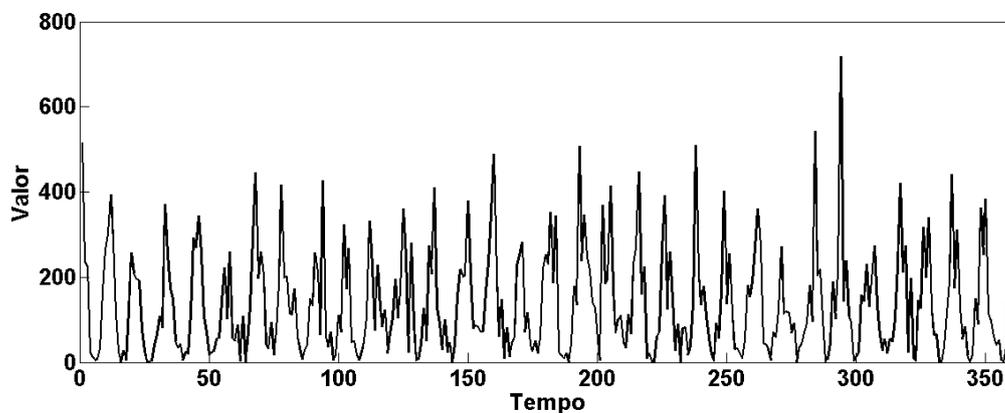
sendo  $\mu_X$  a média amostral e  $\gamma_X(r, s)$  a covariância entre  $X_t$  e  $X_{t+k}$ , sendo  $k \in \mathbb{N}$ , também chamada de autocovariância. A maioria das séries temporais provenientes de situações do cotidiano, tais como séries financeiras, não são estacionárias [28].

A tendência é o comportamento que faz a série ter valores crescentes ou decrescentes em determinados períodos de tempo. Um caso comum de tendência é quando os valores da série crescem em torno de uma reta com inclinação positiva ou negativa, sendo conhecida como tendência linear. Por exemplo, o consumo mensal de energia no estado do Espírito Santo é uma série que apresenta tendência de crescimento, como pode ser verificado na Figura 2.4.



**Figura 2.4** Consumo mensal de energia no Espírito Santo de 1968 a 1979

A sazonalidade é outra característica comum em séries temporais, tal propriedade se faz presente quando existem comportamentos crescente e decrescente em intervalos de tempo regulares ou próximos disso. Por exemplo, é intuitivo supor que uma série de índices pluviométricos semanais de uma cidade apresenta comportamento sazonal. Na Figura 2.5, esse comportamento sazonal pode ser visto através dos índices pluviométricos semanais na cidade de Lavras em Minas Gerais.



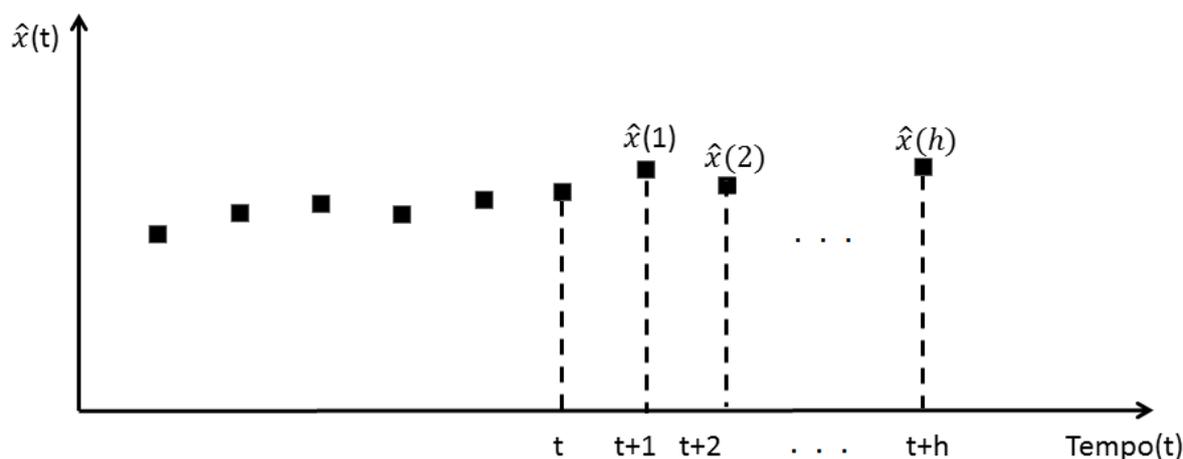
**Figura 2.5** Índice pluviométrico semanal da cidade de Lavras em Minas Gerais de 1966 a 1997

O ciclo é outra característica que pode ser encontrada em uma série temporal, trata-se do comportamento crescente ou decrescente em períodos de tempo não regulares. Variações que não podem ser classificadas como estacionariedade, tendência, sazonalidade ou ciclo são consideradas como ruídos aleatórios e não podem ser previstos por modelos de previsão. Pré-processamentos podem ser realizados na série temporal com o objetivo de reduzir esses ruídos e, como consequência, melhorar as previsões.

## 2.2 PREVISÃO DE SÉRIES TEMPORAIS

A previsão de séries temporais é uma atividade que visa estimar seus valores futuros com a maior precisão possível. Essa tarefa pode auxiliar na tomada de decisões que envolvam esses valores. Por exemplo, prever a quantidade de etanol que será consumida por mês no prazo de um ano, serve para planejar de forma mais adequada a produção desse produto, visando o atendimento da demanda. Através desse exemplo, pode-se perceber que a previsão de séries temporais é uma área de pesquisa importante.

A previsão de valores futuros é baseada em dados históricos da série. Porém, além dos dados, é necessário um modelo de previsão que os utilize e estime os valores futuros. Existem diversos modelos para previsão de séries temporais, por exemplo: os estatísticos [1, 2], os de computação inteligente [4, 3], os híbridos que utilizam modelos estatísticos [29] e os híbridos que utilizam modelos de computação inteligente [19, 26, 8]. A tarefa de escolher um modelo para previsão de séries temporais não é simples, depende muito do comportamento da série e da necessidade do usuário que precisa fazer a previsão. Por isso, existem trabalhos que propõem técnicas de seleção de modelos para previsão. Essas técnicas variam desde as mais simples, como criar um *ranking*, até as mais elaboradas como a utilização de meta-aprendizado [13].



**Figura 2.6** Observações de uma série temporal com previsões de origem  $t$  e horizontes de previsão iguais a um, dois e  $h$ .

A Figura 2.6 mostra um exemplo de série temporal. Pode-se observar que existem  $t$  observações e as previsões de valores futuros para esta série são representadas por

$\hat{x}(h)$  sendo  $h$  denominado de horizonte de previsão. Sendo assim,  $\hat{x}(1)$ ,  $\hat{x}(2)$  e  $\hat{x}(h)$  são previsões com horizontes de previsão 1, 2 e  $h$ , respectivamente. O foco deste trabalho são previsões com horizontes de previsão iguais a um, também chamadas de previsão de um passo. Além do horizonte de previsão, um fator muito importante é o número de elementos históricos anteriores ao horizonte de previsão. Tais elementos formam a janela de previsão e são utilizados pelos modelos. Nas próximas subseções serão expostos alguns conceitos sobre os principais modelos lineares utilizados na área e alguns conceitos sobre redes MLP e Máquina de Vetor de Suporte.

## 2.3 MODELOS PARA PREVISÃO DE SÉRIES TEMPORAIS

Existem inúmeros modelos para previsão de séries temporais, tais modelos podem ser classificados em modelos lineares e modelos não lineares. Dentre os lineares merecem destaque o AR(*Autoregressive*), o MA(*Moving Average*), ARMA(*Autoregressive and Moving Average*) e o ARIMA(*Autoregressive integrated moving average*) [2]. Os quais trouxeram e ainda trazem muitas contribuições para a área de previsão, sendo utilizados em muitos trabalhos e produzindo bons resultados. Como esses modelos não foram utilizados nesta dissertação uma explicação mais detalhada a respeito dos seus funcionamentos foge do escopo deste trabalho. Nesta dissertação foram utilizadas a rede neural do tipo *Multilayer Perceptron* (MLP) e a máquina de vetores de suporte para regressão (SVR). As próximas subseções trazem mais detalhes a respeito dessas duas técnicas.

### 2.3.1 Redes Neurais Artificiais

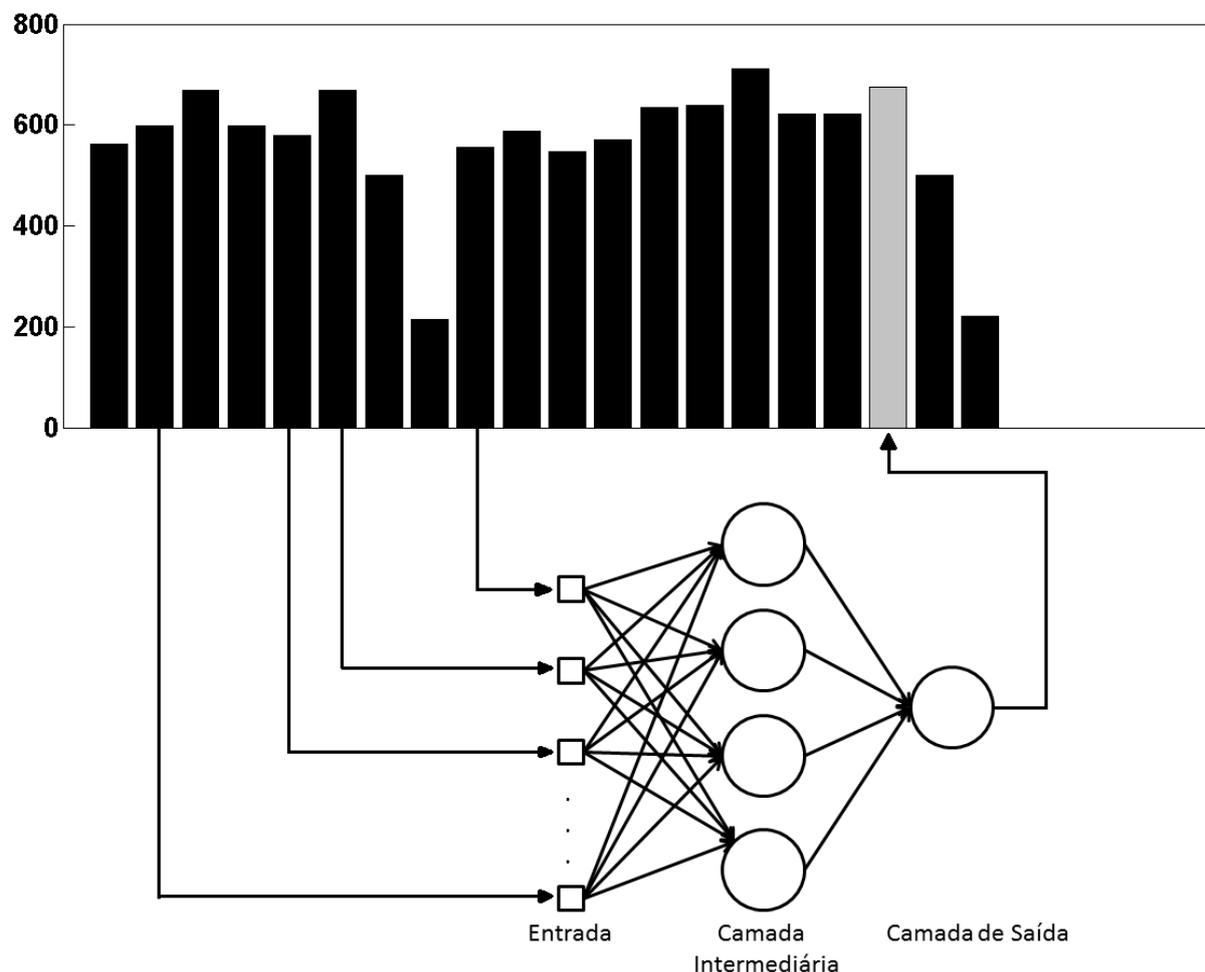
Uma rede neural artificial (RNA) é uma técnica de aprendizagem de máquina baseada no funcionamento do cérebro humano [5, 3]. É formada por unidades de processamento, tais unidades são distribuídas em uma ou mais camadas e interligadas por conexões [5]. Essas unidades são denominadas de neurônios ou nodos da rede. O seu funcionamento se assemelha ao do cérebro na medida em que podem adquirir conhecimento por um processo de aprendizagem [3] e que armazenam o conhecimento nas conexões entre os neurônios [3].

A opção de usar uma RNA para previsão se deve à capacidade de aprender a partir de um conjunto limitado de exemplos e, com esse aprendizado, generalizar as informações aprendidas [5]. A generalização de uma rede é a sua capacidade de apresentar respostas corretas para entradas que não pertenceram ao conjunto de treinamento [3]. Outra capacidade que ela possui é conseguir lidar com dados ruidosos [3]. Além disso, são capazes de aproximar qualquer função [3, 5], por isso são muito usadas para a previsão de séries temporais.

A RNA utilizada neste trabalho foi a *Multilayer Perceptron* (MLP). Essas redes são aplicadas com sucesso nos mais diversos problemas de classificação e de regressão [3]. São do tipo *feedforward*, ou seja, propagam seus sinais para a frente. A sua arquitetura geralmente é composta por uma camada de entrada, uma ou mais camadas intermediárias e uma camada de saída, a Figura 2.7 mostra a arquitetura de rede neural utilizada neste trabalho. Na qual os valores passados da série temporal são as entrada da rede e são

utilizados para o treinamento da mesma para a previsão de valores futuros da série.

Uma MLP possui um conjunto de parâmetros que influenciam seu desempenho e seus valores devem ser cuidadosamente definidos. A taxa de aprendizado, o número de neurônios na camada intermediária, o número de épocas de treinamento e o termo de momento são os principais parâmetros de uma MLP [5]. Dentre as maneiras de escolher os seus valores estão a busca exaustiva, a busca através de um conjunto de experimentos e por algoritmos de otimização. Neste trabalho, foram usadas MLP com uma camada de entrada, uma camada intermediária e uma camada de saída. As quais possuem neurônios que são interligados entre si, as conexões entre eles armazenam o conhecimento da rede. Os valores dessas conexões são denominados de pesos e são calculados através de um algoritmo de treinamento. Neste trabalho foi utilizado o *back-propagation* [5, 3] como algoritmo e os valores dos parâmetros definidos através de experimentos.



**Figura 2.7** Rede MLP tendo como entrada valores passados da série e a saída é um valor futuro. Arquitetura composta por uma camada de entrada, uma camada intermediária e uma camada de saída

O algoritmo *back-propagation* é executado em duas fases. A fase *forward*, que fornece

um padrão à entrada da rede. Em séries temporais, um padrão é composto dos *lags* selecionados e o valor desejado, conforme mostra a Figura 2.7. Os valores fornecidos pelo padrão são propagados até a saída da rede com o objetivo de encontrar o valor desejado. A saída da rede é comparada com esse valor e mede-se o erro. Na fase *backward* os erros são utilizados para ajustar os pesos de todas conexões até que um erro mínimo seja atingido.

### 2.3.2 Máquinas de Vetor de Suporte para Regressão

Uma Máquina de Vetor de Suporte ou *Support Vector Machine* (SVM) é uma técnica fundamentada na Teoria de Aprendizagem Estatística com o objetivo de resolver problemas de classificação [30]. Uma *Support Vector Regression* (SVR) é tipo de máquina de vetor de suporte utilizada para resolver problemas de regressão. São modelos eficientes para trabalhar com dados de alta dimensionalidade que se preocupam em minimizar tanto o erro para o conjunto de treinamento, denominado de risco empírico quanto o erro no conjunto de teste [4].

Em um problema cujo objetivo é aproximar um conjunto de dados  $D = \{(x^1, y^1), \dots, (x^l, y^l)\}$ ,  $x \in \mathfrak{R}^n$ ,  $y \in \mathfrak{R}$  através de uma função  $f(x) = \langle w, \phi(x) \rangle + b$ , sendo  $\phi(x)$  uma função não linear que mapeia as entradas para um espaço de maior de dimensão com o objetivo de torná-los linearmente separáveis [3]. Na SVR o vetor de pesos  $w$  e o coeficiente  $b$  são escolhidos para minimizar a função mostrada na Equação (2.2).

$$\phi_{w,\xi} = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l (\xi_i^- + \xi_i^+) \quad (2.2)$$

Na Equação (2.2),  $\xi_i^-$  e  $\xi_i^+$  são variáveis de folga e medem os custos dos erros de previsão para menos e para mais. O parâmetro  $C$  mede a complexidade do modelo e quando maior o seu valor, mais complexa será a função que fará a aproximação dos pontos. Por outro lado, quando menor o valor de  $C$ , mais suave é a função. Uma SVR possui uma função de *kernel* que é responsável por mapear os pontos de entrada para um espaço de dimensão maior. Nos experimentos realizados, foram investigadas duas implementações de SVR, a de *kernel* linear  $K(x_i, x_j) = \langle 1 + (x_i), (x_j) \rangle^p$  com  $p = 1$  e a de *kernel* com uma função de base Radial (*Radial Basis Function* - RBF)  $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$ , sendo  $\gamma > 0$ . Os parâmetros da SVR foram definidos por meio de experimentos. A escolha dessa técnica teve como base a habilidade dela poder generalizar a partir de um conjunto de treinamento limitado e os bons resultados já reportados na literatura [25].

## 2.4 TRATAMENTOS DOS DADOS

Nesta seção, são apresentadas as duas formas de tratamento dos dados que foram aplicadas nas séries temporais conforme a necessidade.

### 2.4.1 Transformação de Box-Cox

É comum em séries temporais a existência de tendências, nessa situação uma transformação logarítmica pode ser adequada ou, de uma forma geral, a transformação de Box-Cox [31, 32] que pode ser vista na Equação 2.3. Para verificar se essa transformação é necessária, pode-se fazer um procedimento simples que é a divisão da série em janelas e calcula-se a média  $\bar{x}$  e o desvio padrão  $s$  para cada uma delas. Se existir proporcionalidade entre esses valores, então a transformação se faz necessária [27]. Essa transformação tem como objetivo a eliminação da tendência da série temporal, pois esse procedimento pode melhorar os resultados das previsões com redes neurais [28].

$$Z_t^{(\lambda)} = \begin{cases} \frac{Z_t - c}{\lambda}, & \text{se } \lambda \neq 0, \\ \log(Z_t), & \text{se } \lambda = 0, \end{cases} \quad (2.3)$$

Uma forma simples de verificar a relação de proporcionalidade é através do gráfico média  $\times$  desvio. O gráfico elaborado é formado por pontos com coordenadas  $(\bar{x}, s)$ . Sendo  $\bar{x}$  o valor médio de uma janela e  $s$  o seu desvio padrão. Sendo  $Z_{t_1}, \dots, Z_{t_i}, \dots, Z_{t_m}$  as janelas de uma série  $Z_t$ , cada uma com  $k$  observações, têm-se a média na Equação 2.4 e o desvio na Equação 2.5, para cada  $Z_{t_i}$ . Sendo  $x_t \in Z_{t_i}$  os elementos da janela  $Z_{t_i}$ .

$$\bar{x}_i = \frac{1}{k} \sum_{j=1}^k x_t \quad (2.4)$$

$$s_i = \sqrt{\frac{\sum_{j=1}^k (x_t - x_{t-1})^2}{k - 1}} \quad (2.5)$$

Se  $s$  for diretamente proporcional a  $\bar{x}$  então é necessária a transformação de Box-Cox. Neste trabalho, o valor  $\lambda = 0$  foi adotado e definido através da análise dos gráficos média  $\times$  desvio [2] de cada série.

### 2.4.2 Diferenciação da série

Ao fazer a previsão de uma série temporal, um fator que tem impacto sobre os resultados é a estacionariedade [28]. Muitos trabalhos que utilizam redes neurais não levam em conta que a série pode não ser estacionária. Porém, já se sabe que se a série for estacionária o desempenho dos modelos de previsão tendem a ser melhores [28]. Sendo assim, a ação de levá-la a este estado pode ajudar na obtenção de previsões mais corretas. É possível tornar uma série estacionária através de diferenciações sucessivas na mesma [28, 27]. Neste trabalho, todas as séries passaram por apenas uma diferenciação para chegar ao estado de estacionariedade. Apesar de ser um procedimento simples, muitos trabalhos que utilizam redes neurais para previsão não fazem esse processamento [28]. A diferenciação de uma série temporal  $X_t = \{x_t \in \Re | t = 1, 2, 3, \dots, N\}$  é realizada como mostra a Equação 2.6.

$$D_j = x_j - x_{j-1} \quad (2.6)$$

sendo  $D_j$  o valor da série diferenciada. Na Figura 2.8 pode ser vista a diferença de comportamento de uma série temporal antes e depois da diferenciação. Antes da diferenciação, a série temporal tem valor médio de aproximadamente 30, após a diferenciação o valor médio da série se aproxima de 0.

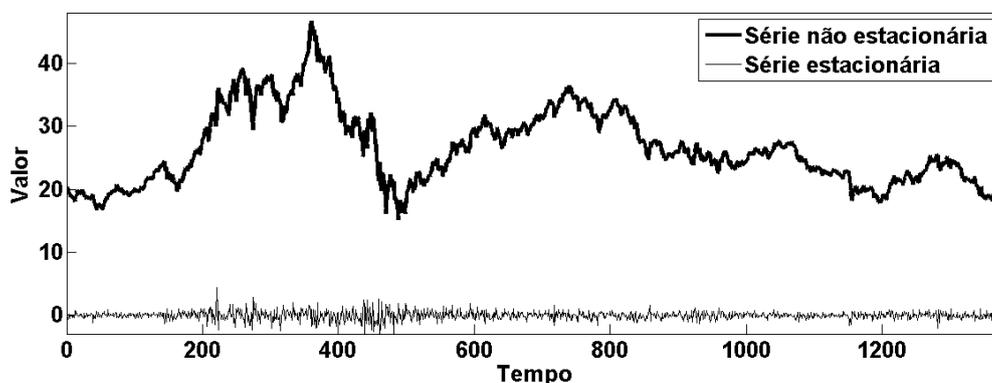


Figura 2.8 Série Não estacionária x Série Estacionária

Geralmente, após uma ou mais diferenciações, a tendência da série é eliminada e seus valores passam a oscilar em torno de 0. Quando o modelo é treinado com uma série diferenciada, os valores da previsão precisam passar pelo processo inverso à diferenciação. Para isso, utiliza-se a Equação 2.7. A estacionariedade da série mostrada na Figura 2.8 foi comprovada através do teste estatístico KPSS (*Kwiatkowski-Phillips-Schmidt-Shin*).

$$x_j = D_j + x_{j-1} \quad (2.7)$$

## 2.5 COMENTÁRIOS FINAIS

Neste capítulo foram apresentados conceitos sobre séries temporais e sobre previsão. Foram citados os principais modelos lineares de previsão e foi explicado o funcionamento e os conceitos básicos sobre as redes MLP e sobre a SVR. Também foram abordadas duas técnicas para pré-processamento de séries temporais, a utilização dessas técnicas pode ajudar a melhorar as previsões realizadas.

## CAPÍTULO 3

# SELEÇÃO DE CARACTERÍSTICAS PARA SÉRIES TEMPORAIS

Neste capítulo serão vistas algumas técnicas empregadas para seleção de características em séries temporais juntamente com suas vantagens e desvantagens. O capítulo também disserta sobre alguns algoritmos de otimização baseados em enxames de partículas e termina falando sobre o FPSO.

### 3.1 O PROBLEMA

No aprendizado supervisionado, um algoritmo de treinamento realiza sua tarefa tendo como base um conjunto de padrões, no qual, cada elemento é descrito por um vetor de características e um valor desejado. No contexto de séries temporais as características também são conhecidas como *lags*. O valor desejado é o alvo da previsão. Por exemplo, para uma série  $A = \{2, 3, 4, 6, 4, 7, 8, 4, 4, 5\}$ , deseja-se realizar a previsão de seus valores com base nos três últimos *lags*. Com isso, o conjunto de padrões adquire o seguinte formato, sendo cada linha um padrão com seu valor desejado em destaque:

2,3,4,**6**  
3,4,6,**4**  
4,6,4,**7**  
6,4,7,**8**  
4,7,8,**4**  
7,8,4,**4**  
8,4,4,**5**

A tarefa de um modelo de previsão é aproximar uma função que receba os *lags* e retorne um valor próximo do desejado. Os *lags* informados na entrada da função são determinantes para o seu valor de saída, por isso, esta tarefa é muito importante no processo de previsão. A seleção tem como objetivo a escolha de um conjunto moderado de *lags* tendo como base algum critério de seleção [33]. Dentre os benefícios que podem ser trazidos por uma seleção correta, podemos destacar: facilita a compreensão do processo gerador dos dados; os modelos de aprendizado são mais simples e mais compreensíveis; há ganhos em eficiência e eficácia; e são utilizados conjuntos de treinamento menores [16, 33].

Os principais desafios do processo de seleção de características são: o tamanho do espaço de busca, a eliminação de características irrelevantes ou redundantes [33, 16] e a manutenção de características interagentes [33]. Sabe-se que quanto maior o espaço de busca, maior o número de subconjuntos de características possíveis e mais difícil é a tarefa de encontrar um subconjunto ótimo. Em um espaço de características grande é muito

provável que existam duas ou mais características que forneçam a mesma informação para o modelo de previsão e outras que não forneçam qualquer informação ou atrapalhem a previsão. Também existem as características interagentes, que só contribuem para a previsão quando estão acompanhadas de outras. A remoção de características desse tipo causa perda de informação útil para a previsão.

A forma mais básica de fazer seleção de características é testar todos os subconjuntos possíveis obtidos a partir de  $n$  características, essa forma é conhecida como busca exaustiva. Sabe-se que existem  $2^n - 1$  possíveis subconjuntos, por isso essa abordagem só é válida quando  $n$  é pequeno. Todavia,  $n$  é grande na maioria dos problemas, tornando-se inviável a utilização da busca exaustiva, tal situação acontece na seleção de características para previsão de séries temporais.

## 3.2 TÉCNICAS PARA SELEÇÃO DE CARACTERÍSTICAS EM SÉRIES TEMPORAIS

Basicamente, as técnicas de seleção de características em séries temporais podem ser classificadas em filtros ou *wrappers*. Nas próximas subseções mais detalhes são apresentados a respeito dessas técnicas.

### 3.2.1 Filtros

Os filtros são técnicas que ordenam as características de acordo com um grau de relevância e retorna as características mais bem ordenadas. O descarte das características consideradas irrelevantes acontece antes do processo de aprendizado do modelo de previsão, por isso os filtros são independentes do modelo que será usado. São exemplos de métricas para definir a relevância de uma característica: a correlação do *lag* com o valor desejado e a informação mútua entre eles. A correlação é muito utilizada para classificar as melhores características e selecioná-las. Tal medida é usada pelos modelos AR, MA e ARIMA [1, 2].

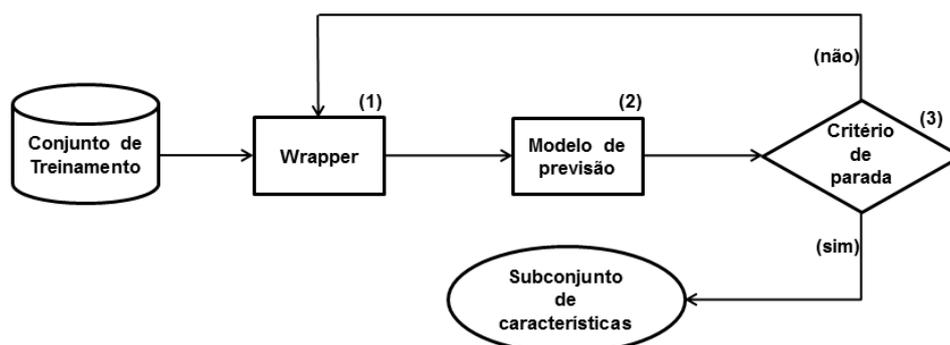
Um procedimento comum ao fazer previsão com métodos estatísticos é fazer análise gráfica da função de autocorrelação parcial para identificar os *lags* estatisticamente significantes. A função de autocorrelação é uma técnica estatística simples e fácil de usar, por isso é muito utilizada para identificar os *lags* mais relevantes para previsão de uma série temporal. Apesar disso, esse procedimento só consegue medir o grau de dependências lineares e não consegue identificar relações não lineares [27]. Apesar de ser muito comum analisar os *lags* mais importantes através do gráfico da função de autocorrelação parcial, esse procedimento pode ser feito de forma automática.

A principal vantagem dos filtros é que consomem poucos recursos computacionais. Porém nem sempre é fácil definir quantos atributos descartar ou escolher, geralmente esse processo é feito por tentativa e erro. Além disso, tais técnicas não levam em consideração o modelo de previsão. Por isso, pode acontecer de a previsão ser melhor com um conjunto de *lags* diferente do selecionado pelo filtro, dependendo da série e do modelo usado. Outro fator desvatajoso dos filtros é que eles assumem que os dados são gerados por um processo linear, por isso não identificam interdependências não lineares [7], sendo assim, são mais

indicados para utilização com modelos de previsão lineares.

### 3.2.2 Wrappers

Uma técnica *wrapper* caracteriza-se por atuar em parceria com um modelo de previsão para selecionar as características [17]. Basicamente são necessários três elementos para realizar seleção de características por essa abordagem: a técnica *wrapper*, um modelo de previsão e um critério de parada. A seleção ou descarte depende da resposta do modelo, como mostra a Figura 3.1.



**Figura 3.1** Funcionamento de um wrapper

O *wrapper* recebe o conjunto de treinamento composto por  $C$  características. O número  $C$  depende do algoritmo. Se é um algoritmo que vai eliminando características conforme a piora do desempenho das previsões ou se é um algoritmo que vai adicionando características de acordo com a melhora.

A busca para frente (*forward selection*) e a busca para trás (*backward elimination*) são os exemplos mais comuns de *wrapper* [17]. A busca para frente inicia com um conjunto formado por um *lag*. Adiciona-se mais um *lag* ao conjunto, avalia-se o desempenho do modelo. Se esse melhorar em relação ao desempenho obtido com o conjunto de *lags* anterior, então o *lag* adicionado é selecionado, caso contrário ocorre seu descarte [17].

A busca para trás inicia com o conjunto completo  $C$  composto por  $n$  *lags* e vai removendo um de cada vez, se a remoção de uma característica  $c_i \in C$  provoca perda de desempenho do modelo, então  $c_i$  é recolocada em  $C$ . Caso contrário,  $c_i$  é descartada.

As duas abordagens possuem a limitação de não considerar a presença de *lags* interagentes, pois cada um é avaliado individualmente. Outra limitação é o alto custo computacional quando o número de características é grande. Tais limitações são comuns a todas as abordagens do tipo *wrapper*. Para um conjunto  $C$  com 10 *lags*, sabe-se que existem  $2^{10} - 1 = 1023$  subconjuntos possíveis. Ou seja, o número de subconjuntos aumenta exponencialmente em relação ao número de *lags*. Sendo assim, a seleção de *lags* via *wrapper* apresenta um problema de escalabilidade, pois, para cada subconjunto  $C' \in C$  investigado, uma execução do modelo de previsão é realizada.

### 3.3 FRANKENSTEIN'S PSO

Neste trabalho, o algoritmo *Frankenstein's PSO* foi utilizado para seleção de características. Já foi mostrado que esse algoritmo apresenta desempenho superior em comparação com outros algoritmos na tarefa de otimizar funções complexas [21]. Além disso, o FPSO já demonstrou melhor desempenho para seleção de características em séries temporais, o que rendeu a publicação de um artigo no decorrer da pesquisa que resultou nesta dissertação [19].

A sigla PSO significa *Particle Swarm Optimization* e trata-se de uma técnica de otimização global baseada em um conjunto de soluções que se inspira no comportamento social de revoada de pássaros [34, 35].

O PSO é formado por um enxame de partículas  $P = \{p_1, p_2, \dots, p_n\}$  que estão distribuídas em um espaço com  $d$  dimensões, esse é o espaço de soluções possíveis para o problema. Em cada instante da execução do PSO, cada partícula armazena sua posição no espaço e sua velocidade de deslocamento pelo espaço. A posição da partícula é avaliada através de uma função  $f : \mathbb{R}^d \rightarrow \mathbb{R}$  que faz o mapeamento de um ponto pertencente ao espaço com  $d$  dimensões para um ponto em um espaço unidimensional. Esse ponto é justamente a qualidade da posição da partícula no  $\mathbb{R}^d$ , ou aptidão.

Ao executar o PSO, o enxame é distribuído no espaço de busca, essa distribuição pode ser aleatória ou de acordo com algum critério. A cada iteração  $t$  do algoritmo, uma partícula  $i$  armazena dois vetores  $x_i^t$  e  $v_i^t$  que informam sua posição e sua velocidade no espaço respectivamente. Na mesma iteração, uma partícula armazena a sua melhor posição visitada  $pb_i^t$  de *personal best* e a melhor posição encontrada pela sua vizinhança  $lb_i^t$  de *local best*. A cada iteração, as partículas trocam informações com suas vizinhas fazendo com que suas posições e velocidades sejam atualizadas conforme as Equações (3.1) e (3.2). Portanto, uma partícula se movimenta sob influência de sua melhor posição e da melhor posição de sua vizinhança.

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (3.1)$$

$$v_i^{t+1} = v_i^t + \varphi_1 U_1^t (pb_i^t - x_i^t) + \varphi_2 U_2^t (lb_i^t - x_i^t) \quad (3.2)$$

Sendo  $v_i^{t+1}$  e  $x_i^{t+1}$  a velocidade e a posição da partícula  $i$  no instante  $t + 1$ . Na Equação (3.2), os termos  $\varphi_1$  e  $\varphi_2$  são dois parâmetros denominados de coeficientes de aceleração. Esses valores representam a influência da melhor posição da partícula e da melhor posição da vizinhança sobre sua próxima velocidade. Os parâmetros  $U_1^t$  e  $U_2^t$  são aleatórios e ficam entre zero e um. Com o deslocamento das partículas no espaço de busca espera-se alcançar um ponto de mínimo ou de máximo global. Esse valor é encontrado através da avaliação da posição, realizada pela função  $f$ . Desde a sua primeira implementação, na década de 90 [35], foram propostas algumas variações deste algoritmo [36, 37, 38, 39]. De todas as variações propostas, quatro contribuíram diretamente para a composição do FPSO [21].

A primeira variação que merece destaque é o *Constricted Particle Swarm Optimizer* [37]. Nessa abordagem, foi adicionado um termo na Equação (3.2) responsável pela atualização

da velocidade, este termo não permite crescimento ilimitado da velocidade, pois valores elevados causam instabilidade no algoritmo, fato que dificulta a convergência do mesmo. Sendo assim, a adição do fator  $\chi$  tem como meta facilitar a convergência. A Equação (3.3) representa a atualização da velocidade após a adição do termo  $\chi$ .

$$v_i^{t+1} = \chi(v_i^t + \varphi_1 U_1^t(pb_i^t - x_i^t) + \varphi_2 U_2^t(lb_i^t - x_i^t)) \quad (3.3)$$

Na Equação (3.3),  $\chi = \frac{2}{|2-\varphi-\sqrt{\varphi^2-4\varphi}|}$ , sendo  $\varphi = \sum_i \varphi_i$  e  $\varphi > 4$ .

A segunda variação que influenciou o FPSO foi o algoritmo *Time-Varying Inertia Weight Particle Swarm Optimizers* [38, 40]. De acordo com essa abordagem, o primeiro termo da Equação (3.3) comporta-se como um fator de inércia na atualização da velocidade. Devido a isso, foi proposta uma maneira de atualizá-la de forma que ocorre a ponderação da influência do termo de inércia [38]. Essa ponderação pode ser determinada pelo valor de  $w^t$  que no início da execução do algoritmo fornece ao mesmo um comportamento de exploração do espaço de busca. Posteriormente, a busca ocorre nas regiões mais promissoras, comportamento denominado de exploração. A Equação (3.4) mostra como é realizada a atualização da velocidade ao levar em consideração o peso da inércia.

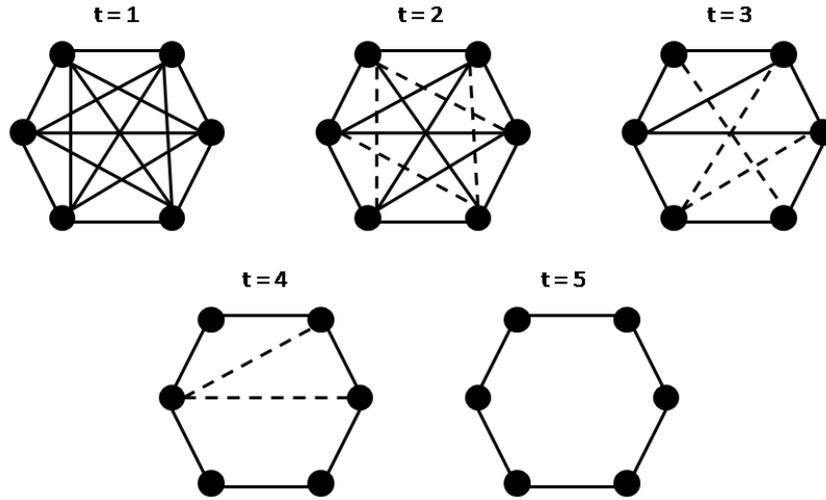
$$v_i^{t+1} = w^t v_i^t + \varphi_1 U_1^t(pb_i^t - x_i^t) + \varphi_2 U_2^t(lb_i^t - x_i^t) \quad (3.4)$$

A terceira proposta de melhoria foi denominada de *Fully Informed Particle Swarm Optimizer* [39]. Tal abordagem assegura que cada partícula receba informações de todas as outras da população, por isso o somatório está presente na Equação (3.5). Para uma partícula  $p_i$ ,  $\varphi$  é decomposto como  $\varphi_k = \frac{\varphi}{N_i} \forall p_k \in N_i$ . Como resultado, a atualização da velocidade é realizada pela Equação (3.5).

$$v_i^{t+1} = \chi \left( v_i^t + \sum_{p_k \in N_i} \varphi_k U_k^t(pb_i^t - x_i^t) \right) \quad (3.5)$$

A quarta proposta, denominada de *Adaptive Hierarchical Particle Swarm Optimizer* (AHP SO) [41], propõe um mecanismo de modificação da vizinhança das partículas ao longo da execução do algoritmo. Essa abordagem usa uma estrutura em forma de árvore para realizar podas na vizinhança. As partículas com as posições melhores avaliadas são os pais, enquanto as partículas com avaliações piores são os filhos. Na medida em que as posições vão se modificando, pais e filhos podem trocar de lugar. Enquanto executa, o algoritmo vai efetuando podas na árvore até que a mesma atinja uma profundidade mínima. Com isso, também é possível realizar a exploração e a exploração no espaço de busca.

A abordagem (AHP SO) serviu de base para o procedimento de poda realizado pelo FPSO. Porém o procedimento de poda realizado no FPSO utiliza uma estrutura que se assemelha a um grafo, conforme pode ser visto na Figura 3.2. Nesta figura há o passo a passo do procedimento de poda, uma aresta que liga duas partículas significa que existe comunicação entre as mesmas. O procedimento utilizado pelo FPSO contém um



**Figura 3.2** Procedimento de poda aplicado numa topologia de partículas

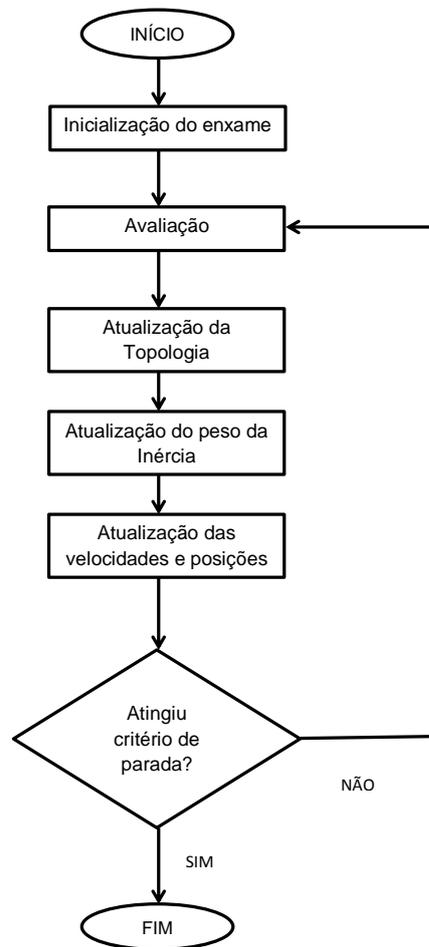
parâmetro  $k$ . Através deste parâmetro é calculado o número de iterações que devem ser realizadas para a execução de podas na vizinhança.

Para um melhor entendimento do processo de poda feito pelo FPSO, pode-se utilizar um exemplo. Supondo um enxame com 6 partículas e definindo-se  $k = 12$ , para cada  $\left\lceil \frac{k}{(n-3)} \right\rceil = \left\lceil \frac{12}{(6-3)} \right\rceil = 4$  iterações, o algoritmo remove algumas arestas. Na situação mostrada na Figura 3.2, existem  $\frac{n \cdot (n-1)}{2} = \frac{6 \cdot 5}{2} = \frac{30}{2} = 15$  arestas e  $\frac{n \cdot (n-3)}{2} = \frac{6 \cdot 3}{2} = \frac{6 \cdot 3}{2} = 9$  devem ser removidas para atingir uma topologia em anel. As podas ocorrem da seguinte maneira: na primeira poda são removidas  $(n-2) = (6-2) = 4$  arestas, na segunda serão removidas  $(n-3) = (6-3) = 3$ , até que são removidas apenas  $(n-4) = (6-4) = 2$  arestas.

Nota-se que o FPSO começa com uma topologia totalmente conectada na qual todas as partículas se comunicam. Isso permite que elas se movam ao longo de todo o espaço de busca. Com o andamento da execução, a conectividade entre as partículas vai diminuindo até alcançar uma topologia em anel. A topologia em anel faz com que uma partícula influencie a posição e a velocidade apenas das partículas mais próximas dela.

O FPSO é composto por partes de cada algoritmo mencionado. Tendo como meta ser o mais eficiente [21]. Seu desempenho foi melhor que os outros algoritmos na tarefa de otimizar diversas funções [21, 41, 39, 38, 40, 37].

O funcionamento do FPSO pode ser resumido a partir da Figura 3.3. O algoritmo inicia o enxame de partículas aleatoriamente e cada uma delas passa por uma avaliação. Com base nesta, tem-se a aptidão de cada partícula e são obtidas as melhores posições locais e globais. Sabendo essas informações, são feitas a atualização da vizinhança, do peso da inércia, da velocidade e posição de cada partícula [21]. Após todas as atualizações, avalia-se o critério de parada, que pode ser composto por um número máximo de execução e um limiar de erro. Caso o critério seja obedecido, a execução para. O Algoritmo 1 mostra o conjunto de etapas realizadas pelo FPSO até que seja obtido um ponto de convergência.



**Figura 3.3** Fluxo de execução do FPSO

```

/*Inicialização da População*/
for i = 1 : n do
    Cria partícula  $p_i$  e adiciona ao enxame  $P$ ;
    Inicializa os vetores  $x_i$  e  $v_i$ ;
     $pb_i = x_i$ ;
     $N_i = P$ ;
end
t = 0;
esteps = 0;
/*Laço principal*/
repeat
    Laço de avaliação;
    for i = 1 : n do
        if  $f(x_i) < f(pb_i)$  then
             $pb_i = x_i$ 
        end
    end
    /*Atualização da Topologia*/
    if  $t > 0$  e  $t \leq k$  e  $t \bmod \left\lceil \frac{k}{(n-3)} \right\rceil = 0$  then
        for i = 1 : n - (2 + esteps) do
            if  $|N_i| > 2$  then
                Selecione uma partícula  $p_r \in N_i$  aleatoriamente tal que  $p_r$  não
                seja adjacente a  $p_i$ ;
                Elimina  $p_r$  de  $N_i$ ;
                Elimina  $p_i$  de  $N_r$ ;
            end
        end
        esteps = esteps + 1;
    end
    /*Atualização do peso da Inércia*/
    if  $t \leq wt_{max}$  then
         $w^t = \frac{wt_{max}-t}{wt_{max}}(w_{max} - w_{min}) + w_{min}$ ;
    end
    else
         $w^t = w_{min}$ ;
    end
    /*Atualização das velocidades e das posições*/
    for i = 1 : n do
        Gera  $U_m^t \forall p_m \in N_i$ ;
         $\varphi_m = \frac{\varphi}{|N_i|} \forall p_m \in N_i$ ;
         $v_i^{t+1} = w^t v_i^t + \sum \varphi_k U_k^t (pb_k^t - x_i^t)_{p_m \in N_i}$ ;
         $x_i^{t+1} = x_i^t + v_i^{t+1}$ ;
    end
    t = t + 1;
    solucao = argmin( $f(pb_i^t)$ );
until  $f(\text{solucao})$  é boa o suficiente ou  $t = t_{max}$ ;

```

Algorithm 1: Algoritmo FPSO

Nesse trabalho, a aptidão de uma partícula é medida pela função  $f$  que foi o MAPE das previsões sobre o conjunto de treinamento. O limiar utilizado foi o  $\text{MAPE} = 0$ , já o número de iterações máximo de iterações foi  $T = S^2$ , sendo  $S$  o número de partículas do enxame. A maneira de determinar o número máximo de iterações utilizada nesse trabalho foi a mesma adotada em trabalhos que já avaliaram o desempenho do FPSO anteriormente [21].

### 3.4 COMENTÁRIOS FINAIS

Neste capítulo foi apresentado o problema de seleção de características para séries temporais. Foram mostrados os dois principais grupos de técnicas de seleção de características utilizados, os filtros e os *wrappers* bem como suas vantagens e desvantagens.

Além disso, foi realizada uma exposição dos principais conceitos sobre algoritmos de otimização baseados em enxames de partículas e foram explicados quatro algoritmos pertencentes a essa família. Por fim, foi discutido o funcionamento do FPSO que é formado por partes dos algoritmos expostos e que foi utilizado nesta dissertação na tarefa de seleção de características.

# SELEÇÃO DO CONJUNTO DE TREINAMENTO

Este capítulo contém detalhes sobre a forma de seleção do conjunto de treinamento. A sua escolha foi baseada na similaridade entre subconjuntos da série alvo da previsão. No processo de seleção, uma variação do algoritmo *Dynamic Time Warping* (DTW) foi utilizada. Por isso, o capítulo expõe alguns conceitos para o entendimento deste algoritmo, faz uma comparação deste com a distância euclidiana. Feita a comparação, serão apresentados detalhes sobre o *FastDTW*.

## 4.1 DYNAMIC TIME WARPING

A técnica usada para fazer a seleção do conjunto de treinamento foi uma variação do DTW [23], o *FastDTW*. Porém, antes de explicar o funcionamento desta última técnica é necessário dissertar um pouco sobre o funcionamento do DTW.

O DTW é uma técnica que serve para analisar similaridade entre sinais, os quais podem ser desde a fala de uma pessoa até uma série temporal [22]. Por se tratar de uma técnica de programação dinâmica [42], é importante dissertar um pouco sobre essa área.

A Programação Dinâmica (PD) [42] é uma metodologia para a construção de algoritmos para solucionar problemas de otimização. Problemas que requerem decisões sequenciais e que se relacionam são o alvo da PD. Nesse contexto, uma decisão tem custo imediato e afeta as decisões futuras.

A PD se baseia no Princípio de Otimalidade de Bellman, o qual afirma que uma política de decisões ótimas só pode ser formada por subpolíticas ótimas. Dessa forma, os objetivos da programação dinâmica são basicamente: como obter a sequência de decisões para resolver determinado problema, minimizar o custo total em um número de estágios finito e ter compromisso entre os custos imediato e futuro.

Nas últimas décadas, pesquisas envolvendo séries temporais ganharam mais espaço na comunidade científica [22]. Em muitos desses trabalhos a tarefa de classificação se faz presente [23]. Nesse sentido, medir a similaridade entre séries temporais é muito importante.

A distância euclidiana é muito utilizada para medir similaridade entre séries temporais, devido a facilidade de implementação e ao seu baixo custo computacional. Apesar disso, ela é sensível a séries que apresentem diferenças de fase, ou seja, pequenas distorções nos dados em relação ao eixo temporal. Essa limitação faz com que tal medida torne-se imprecisa em diversas situações que exijam análise de similaridade. Por isso, são necessárias outras formas de executar essa tarefa.

O DTW é uma técnica de programação dinâmica que surgiu como uma alternativa à utilização da distância euclidiana. A referida técnica é capaz de lidar com diferenças de fase, sendo portanto mais confiável para analisar similaridade entre séries temporais.

A principal limitação do DTW é a sua complexidade quadrática, em outras palavras, se duas séries temporais  $P$  e  $Q$  possuírem 30 observações cada uma, o DTW fará 900 operações, enquanto que a distância euclidiana fará apenas 30 operações. Tal limitação pode se transformar em um grande problema quando se quer medir similaridade entre conjuntos de dados grandes. No entanto, já existem variações do DTW que conseguem ser fiéis na medição com menos esforço computacional [24].

A Equação (4.1) mostra como é calculada a distância euclidiana entre duas séries  $P = \{p_1, p_2, \dots, p_m\}$  e  $Q = \{q_1, q_2, \dots, q_n\}$ .

$$d(P, Q) = \sum_{i=1}^T \sqrt{(p_i - q_i)^2} \quad (4.1)$$

Na Equação (4.1),  $i$  é o índice temporal dos respectivos elementos de  $P$  e  $Q$ . Pode-se notar que para calcular a distância euclidiana é necessário que os dois conjuntos de dados possuam o mesmo número de elementos. Essa condição não é necessária para o cálculo do DTW. Tal procedimento é realizado da seguinte maneira, cria-se uma matriz  $H_{n \times m}$  na qual o elemento  $h_{i,j}$  é a distância euclidiana entre o ponto  $q_i$  e  $p_j$ , ou seja,  $h_{i,j} = d(q_i, p_j) = \sqrt{(q_i - p_j)^2}$ . Após criar a matriz  $H$ , o objetivo do DTW é encontrar um caminho  $W$  com menor somatório de distâncias possível. Cada caminho dentro da matriz é denominado de *Warping Path*. Cada caminho é formado por um conjunto de pontos, conforme pode ser visto na seguinte expressão:

$$W = \{w_1, w_2, \dots, w_k, \dots, w_Z\}$$

sendo  $w_k = h_{k,k}$  e  $\max(m, n) \leq Z \leq m + n - 1$ . Cada caminho dentro da matriz está sujeito a três restrições [23]:

- Limite

Define o ponto de início e final de um *Warping Path*  $w_1 = (1, 1)$  e  $w_k = (m, n)$ , ou seja cada caminho deve começar em  $h_{1,1}$  e terminar em  $h_{m,n}$ .

- Monotonicidade

Força com que os passos de um *warping path* sejam sempre em direção ao ponto  $h_{n,m}$ .

- Continuidade

Limita o tamanho dos passos que podem ser dados entre os elementos da matriz, ou seja, sendo  $w_k = (a, b)$  então  $w_{k-1} = (a', b')$ , com  $a - a' \leq 1$  e  $b - b' \leq 1$ . Essa restrição faz com que os passos sejam dados apenas entre células adjacentes, incluindo células diagonais.

Inúmeros caminhos na matriz  $M$  satisfazem as restrições apresentadas, mas o objetivo do DTW é encontrar o caminho da matriz  $H_{n,m}$  com o menor somatório de distâncias. Para encontrá-lo, a Equação (4.2) é usada.

$$DTW_2(P, Q) = \sqrt{\gamma(i, j)} \quad (4.2)$$

Na Equação (4.2),  $\gamma(i, j)$  representa a distância acumulada e deve ser calculada como mostra a Equação (4.3).

$$\gamma(i, j) = d(q_i, p_j) + \min\{\gamma(i-1, j-1), \gamma(i-1, j), \gamma(i, j-1)\} \quad (4.3)$$

Através de um exemplo é possível perceber a limitação da distância euclidiana para medir a similaridade entre duas séries. Supondo duas séries do mesmo tamanho  $P = \{6, 6, 7, 7, 8, 8, 9, 9, 9, 9, 9, 8, 8, 7, 7, 6, 6\}$  e  $Q = \{6, 6, 6, 6, 7, 7, 8, 8, 9, 8, 7, 7, 6, 6, 6, 6\}$ . Usando a Equação (4.1) é possível obter  $d(P, Q) = \sqrt{11} \approx 3,3$ .

Para os mesmos conjuntos  $P$  e  $Q$ , o DTW monta uma matriz das distâncias entre cada um dos pontos de  $P$  e  $Q$ , conforme a Tabela 4.1. Através da Equação (4.3) encontra-se o caminho destacado na referida tabela.

6	0	0	0	0	1	1	4	4	9	4	1	1	0	0	0	<b>0</b>
6	0	0	0	0	1	1	4	4	9	4	1	1	<b>0</b>	<b>0</b>	<b>0</b>	0
7	1	1	1	1	0	0	1	1	4	1	0	<b>0</b>	1	1	1	1
7	1	1	1	1	0	0	0	0	4	1	<b>0</b>	0	1	1	1	1
8	4	4	4	4	1	1	0	0	1	<b>0</b>	1	1	4	4	4	4
8	4	4	4	4	1	1	0	0	1	<b>0</b>	1	1	4	4	4	4
9	9	9	9	9	4	4	1	1	<b>0</b>	1	4	4	9	9	9	9
9	9	9	9	9	4	4	1	1	<b>0</b>	1	4	4	9	9	9	9
9	9	9	9	9	4	4	1	1	<b>0</b>	1	4	4	9	9	9	9
9	9	9	9	9	4	4	1	1	<b>0</b>	1	4	4	9	9	9	9
8	4	4	4	4	1	4	0	<b>0</b>	1	0	1	1	4	4	4	4
8	4	4	4	4	1	1	<b>0</b>	0	1	0	1	1	4	4	4	4
7	1	1	1	1	0	<b>0</b>	1	1	4	1	0	0	1	1	1	1
7	1	1	1	1	<b>0</b>	1	1	1	1	1	0	0	1	1	1	1
6	0	0	0	<b>0</b>	1	1	4	4	9	4	1	1	0	0	0	0
6	<b>0</b>	<b>0</b>	<b>0</b>	0	0	1	4	4	9	4	1	1	0	0	0	0
	6	6	6	6	7	7	8	8	9	8	7	7	6	6	6	6

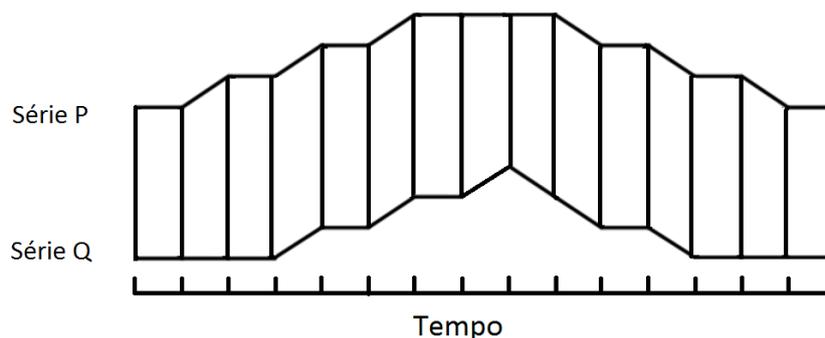
**Tabela 4.1** Tabela  $H_{16,16}$  criada pelo DTW

O caminho destacado na Tabela 4.1 é o que minimiza a distância acumulada, dessa forma o valor retornado pelo DTW é:

$$DTW_2(P, Q) = \sqrt{0} = 0$$

Ou seja, foi encontrado um caminho na matriz com custo zero. Vale salientar que o valor retornado pelo DTW não significa que as séries  $P$  e  $Q$  são iguais.

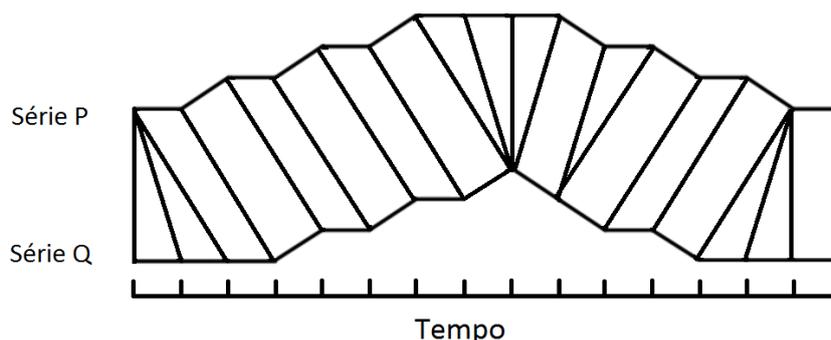
As Figuras 4.1 e 4.2 foram elaboradas através do alinhamento das séries  $P$  e  $Q$ . Na Figura 4.1 observa-se que a distância euclidiana é calculada apenas entre os pontos que



**Figura 4.1** Similaridade através da distância Euclidiana

apresentam mesmo índice temporal. Ou seja, mede-se a distância entre  $p_i \in P$  e  $q_j \in Q$  para  $i = j$ .

Na Figura 4.2 observa-se que o DTW associa pontos de  $P$  e  $Q$  mesmo que os mesmos não possuam o mesmo índice temporal, por isso consegue lidar com diferenças de fase e considera similares séries temporais que aparentemente não são.



**Figura 4.2** Similaridade através do DTW

#### 4.1.1 Fast Dynamic Time Warping

O DTW é um algoritmo que encontra o caminho de menor custo numa matriz, como foi demonstrado na Tabela 4.1. O custo de encontrar exatamente a menor distância é alto. Para duas séries de tamanho  $n$  o DTW percorre uma matriz  $n \times n$  para encontrar o menor caminho dentro da matriz. Uma alternativa à utilização do DTW é o algoritmo *FastDTW*, o qual utiliza uma abordagem que permite encontrar um caminho muito próximo do mínimo sem precisar percorrer todas as células da matriz. Sua complexidade é linear enquanto que o DTW tem complexidade quadrática [24]. Basicamente, três operações chave são realizadas pelo *FastDTW* [24], são elas:

- **Redução:** Nesta etapa o algoritmo reduz a resolução das séries envolvidas no cálculo da similaridade. Ou seja, encontra duas séries com menos pontos que represente, com maior precisão possível, as séries iniciais.

- **Projeção:** Etapa na qual encontra-se o *Warping Path* entre as séries com resoluções reduzidas. Esse caminho na matriz de baixa resolução é projetado numa matriz de resolução mais alta.
- **Refinamento:** Através de ajustes, faz-se o refinamento do caminho que foi projetado na etapa anterior.

O *FastDTW* reduz a resolução de uma série ao tirar a média de duas observações adjacentes. Dessa forma, uma matriz com metade dos pontos é gerada. A etapa de redução é realizada algumas vezes para produzir diferentes resoluções. A etapa de projeção pega um *warping path*  $wp_i$ , calculado numa matriz de baixa resolução, e determina as células vizinhas pelas quais ele pode passar numa matriz de resolução mais alta. O refinamento encontra o caminho ótimo nessas células. O tamanho dessa vizinhança é controlado por um parâmetro denominado de raio.

A Figura 4.3 mostra as etapas executadas pelo algoritmo para encontrar o caminho ótimo do exemplo mostrado na Tabela 4.1. Na referida figura foi utilizado um raio de tamanho 1. Da esquerda para a direita o algoritmo executa os seguintes passos:

1. Calcula-se o *warping path*  $wp_0$  ótimo na matriz de mais baixa resolução  $m_0$
2.  $wp_0$  é projetado numa matriz com o dobro de resolução  $m_1$ . Dentre as células numa vizinhança de raio 1 é encontrado o  $wp_1$ . As células com cores mais escuras foram as visitadas pelo algoritmo. Pela Figura 4.3 apenas 10 células foram visitadas nessa etapa.
3.  $wp_1$  é projetado numa matriz  $m_2$  com o dobro de resolução de  $m_1$ . O  $wp_2$  é encontrado entre as células vizinhas de  $wp_1$ . Conforme pode ser observado, apenas 22 células são visitadas nessa etapa.
4.  $wp_2$  é projetado na matriz  $m_3$  com o dobro de resolução de  $m_2$ . Por fim, calcula-se o  $wp_3$ , na última etapa executada pelo algoritmo. Nessa etapa foram visitadas 50 células.

Vale salientar que, com essas etapas o *FastDTW* encontra um caminho bem próximo do ótimo com uma complexidade linear [24] e que no lugar de percorrer 256 células, como ocorreria no DTW, foram percorridas 84 células.

## 4.2 COMENTÁRIOS FINAIS

Neste capítulo foi mostrado alguns conceitos sobre programação dinâmica, área a qual pertence o algoritmo DTW. Foram mostrados detalhes a seu respeito para analisar a similaridade entre séries temporais e comparado com a distância euclidiana.

Por fim, este capítulo expôs detalhes sobre o *FastDTW*, um algoritmo que é capaz de aproximar o cálculo realizado pelo DTW com menos esforço computacional.

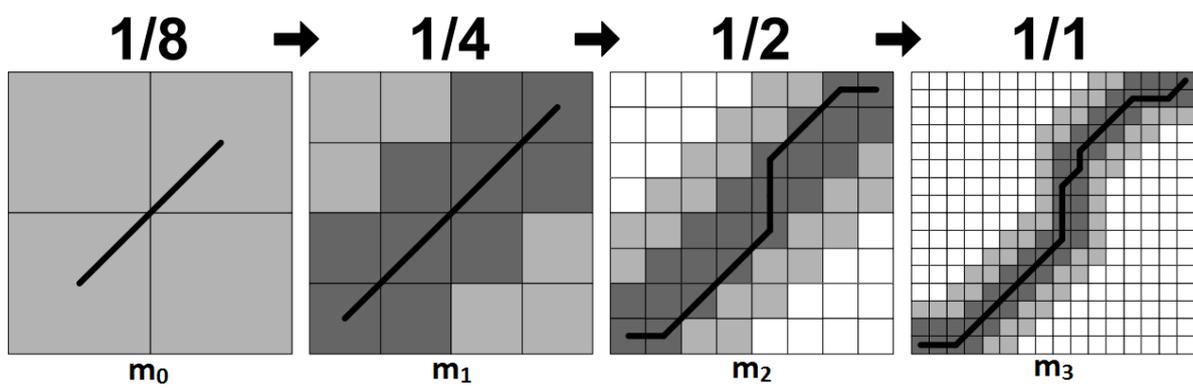


Figura 4.3 Quatro diferentes resoluções avaliadas durante execução do FastDTW

## MÉTODO PROPOSTO

Neste capítulo é apresentada a metodologia de previsão completa, que é realizada em três etapas: tratamento dos dados, seleção do conjunto de treinamento e seleção das características. O principal motivo que levou a implementação dessa metodologia foi conseguir fazer previsões com maior grau de precisão.

## 5.1 ARQUITETURA UTILIZADA

A arquitetura da metodologia proposta, basicamente, faz o tratamento da série temporal, seleciona o conjunto de treinamento e as características. Por fim, realiza previsão do conjunto de teste. A Figura 5.1 resume a arquitetura do sistema. Nas próximas subseções serão explicadas em detalhes cada uma das etapas realizadas para alcançar a previsão.

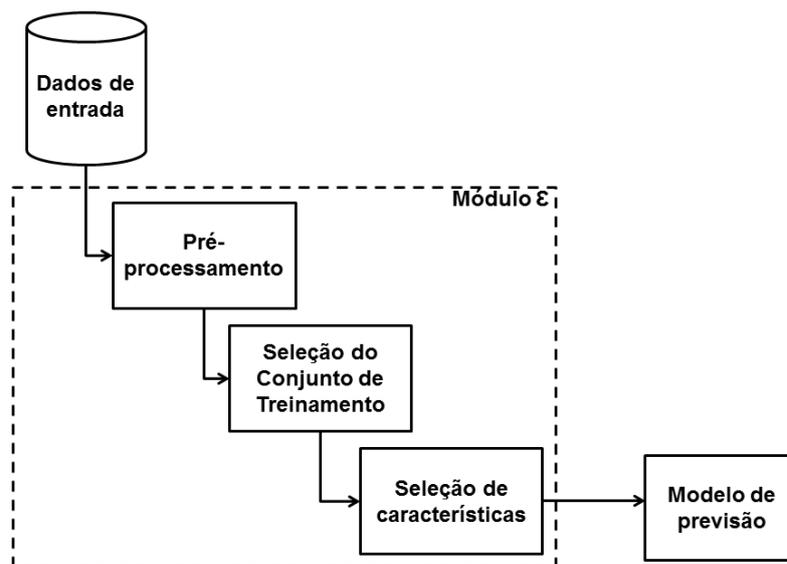
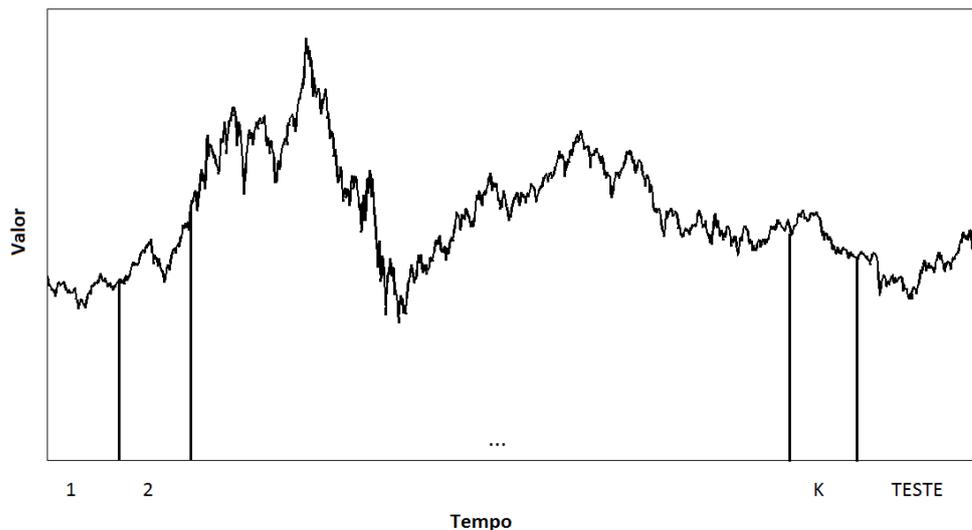


Figura 5.1 Arquitetura da metodologia de previsão

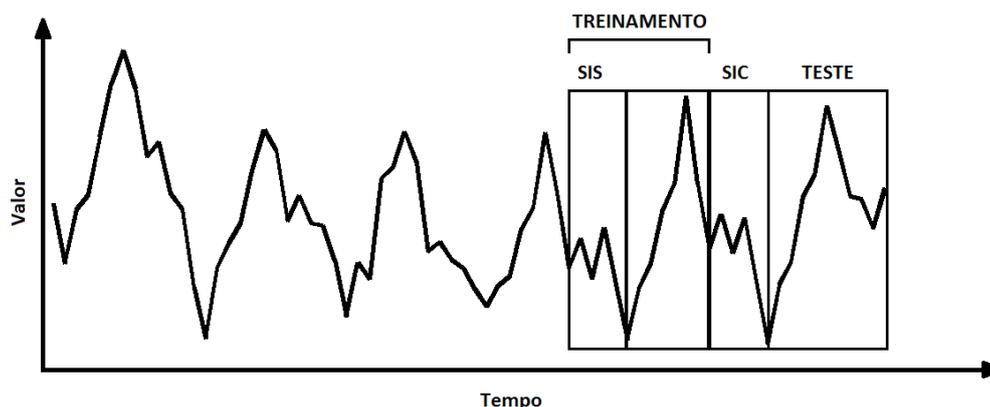
## 5.1.1 Seleção do conjunto de treinamento

A Figura 5.2 mostra a divisão do conjunto de dados empregada para seleção da janela de dados mais similar à janela vizinha ao conjunto de teste. Supondo uma série temporal com  $N$  observações, separa-se o conjunto de teste composto por  $T$  observações. O restante do conjunto é dividido em  $K$  partes de tamanhos iguais, cada parte com  $\frac{N-T}{K}$  observações. O



**Figura 5.2** Divisão empregada em cada série temporal para seleção do conjunto de treinamento

$K$ -ésimo subconjunto é denominado de Subconjunto Indicador de Comportamento (SIC). Depois disso, o procedimento busca dentre os outros  $K - 2$  subconjuntos qual possui maior similaridade em relação ao SIC, esse conjunto foi denominado de Subconjunto Indicador de Similaridade (SIS). A medida de similaridade é calculada pelo *FastDTW*. Após encontrar o SIS, o conjunto de treinamento é composto pela união do SIS com as  $T$  observações imediatamente posteriores a ele, conforme mostra a Figura 5.3. Feito isso, o tamanho do conjunto de treinamento é  $\frac{N-T}{K} + T$ . Caso não seja possível pegar as  $T$  observações posteriores, são pegadas as  $\frac{N-T}{K}$  observações.



**Figura 5.3** Detalhamento sobre a montagem do conjunto de treinamento

Essa abordagem é baseada na idéia que as  $T$  observações imediatamente posteriores ao SIS têm um comportamento similar ao conjunto de teste, por isso tais observações foram adicionadas ao conjunto de treinamento. O número de partes que o conjunto de dados será dividido foi escolhido através de experimentos. Vários experimentos foram realizados

com diferentes tamanhos de janela, sempre visando a escolha do melhor conjunto de treinamento.

### 5.1.2 Tratamento dos dados

A etapa de tratamento dos dados foi realizada de acordo com a necessidade apresentada pelas séries temporais conforme foi mostrado na Seção 2.1.1. Essa necessidade foi identificada através de comportamentos de algumas propriedades, tais como: média amostral, desvio padrão amostral e autocorrelação amostral.

### 5.1.3 Seleção de características

A seleção de características é determinante para o desempenho das previsões. Para realizar essa tarefa foi utilizado o algoritmo FPSO [21]. O espaço de busca do FPSO é definido pelo número máximo características que se deseja utilizar. Com as características selecionadas, o modelo de previsão extrai o conhecimento e realiza a previsão dos valores futuros da série. O número inicial de características foi denominado de características candidatas e sua quantidade é um parâmetro da metodologia de previsão utilizada neste trabalho. Foram investigadas conjuntos de características candidatas de tamanho 5, 10, 15, 20 e 30 para cada série temporal. Para cada conjunto foram realizadas execuções do FPSO e previsões sobre o conjunto de treinamento, com base nos resultados desses experimentos é que foram escolhidos o número de características candidatas para cada série temporal utilizada neste trabalho.

Na metodologia empregada neste trabalho, cada partícula contém um modelo de previsão e a posição armazenada por ela é convertida para o formato binário. O vetor de bits determina quais características serão utilizadas para treinamento do modelo embutido na partícula. Após treinar o modelo de previsão é possível verificar o desempenho do mesmo ao prever o conjunto de treinamento, que foi medido pelo MAPE (5.4). Com isso, é possível determinar se a posição da partícula está perto ou não de uma solução ótima.

Sendo assim, supondo uma situação na qual o número de características candidatas foi 5, sabe-se que o espaço de busca contém  $2^5 - 1 = 31$  posições possíveis. Se uma partícula da população tem o valor 9 como melhor posição, então, escrevendo em binário,  $9 = [0, 1, 0, 1, 0]$ . Nesse contexto, o valor 1 indica que a característica foi selecionada enquanto que o valor 0 indica que a mesma não foi selecionada. Pode-se notar que a posição da partícula indica que a segunda e a quarta característica foram selecionadas. Para uma série temporal  $T = \{3, 5, 2, 4, 6, 7, 5, 3, 7, 2\}$ , sabendo que para 5 características candidatas podemos realizar previsões a partir da sexta observação, é possível montar a base de dados abaixo:

5,4,7  
 2,6,5  
 4,7,3  
 6,5,7  
 7,3,2

Os termos sublinhados representam os valores alvo da previsão e os termos não sublinhados representam as entradas do modelo de previsão, características selecionadas para o valor alvo da previsão. Ou seja, para o valor alvo 7, as características selecionadas foram a segunda, cujo valor é 4 e a quarta, cujo valor é 5. É importante observar que apenas é possível fazer previsões a partir do sexto valor da série temporal devido ao número de características candidatas que foi escolhido. De uma forma geral se o número de características candidatas for  $X$ , então só é possível fazer previsões a partir da observação  $X + 1$ .

#### 5.1.4 Métricas para avaliação das previsões

A avaliação das previsões é realizada tendo como base a Equação (5.1) que calcula o erro entre o valor esperado para a previsão  $T_t$  e o valor previsto  $Y_t$  ambas no instante de tempo  $t$ . Quatro métricas foram utilizadas para avaliar o desempenho das previsões: o MSE (*Mean Square Error*), a raiz quadrada do MSE (RMSE), o MAPE (*Mean Absolute Percentage Error*) e a estatística U de THEIL, mostradas nas Equações (5.2), (5.3), (5.4) e (5.5), respectivamente.

$$E_t = (T_t - Y_t) \quad (5.1)$$

O MSE é muito utilizado para avaliar a precisão dos modelos de previsão de séries temporais. Na equação mencionada,  $N$  é o número de observações da série temporal. A meta de qualquer modelo de previsão é obter o menor MSE possível fazendo previsão de dados desconhecidos.

$$MSE = \frac{1}{N} \sum_{t=1}^N (E_t)^2 \quad (5.2)$$

O RMSE foi utilizado para comparar os resultados obtidos nesta dissertação com os resultados de um trabalho anterior [25].

$$RMSE = \sqrt{\frac{1}{N} \sum_{t=1}^N (E_t)^2} \quad (5.3)$$

O MAPE, mostrado na Equação (5.4) mede o erro médio percentual absoluto, o seu valor não depende da escala ou dos valores apresentados pela série analisada, pois seu resultado é um valor percentual. Quanto menor o MAPE, mais precisa é a previsão.

$$MAPE = \frac{1}{N} \sum_{t=1}^N \left| \frac{E_t}{Y_t} \right|, \text{ para } Y_t \neq 0 \quad (5.4)$$

O valor do THEIL mede o desempenho do modelo de previsão em relação a uma previsão feita através de passeio aleatório (*Random Walk*). A previsão através de passeio aleatório consiste em atribuir à próxima observação futura o último valor conhecido da série acrescido de um ruído aleatório. A meta de qualquer modelo de previsão é obter o

menor valor possível para o THEIL, que deve ser menor que um, caso contrário, é mais vantajoso utilizar a previsão através de passeio aleatório. Se o THEIL for igual a um, tanto faz utilizar o modelo proposto ou o passeio aleatório.

$$THEIL = \frac{\sum_{t=1}^N (E_t)^2}{\sum_{t=1}^N (T_t - T_{t-1})^2} \quad (5.5)$$

## 5.2 COMENTÁRIOS FINAIS

Neste capítulo foi apresentada a metodologia de previsão desenvolvida. Foi mostrada a forma de divisão do conjunto de dados para definição do conjunto de treinamento. Para essa tarefa foi utilizada a técnica *FastDTW*. Outro ponto da metodologia é a montagem do conjunto de dados tendo como base as características selecionadas pelo FPSO.

Por fim, o capítulo apresentou detalhes sobre as métricas de desempenho utilizadas para avaliar as previsões deste trabalho: MSE, RMSE, MAPE e THEIL.

# EXPERIMENTOS

Este capítulo contém a descrição de todos os passos que foram executados nos experimentos. A abordagem de previsão proposta neste trabalho foi testada em oito séries temporais provenientes de contextos diferentes. Os experimentos foram realizados em uma máquina Quad-Core com 4Gb de memória RAM. Todo ambiente de teste foi desenvolvido com a linguagem Java [43] e o software Matlab [44]. O banco de dados Postgres [45] foi usado para armazenamento das configurações dos experimentos e dos resultados. Os modelos de previsão utilizados pertencem à API da ferramenta de mineração de dados Weka [46]. Os testes de hipótese foram feitos usando o ambiente R [47].

O capítulo é organizado da seguinte maneira: Primeiro, são apresentados detalhes sobre as séries temporais usadas nos experimentos. Segundo, são expostos, analisados e comparados os resultados obtidos com outros reportados na literatura. Por último, são apresentadas algumas conclusões baseadas em testes de hipótese.

### 6.1 ETAPAS DE UM EXPERIMENTO

A execução completa de um experimento resume-se a cinco etapas, são elas:

1. Pré-processamento dos dados
2. Seleção do conjunto de treinamento.
3. Seleção de características.
4. Treinamento.
5. Previsão.

No pré-processamento, a série pode sofrer a transformação de Box-Cox para eliminação de tendência e sazonalidade, diferenciação para tornar a série estacionária ou ambos. O primeiro é feito após a análise dos gráficos média  $\times$  desvio. A diferenciação foi aplicada nas séries não estacionárias, a não estacionariedade foi verificada através do teste KPSS (*Kwiatkowski-Phillips-Schmidt-Shin*).

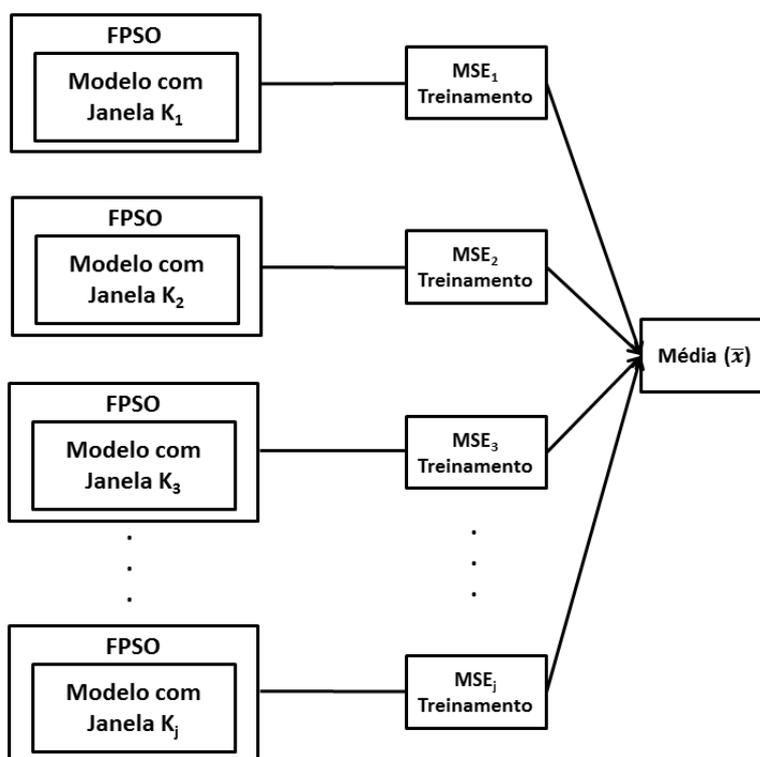
$$\{L_t = \ln(X_t) | t \in \{1, 2, 3, \dots, N\}\} \quad (6.1)$$

A Equação (6.1) mostra como foi aplicada a transformação de Box-Cox [31, 32]. Para cada série foi traçado o gráfico média  $\times$  desvio, em quatro delas considerou-se necessária a aplicação do procedimento da Equação (6.1).

Para analisar a necessidade da diferenciação, foram plotados os gráficos de Média Amostral e da Autocovariância Amostral de cada série. Ao perceber variação de pelo

menos uma dessas propriedades foi aplicada a diferenciação na série, conforme a Equação (2.6). Além disso, foi realizado o teste de estacionriedade KPSS em cada série temporal para validar a análise gráfica.

Terminada a etapa de pré-processamento, o segundo passo de um experimento é a seleção do conjunto de treinamento. Tal seleção foi realizada através da análise de similaridade entre janelas de dados. Para tal, o algoritmo *FastDTW* foi utilizado. Alguns tamanhos de janelas foram testados, como pode ser visto na Tabela 6.4. O método de escolha da janela é mostrado na Figura 6.1. A figura retrata que uma execução foi realizada para cada janela e mediu-se o MSE no conjunto de treinamento para o modelo construído. Após isso, foi obtido o MSE médio. O modelo cujo MSE mais se aproximou do valor médio teve seu tamanho de janela escolhido.

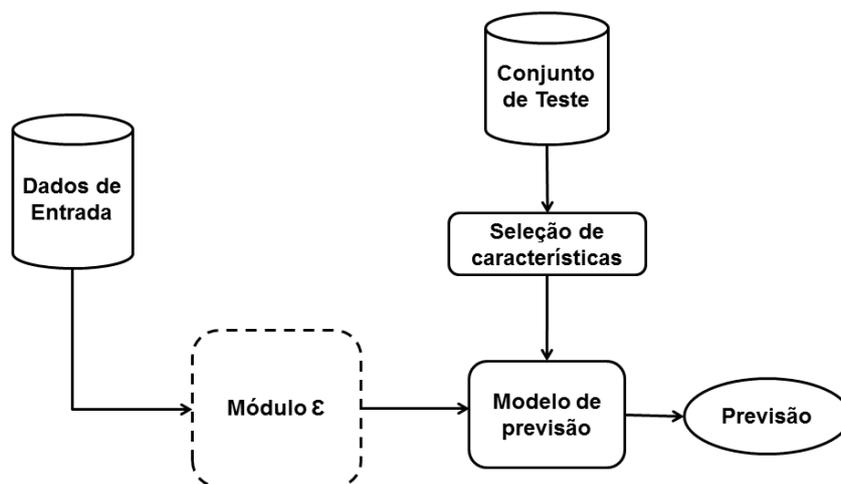


**Figura 6.1** Abordagem de escolha da janela para o *FastDTW*

Pela Figura 6.1, sendo  $\delta = \min\{\bar{x} - MSE_j\}$  e  $(\bar{x} - MSE_p) = \delta$  então o modelo escolhido para realizar os experimentos será o modelo com janela  $K_p$ . Para evitar que o menor erro no treinamento esteja atrelado ao tamanho do conjunto de treinamento utilizado, foi escolhida a janela cujo MSE foi mais próximo do valor médio dessa medida.

A terceira etapa é a seleção das características para treinamento do modelo. Nesta etapa o algoritmo FPSO foi utilizado. Ao término da execução do FPSO, tem-se um modelo de previsão treinado com as características consideradas ótimas e com o conjunto de treinamento definido. Por fim, o conjunto de teste que não foi utilizado nas etapas anteriores é utilizado para montar a base de dados de teste com as mesmas características

que foram selecionadas na etapa de seleção. A partir dessa base de teste avalia-se a capacidade de previsão dos modelos treinados. A Figura 6.2 mostra o fluxo completo da execução de um experimento.



**Figura 6.2** Formatação dos experimentos

O módulo  $\varepsilon$  em destaque na Figura 6.2 é formado pelos componentes de pré-processamento, seleção do conjunto de treinamento e seleção das características conforme foi mostrado na Figura 5.1. Dois modelos de previsão foram usados nos experimentos: redes do tipo MLP e SVR. Todos os experimentos foram repetidos 30 vezes para extrair a média ( $\bar{x}$ ) e o desvio padrão ( $s$ ) dos mesmos. As métricas utilizadas variam de acordo com a série utilizada.

Nas subseções seguintes são expostos detalhes sobre as séries usadas e discutidos em detalhes os experimentos para cada uma delas.

## 6.2 SÉRIES TEMPORAIS UTILIZADAS

- ESTSP1: Foi fornecida pelo *European Symposium on Time Series Prediction* (ESTSP'08), composta de 354 observações. Não se têm informações a respeito da origem desta série, pois, por se tratar de uma série de competição, poucas ou nenhuma informação é fornecida. Esta série foi escolhida para com o objetivo de comparar os resultados da metodologia proposta com os reportados anteriormente [16, 48].
- ESTSP2: Foi fornecida pelo ESTSP'08, composta de 1300 observações. O nível de informações a respeito desta série é o mesmo apresentado pela série ESTSP1 [16, 48].
- Pollution: É composta por 130 observações mensais de transferências em equipamentos de poluição (em milhares de francos) de janeiro de 1966 a outubro de 1976 [25, 49].
- Paper: É composta de 120 observações mensais das vendas da indústria de papel (em milhares de francos) de janeiro de 1963 a dezembro de 1972 [25, 49].

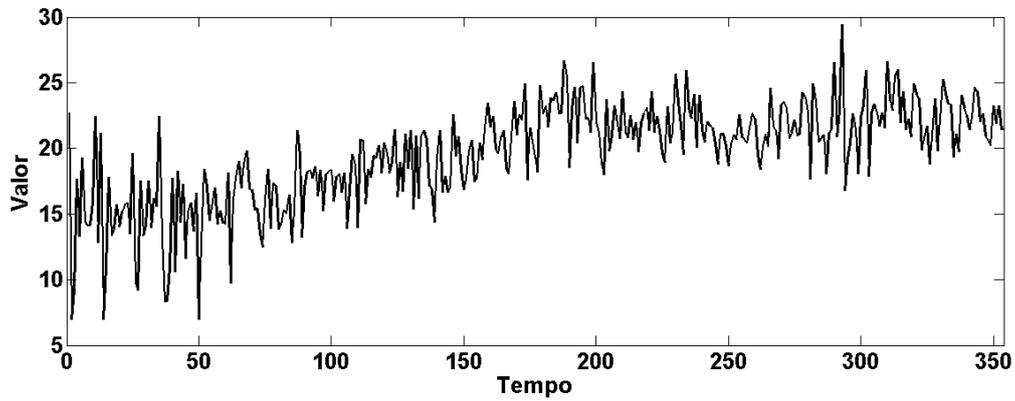


Figura 6.3 Representação gráfica da série ESTSP1

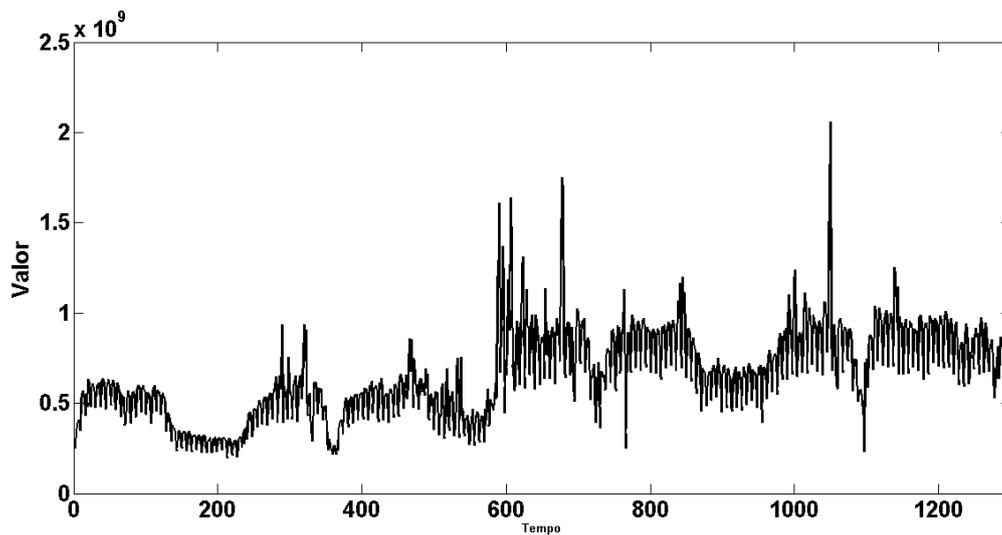


Figura 6.4 Representação gráfica da série ESTSP2

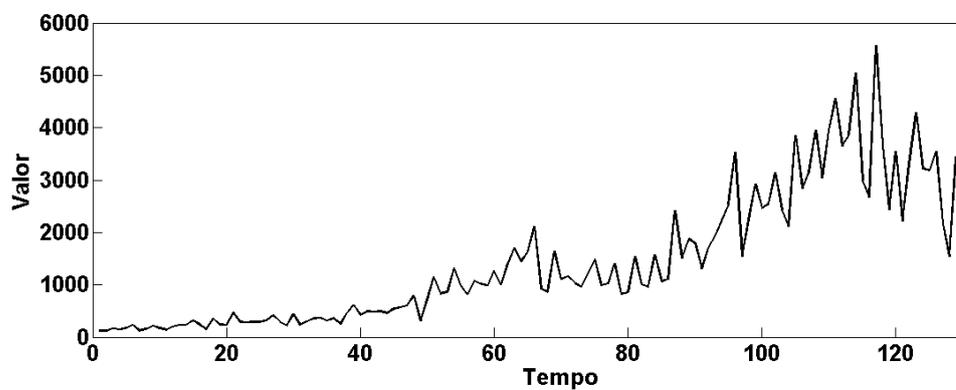


Figura 6.5 Representação gráfica da série Pollution

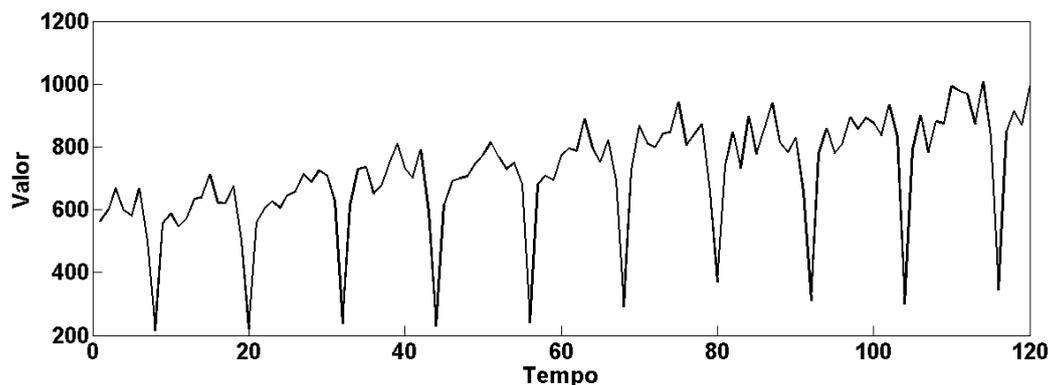


Figura 6.6 Representação gráfica da série Paper

- PETR4: Contém os valores de fechamento da ação da Petrobrás a partir de 03 janeiro de 2007 até 06 de julho de 2012 [50]. Totalizando 1379 observações. Esta série foi escolhida para realizar comparações com as previsões desenvolvidas em trabalho anteriores [19, 26].

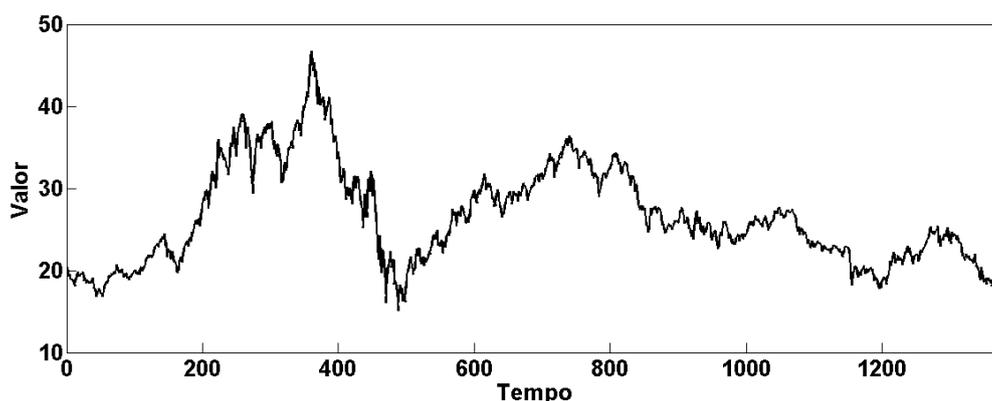


Figura 6.7 Representação gráfica da série PETR4

- VALE5: Formada pelos valores de fechamento da ação da Companhia Vale do Rio Doce a partir de 01 de janeiro de 2007 até 06 de julho de 2012 [50]. Totalizando 1381 observações. Esta série foi escolhida para realizar comparações com as previsões desenvolvidas em trabalho anteriores [19, 26].
- Poluição Monóxido de Carbono (PCO): É composta por 365 observações que correspondem aos valores de emissão diária de monóxido de carbono na cidade de São Paulo do dia 01 de janeiro até o dia 31 de dezembro de 1997 [51]. Esta série foi escolhida para que seus resultados fossem comparados com outros reportados anteriormente [18]. A Figura 6.9 mostra a representação gráfica da série.
- Umidade: Série composta por 365 observações da umidade relativa do ar ao meio dia, na cidade de São Paulo. As observações são diárias no período de 01 de janeiro

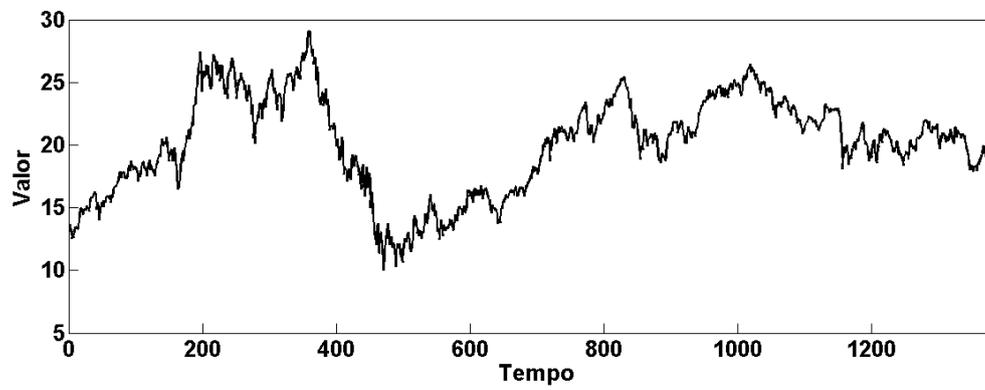


Figura 6.8 Representação gráfica da série VALE5

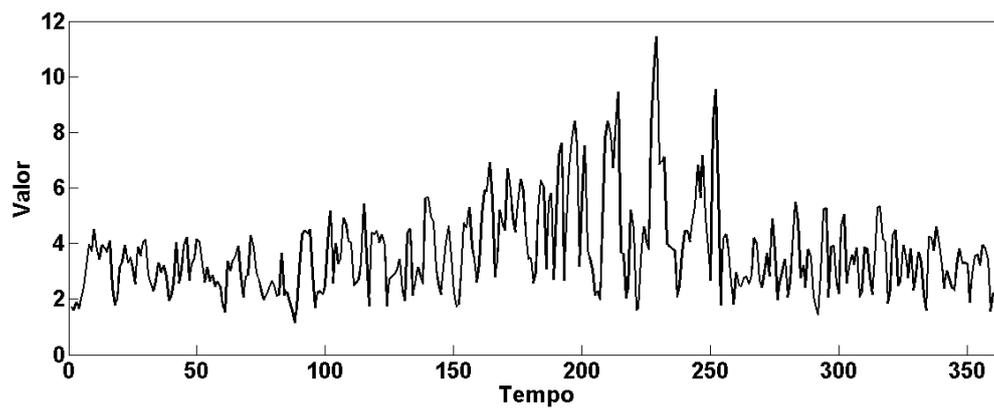
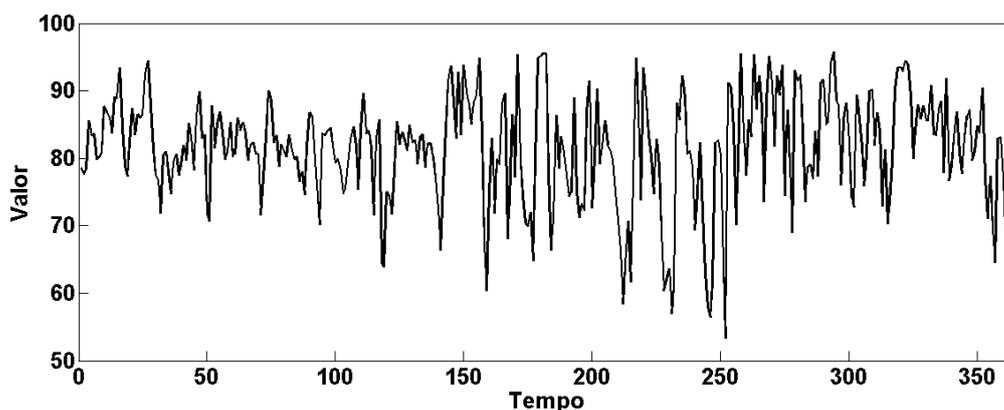


Figura 6.9 Representação gráfica da série PCO

Série Temporal	Mínimo	Máximo	Média	Variância	Desvio Padrão	Tamanho
ESTSP1	7	29,43	19,53	14,33	3,78	354
ESTSP2	$1,98 \cdot 10^8$	$2,05 \cdot 10^9$	$6,47 \cdot 10^8$	$5,59 \cdot 10^{16}$	$2,36 \cdot 10^8$	1300
Pollution	120,88	5566,1	1439,3	$1,59 \cdot 10^6$	$1,26 \cdot 10^3$	130
Paper	215,187	1006,9	713,74	30294	174,05	120
PETR4	15,15	46,69	26,44	36,01	6,0	1379
VALE5	10,02	29,07	20,07	15,03	3,87	1380
PCO	1,14	11,44	3,67	2,52	1,59	365
Umidade	53,34	95,79	81,15	63,33	7,96	365

**Tabela 6.1** Propriedades das séries utilizadas neste trabalho

de 1997 até 31 de dezembro do mesmo ano [51]. A série foi escolhida para efeito de comparação dos resultados obtidos com outros reportados anteriormente [18]. A Figura 6.10 mostra a representação gráfica da série.



**Figura 6.10** Representação gráfica da série umidade

A Tabela 6.1 expõe algumas propriedades importantes para cada série mencionada anteriormente.

Na Tabela 6.2,  $W$  é o número de neurônios na camada intermediária (sendo  $n_c$  o número de entradas da rede neural),  $E$  é número de épocas que o algoritmo de treinamento executou durante o treinamento da rede neural,  $L$  é a taxa de aprendizado da rede e  $M$  o seu termo de momento. Os valores dos parâmetros da MLP foram escolhidos através da execução de diversos experimentos, combinando os possíveis valores para cada parâmetro. Para a SVR, foram realizados experimentos com dois tipos de *kernel*, o Linear e o com funções de base radial (RBF). Os parâmetros utilizados foram o  $C$  que determina a suavidade do modelo, o  $\varepsilon$  e o  $\gamma$ . O  $\gamma$  é o expoente utilizado na função de *kernel* da SVR. Os valores dos parâmetros da SVR foram determinados por um conjunto de experimentos realizados.

Na Tabela 6.3, os parâmetros  $w_{min}$  e  $w_{max}$  são os limites do peso da inércia, o algoritmo começa com o valor máximo do peso da inércia e termina com o valor mínimo.  $\phi$  é a

**Tabela 6.2** Parâmetros dos modelos de previsão e os valores que foram investigados nesta dissertação

Modelo	Parâmetros	Valores dos Parâmetros
MLP	$W$	$\left\{ \left\lfloor \frac{(n_c+1)}{2} \right\rfloor \right\}$
	$E$	$\{ 50;100;200;400 \}$
	$L$	$\{ 0,01;0,1 \}$
	$M$	$\{ 0,02;0,2;0,5 \}$
SVR Linear	$C$	$\{1,2\}$
	$\varepsilon$	$\{0,001;0,01;0,1 \}$
SVR RBF	$C$	$\{1,2\}$
	$\varepsilon$	$\{0,001;0,01;0,1 \}$
	$\gamma$	$\{0,001;0,01;0,1 \}$

**Tabela 6.3** Parâmetros do FPSO

Algoritmo	Parâmetros	Valores dos Parâmetros
FPSO	$S$	$\{10,20,30\}$
	$F$	$\{5,10,15,20,30\}$
	$w_{max}$	0.9
	$w_{min}$	0.4
	$k$	$\{2S, 3S\}$
	$\phi$	4

soma dos coeficientes de aceleração. O parâmetro  $k$  é utilizado no processo de poda. Finalmente,  $S$  é o tamanho do enxame. Na Tabela 6.4 são mostrados os tamanhos das janelas investigados na etapa de seleção do conjunto de treinamento. Os tamanhos experimentados começam com a menor janela possível, evitando que o conjunto treinamento seja menor que o conjunto de teste e termina com a maior janela possível de acordo com a disponibilidade dos dados.

**Tabela 6.4** Tamanhos dos subconjuntos de dados usados no DTW para cada série temporal

Série Temporal	Tamanhos de janelas investigados
ESTSP1	{50,55,60,65,70,75,80,85,90,95,100,105,110}
ESTSP2	{50,100,150,200,250}
Paper	{10,15,20,25,30,35,40,45}
Pollution	{10,15,20,25,30,35,40,45}
PETR4	{100,150,200,250,300,350,400,450}
VALE5	{100,150,200,250,300,350,400,450}
PCO	{50,55,60,65,70,75,80,85,90,95,100,105,110}
Umidade	{50,55,60,65,70,75,80,85,90,95,100,105,110}

Na Tabela 6.5, são mostradas as legendas das abordagens utilizadas neste trabalho. Na referida tabela, o TR indica se foi aplicado o tratamento dos dados.

**Tabela 6.5** Legenda das metodologias

Legenda	Metodologia
DTFM	DTW+TR+FPSO+MLP
DTFSL	DTW+TR+FPSO+SVR Linear
DTFSR	DTW+TR+FPSO+SVR RBF
DFM	DTW+FPSO+MLP
DFSL	DTW+FPSO+SVR Linear
DFSR	DTW+FPSO+SVR RBF
FM	FPSO+MLP
FSL	FPSO+SVR Linear
FSR	FPSO+SVR RBF

### 6.3 RESULTADOS

Esta seção expõe, compara e discute os resultados obtidos para cada uma das séries temporais utilizadas neste trabalho. Na Figura 6.11 são representados todos os passos executados pela metodologia, desde o recebimento de uma série temporal até a responder com a previsão do conjunto de teste. Vale salientar que são executadas diferenciações até que a série torne-se estacionária. Neste trabalho foi necessária apenas uma diferenciação para cada série estudada. Nas tabelas de resultados as metodologias com asterisco foram as propostas nesta dissertação.

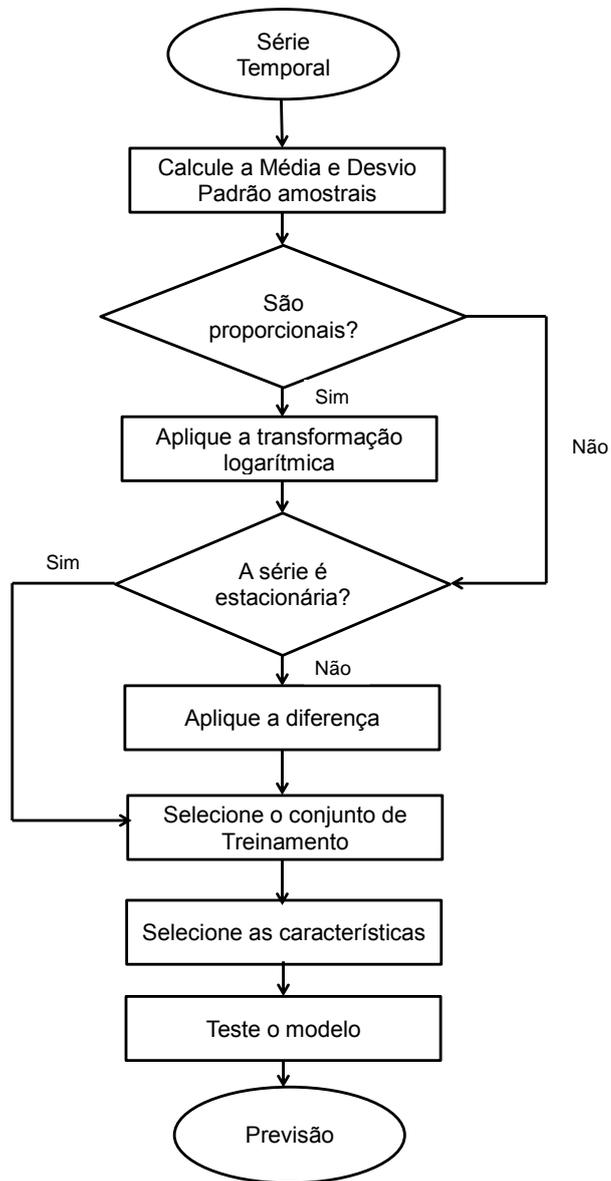


Figura 6.11 Estrutura básica da metodologia proposta

### 6.3.1 Experimentos com a série ESTSP1

Para os experimentos realizados com a base ESTSP1, foi utilizado o MSE como medidor de desempenho. A técnica *MLP Ensemble* propôs a combinação de filtros e *wrappers* para seleção de características [16] e também usou essa métrica. O resultado da *MLP Ensemble* foi o segundo colocado no *European Symposium on Time Series Prediction 2008* (ESTSP'08). O conjunto de teste foi composto por 18 padrões.

Uma pesquisa recente investigou o desempenho do FPSO [21] em relação à implementação original do PSO [35] para selecionar as características [19]. Foram feitos experimentos com redes MLP e modelos de SVR para a previsão. Nesses experimentos o conjunto de teste foi composto por 97 padrões. Apesar de utilizar um conjunto de teste maior, os resultados obtidos foram melhores em termos de MSE. Além disso, foi possível concluir que o FPSO é melhor que a variação do PSO utilizada, além de exigir menos esforço computacional devido aos mecanismos que possui.

A primeira etapa dos experimentos é verificar se a transformação logarítmica é necessária. A análise do gráfico média  $\times$  desvio, mostrado na Figura 6.12, para a série ESTSP1 demonstra que não foi necessária essa transformação, pois não há relação de proporcionalidade entre a média e o desvio padrão das janelas de dados da série [27]. A análise da estacionariedade apresentou indícios de que a série ESTSP1 não é estacionária, conforme pode ser visto pelos gráficos da média amostral na Figura 6.13 e da função de autocovariância na Figura 6.14, nos quais, nem a média amostral nem a autocovariância são constantes. Outro indicador da não estacionariedade da série foi o teste estatístico KPSS, o teste consiste em supor na hipótese nula que a série é estacionária. No teste executado houve a rejeição da hipótese nula com 99% de confiança. Após uma diferenciação, não houve indícios para rejeitar a hipótese nula de estacionariedade, por isso a série tornou-se estacionária após apenas uma diferenciação.

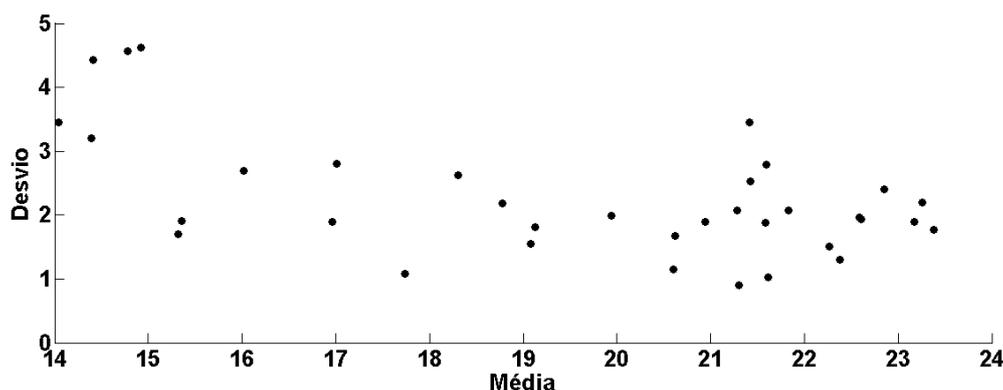


Figura 6.12 Média  $\times$  Desvio para a série ESTSP1

Passada a etapa de tratamento dos dados, a seleção do conjunto de treinamento foi realizada. O único parâmetro nesta etapa é o tamanho da janela utilizado na análise do *FastDTW*. Janelas de diferentes tamanhos foram testadas, conforme pode ser visto na Tabela 6.4. Após esses experimentos foi utilizada a abordagem demonstrada na

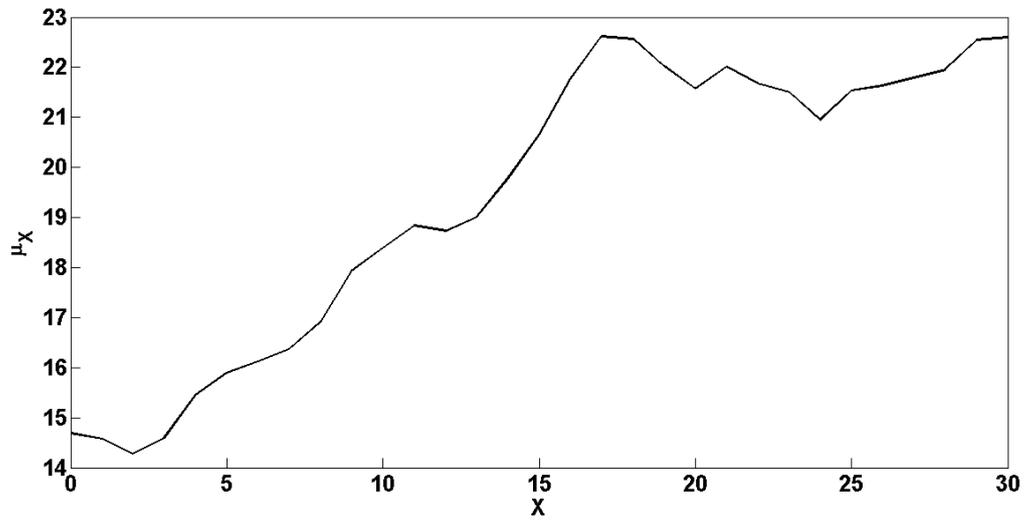


Figura 6.13 Gráfico da média amostral para a série ESTSP1

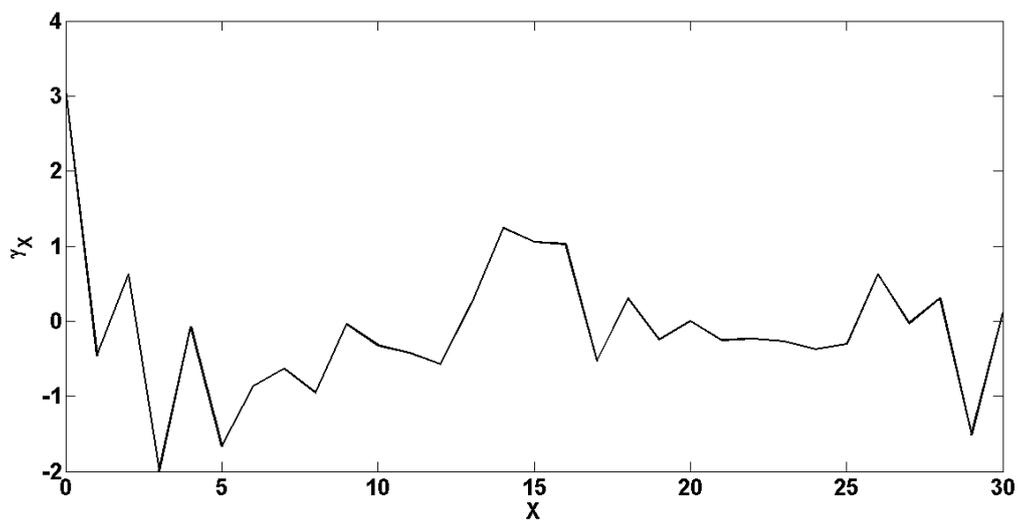


Figura 6.14 Gráfico da autocovariância amostral para a série ESTSP1

Figura 6.1. Ao utilizar a MLP como modelo, o conjunto de treinamento teve 67 padrões. Já para a SVR, 87 padrões. Para selecionar as características, o parâmetro mais importante é o número de *lags* usados no espaço de busca. Alguns valores foram testados, conforme pode ser visto na Tabela 6.3. Para essa série, os melhores resultados foram produzidos para um espaço de busca composto por 5 *lags* e com um enxame de tamanho 10. A Tabela 6.6 mostra os resultados obtidos e o reportado pela técnica *MLP-Ensemble* [16]. Os seguintes experimentos foram realizados: FPSO + Modelo, DTW + FPSO + Modelo e, por último DTW + Tratamento dos dados + FPSO + Modelo. Pode-se perceber que a metodologia empregada obteve melhores resultados que a *MLP-Ensemble* [16] tanto para a MLP quanto para a SVR como modelos de previsão em conjunto de teste de tamanho 18. Vale salientar que os resultados apresentados pela metodologia proposta são mostrados em termos de média ( $\bar{x}$ ) e desvio padrão ( $s$ ) de 30 execuções, enquanto que o resultado apresentado pela técnica *MLP-Ensemble* é o melhor obtido por ela [16].

O número de características utilizadas para a construção dos modelos de previsão foi bem menor ao utilizar a metodologia proposta. Ou seja, os modelos construídos são parcimoniosos em relação ao número de características, fator importante na definição de modelos de previsão. Os resultados para os modelos MLP e SVR foram superiores a todos os encontrados até o momento. Dentre esses dois modelos, o SVR apresentou resultados ainda melhores que o MLP, fato que comprova a grande capacidade do modelo em generalizar a partir de um pequeno conjunto de treinamento.

**Tabela 6.6** Resultados para a série ESTSP1

Técnica(Parâmetros)	MSE $\bar{x}(s)$
MLP-Ensemble	4,76
FSR( $S=20;F=10;\varepsilon = 0,1;\gamma = 0,001$ )	3,76(0,044)
DFSR( $S=10;F=5,\varepsilon = 0,01;\gamma = 0,5;C=1$ )	2,4402(0,0322)
DTFSR*( $S=10;F=5;\varepsilon = 0,01;\gamma = 0,001;C=1$ )	0,1139(7,0.10 <sup>-17</sup> )
FM( $S=20;F=10;L=0,01;M=0,02;E=100$ )	2,24(1,35.10 <sup>-15</sup> )
DFM( $S=20;F=10;L=0,01;M=0,02;E=100$ )	2,11(0,0331)
DTFM*( $S=10;F=5;L=0,1;M=0,2;E=50$ )	1,85(1,35.10 <sup>-15</sup> )

A Figura 6.15 mostra uma comparação entre o número de padrões utilizados antes e depois de fazer a seleção do conjunto de treinamento. Houve melhora de precisão com redução de aproximadamente 78% no número de padrões no conjunto de treinamento. Essa série foi a que apresentou maior redução no conjunto de treinamento.

A Figura 6.16 mostra o comportamento do gráfico da série prevista e da série real no conjunto de teste.

### 6.3.2 Experimentos com a série ESTSP2

Para esta série, os resultados também foram reportados em termos do MSE. Essa escolha deve-se à comparação com a metodologia *MLP Ensemble* [16]. O conjunto de teste utilizado foi composto por 100 observações.

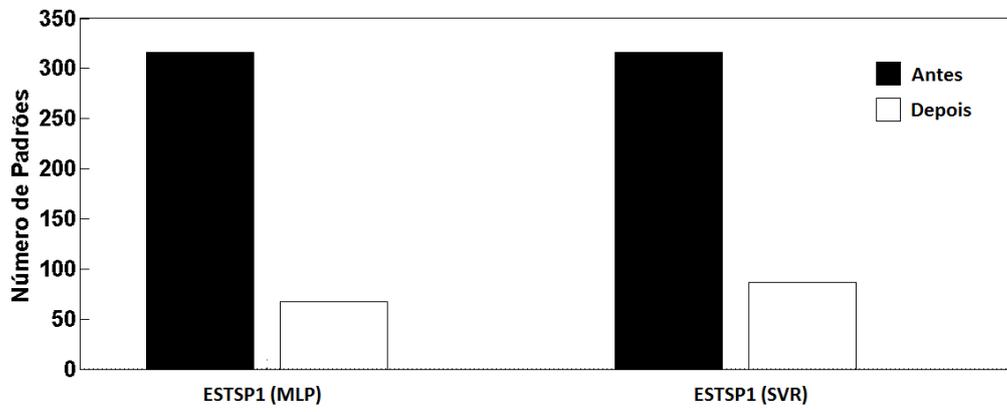


Figura 6.15 Número de Padrões de treinamento antes e depois da seleção na Série ESTSP1

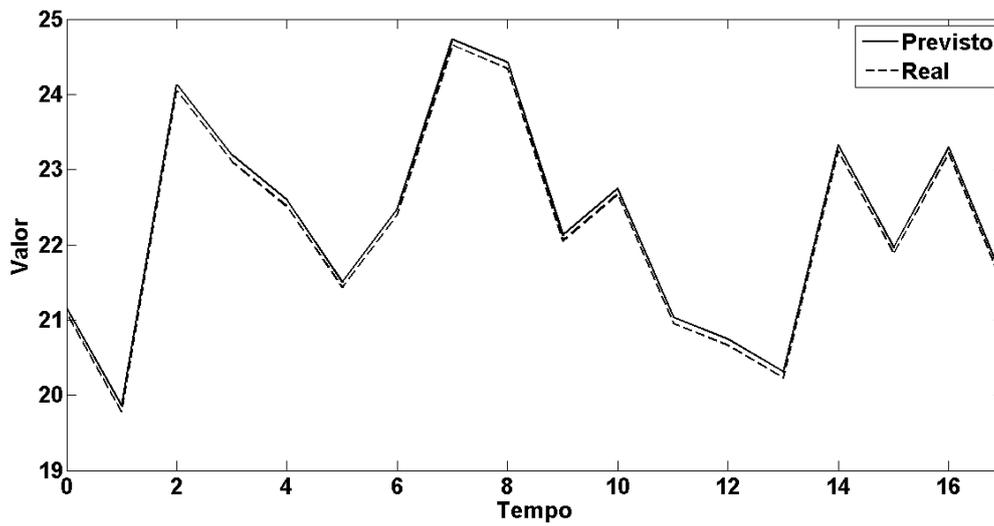
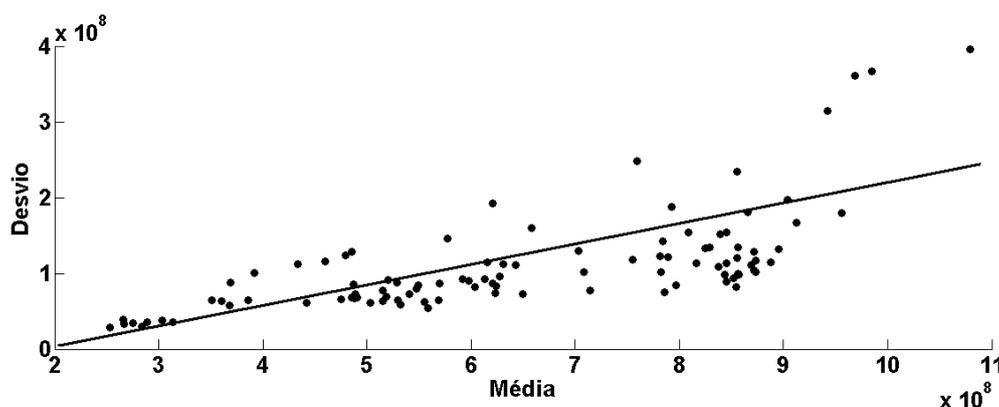


Figura 6.16 Gráfico previsto  $\times$  real para a série ESTSP1

Resultados preliminares com o FPSO e modelos MLP e SVR [19] já conseguiram melhorar o desempenho obtido pela *MLP Ensemble* [16]. Tais resultados utilizaram uma divisão da série temporal em conjunto de treinamento e conjunto de teste. Este último foi composto por 268 padrões e, ainda assim, os resultados obtidos foram melhores em termos de MSE quando o modelo SVR foi utilizado. Comparações de desempenho entre o FPSO e o PSO original também foram realizadas para essa base e mais uma vez o FPSO produziu resultados melhores tanto no tempo de execução quanto no número de características selecionadas.

Ao analisar o gráfico média  $\times$  desvio, mostrado na Figura 6.17, nota-se uma certa proporcionalidade entre a média amostral e o desvio padrão amostral, por isso foi aplicada a transformação logarítmica expressa na Equação (6.1)[27].



**Figura 6.17** Gráfico desvio  $\times$  média na série ESTSP2

Após aplicar a transformação logarítmica, uma análise da média amostral mostrada na Figura 6.13 e da autocovariância amostral apresentada pela Figura 6.14 sugere que autocovariância adquiriu um comportamento constante. O mesmo não aconteceu com a média amostral, por isso, essa série não pode ser considerada estacionária. A variância da média amostral e da autocovariância podem ser vistas na Tabela 6.7 e confirma que há variação na média amostral. Além da análise gráfica, foi aplicado o teste estatístico KPSS e foi rejeitada com 99% de confiança a hipótese de estacionariedade. Devido a essas análises, foi aplicada uma diferenciação na série para torná-la estacionária. Após uma diferenciação o mesmo teste KPSS foi executado e não houve rejeição da hipótese de estacionariedade, sendo assim pode-se considerar com 99% de certeza que a série tornou-se estacionária após apenas uma diferenciação.

Média Amostral	Autocovariância Amostral
0,115	$1,83 \cdot 10^{-4}$

**Tabela 6.7** Variância da média amostral e da autocovariância para a série ESTSP2 após transformação logarítmica

A análise de similaridade para seleção do conjunto de treinamento foi executada em janelas com alguns tamanhos, conforme pode ser visto na Tabela 6.4. Ao utilizar a MLP

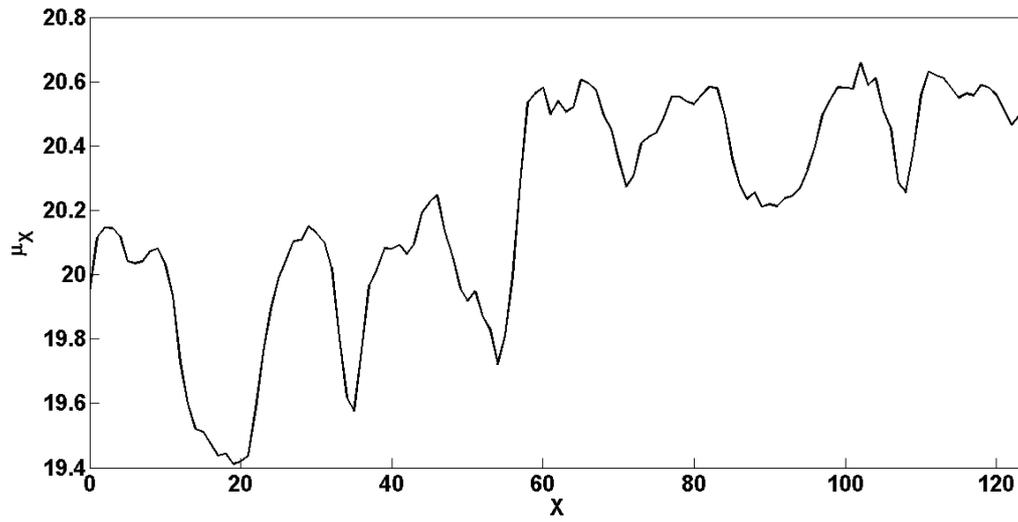


Figura 6.18 Gráfico da média amostral para a série ESTSP2

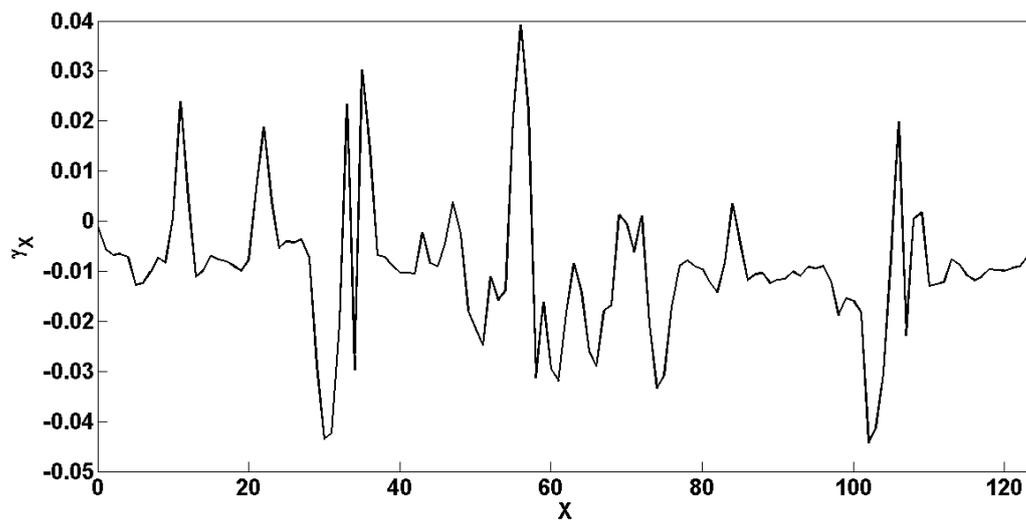


Figura 6.19 Gráfico da autocovariância amostral para a série ESTSP2

e SVR, os conjuntos de treinamento tiveram 299 e 450 padrões respectivamente. Na seleção das características, os melhores resultados foram produzidos para um espaço de busca composto de tamanho 10, tanto para a MLP quanto para a SVR. Os resultados alcançados neste trabalho foram melhores em relação aos obtidos pela *MLP-Ensemble* [16] utilizando MLP e também SVR ao prever um conjunto com 100 padrões. A Tabela 6.8 mostra os resultados para cada um dos experimentos realizados.

É importante salientar que os resultados em termos de média e desvio padrão para 30 execuções foram superiores ao melhor MSE obtido pela metodologia *MLP-Ensemble* [16].

A metodologia aqui proposta conseguiu melhorar os resultados de trabalhos anteriores [16, 19] com um número menor de características. O número de *lags* e conjunto de treinamento sofreram reduções, proporcionando uma modelagem menos custosa, além disso, com maior precisão nas previsões. Os resultados para os modelos MLP e SVR foram superiores a todos os encontrados até o momento. Dentre esses dois modelos, a MLP foi superior ao SVR.

**Tabela 6.8** Resultados para a série ESTSP2

Técnica(Parâmetros)	MSE $\bar{x}(s)$
<i>MLP-Ensemble</i>	$1,01 \cdot 10^{16}$
FSL( $S=20; F=10; \varepsilon = 0,001; C=1$ )	$2,65 \cdot 10^{15}(1,6 \cdot 10^{14})$
DFSR( $S=10; F=10; \varepsilon = 0,001; \gamma = 0,5; C=2$ )	$8,65 \cdot 10^{15}(1,38 \cdot 10^{15})$
DTFSR*( $S=10; F=5; \varepsilon = 0,01; \gamma = 0,5; C=2$ )	$2,40 \cdot 10^{15}(6,17 \cdot 10^{14})$
FM( $S=20; F=30; L=0,1; M=0,2; E=200$ )	$4,18 \cdot 10^{15}(4,83 \cdot 10^{14})$
DFM( $S=20; F=10; L=0,1; M=0,2; E=200$ )	$4,02 \cdot 10^{15}(1,93 \cdot 10^{14})$
DTFM*( $S=20; F=10; L=0,01; M=0,05; E=50$ )	$3,2 \cdot 10^{14}(4,16 \cdot 10^{13})$

A Figura 6.20 mostra uma comparação entre o número de padrões utilizados antes e depois de fazer a seleção do conjunto de treinamento. Houve melhora de precisão com redução de aproximadamente 72% no número de padrões no conjunto de treinamento. Essa série apresentou segunda maior redução no conjunto de treinamento.

A Figura 6.21 mostra o comportamento da série prevista e da série real para conjunto de teste. Pode-se notar a metodologia de previsão composta pelo tratamento dos dados, seleção de características e seleção do conjunto de treinamento conseguiu modelar bem o comportamento da série em questão.

### 6.3.3 Experimentos com a série Pollution

Os experimentos para esta série foram reportados em termos de RMSE e MAPE. A escolha se deve ao fato das mesmas terem sido utilizadas para avaliar os resultados do *Seasonal Support Vector Regression* (SSVR)[25], os quais foram medidos para um conjunto de teste com 12 padrões.

Uma análise do gráfico média  $\times$  desvio para a série Pollution demonstra que existe uma relação de proporcionalidade entre a média e o desvio em janelas da série, como mostra a Figura 6.22. Por isso, foi aplicada a transformação logarítmica na série.

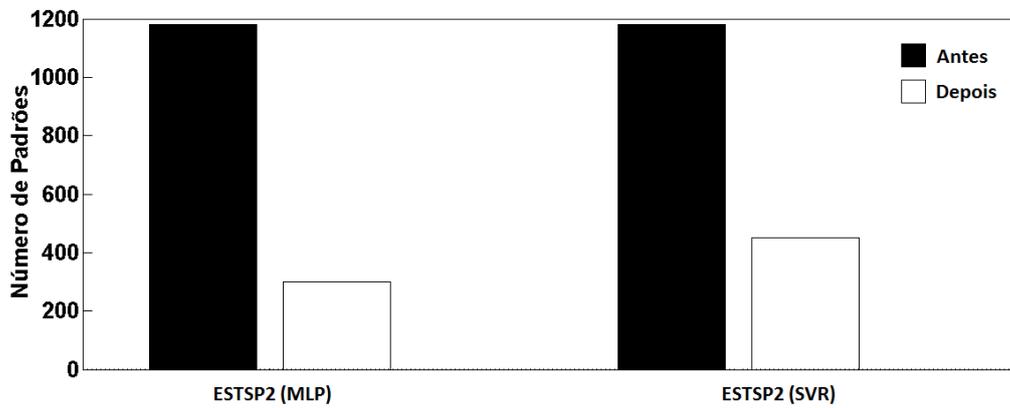


Figura 6.20 Número de Padrões de treinamento antes e depois da seleção na Série ESTSP2

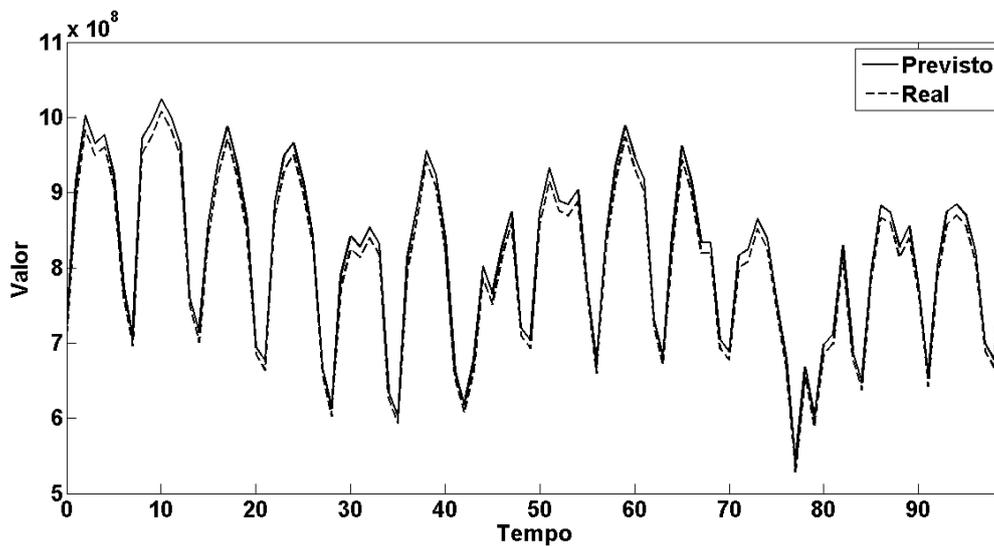


Figura 6.21 Gráfico previsto  $\times$  real para série ESTSP2

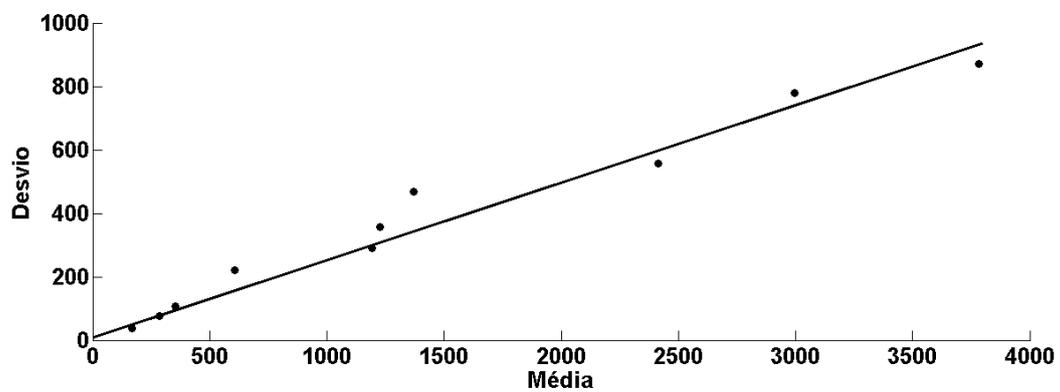
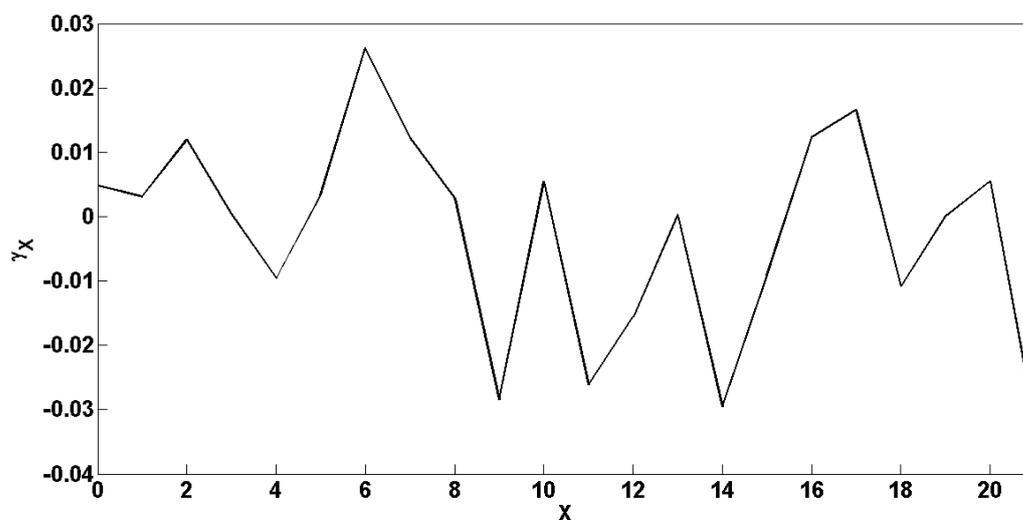


Figura 6.22 Gráfico desvio  $\times$  média na série Pollution

A Tabela 6.9 mostra que a variância das médias amostrais não está próxima de zero. Esse é um indicativo de que as médias amostrais sofrem variação, fato que implica a não estacionariedade da série. A Figura 6.23 e a Figura 6.24 mostram o comportamento amostral da média e da autocovariância. Pode-se notar que a média amostral tem um comportamento crescente enquanto que a autocovariância tem um comportamento que oscila em torno de zero. O teste KPSS também não rejeitou a hipótese de estacionariedade com 99% de confiança. Por isso, pode-se concluir que a série não é estacionária. Sendo assim, foi aplicada uma diferenciação na série e o teste KPSS verificou que após isso não houve rejeição da hipótese de estacionariedade.



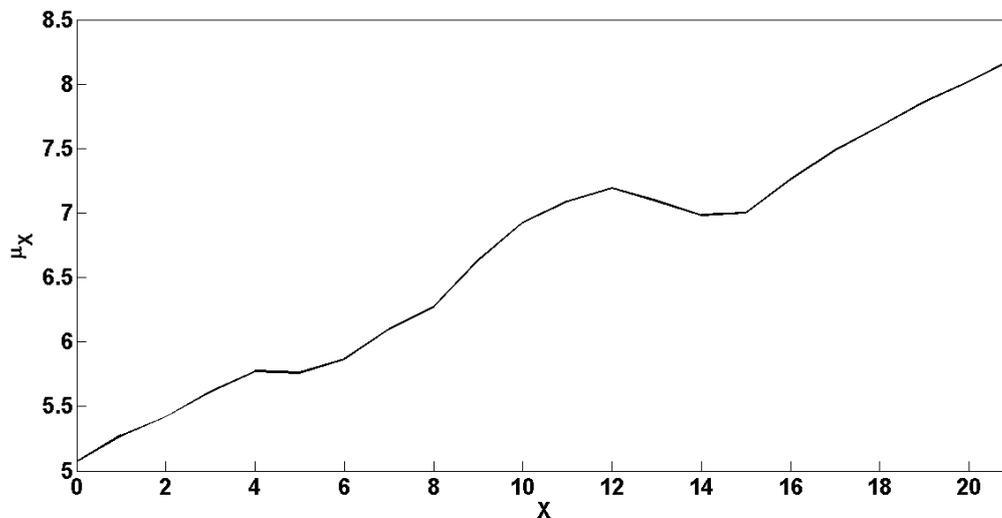
**Figura 6.23** Gráfico da autocovariância amostral para a série Pollution após transformação logarítmica

Média Amostral	Autocovariância Amostral
0,879	$2,52 \cdot 10^{-4}$

**Tabela 6.9** Variância da média amostral e da autocovariância para a série Pollution após transformação logarítmica

Na seleção do conjunto de treinamento, a janela selecionada dentre as possíveis teve 35 observações. Com isso, o conjunto de treinamento foi composto por 46 padrões, 52% menor que o utilizado pela abordagem (SSVR) [25]. O espaço de busca utilizado para a seleção de características foi formado por apenas 5 características. O FPSO encontrou um ponto de menor MAPE para os *lags* 1, 3 e 4.

A Tabela 6.10 expõe os resultados obtidos para cada experimento com esta série e mostra o melhor resultado obtido pela SSVR. Mais uma vez a seleção das características e do conjunto de treinamento proporcionaram um melhor desempenho que outras abordagens [25, 19]. Nota-se também que o número de características utilizadas para a cons-



**Figura 6.24** Gráfico da média amostral para a série Pollution após transformação logarítmica

trução dos modelos de previsão foi bastante reduzido em relação aos resultados apresentados em outros trabalhos [19].

**Tabela 6.10** Resultados para a série Pollution

Técnica(Parâmetros)	RMSE $\bar{x}(s)$	MAPE $\bar{x}(s)$
SSVR	881,451	0,1938
FSR( $S=20;F=20;\varepsilon = 0,01;\gamma = 0,5; C=2$ )	813,42(66,89)	0,2602(0,0255)
DFSL ( $S=10;F=5;\varepsilon = 0,001;C=1$ )	816,58(6,65)	0,1964(0,013)
DTFSL*( $S=10;F=5;\varepsilon = 0,001;C=2$ )	528,09(42,63)	0,1269(0,0110)
FM( $S=30;F=20;L=0,1;M=0,2;E=100$ )	623,31(37,66)	0,1972(0,0181)
DFM( $S=30;F=20;L=0,1;M=0,2;E=100$ )	667,35(110,36)	0,198(0,0314)
DTFM*( $S=20;F=5;L=0,01;M=0,02$ )	90,8(1,5.10 <sup>-4</sup> )	0,025(1,06.10 <sup>-17</sup> )

A Figura 6.25 mostra uma comparação entre o número de padrões utilizados antes e depois de fazer a seleção do conjunto de treinamento. Houve melhora de precisão com redução de aproximadamente 41% no número de padrões no conjunto de treinamento.

Na Figura 6.26 pode ser observado o comportamento da série prevista e da série real, ambas para conjunto de teste. Nota-se que a metodologia proposta conseguiu modelar bem o comportamento da série temporal.

### 6.3.4 Experimentos com a série Paper

Os experimentos com a série Paper também foram avaliados em termos de RMSE e o MAPE. Tais métricas foram escolhidas para comparação com resultados publicados anteriormente [19, 25]. A metodologia SSVR [25] produziu melhores previsões em comparação

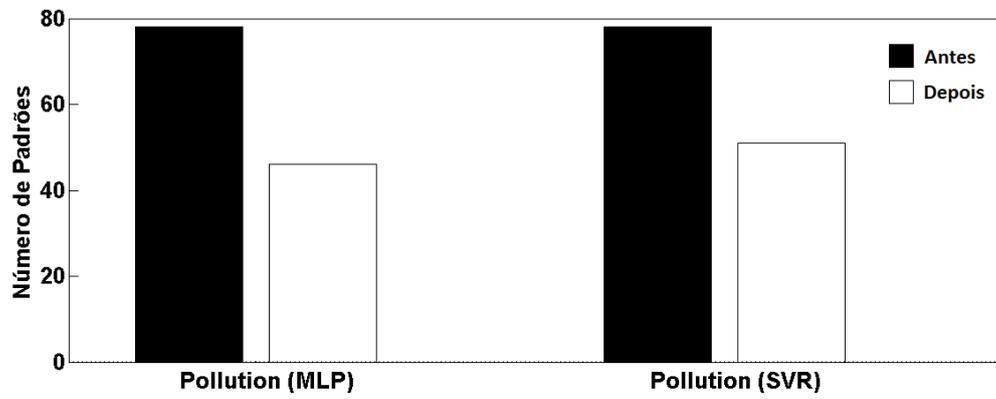


Figura 6.25 Número de Padrões de treinamento antes e depois da seleção na Série Pollution

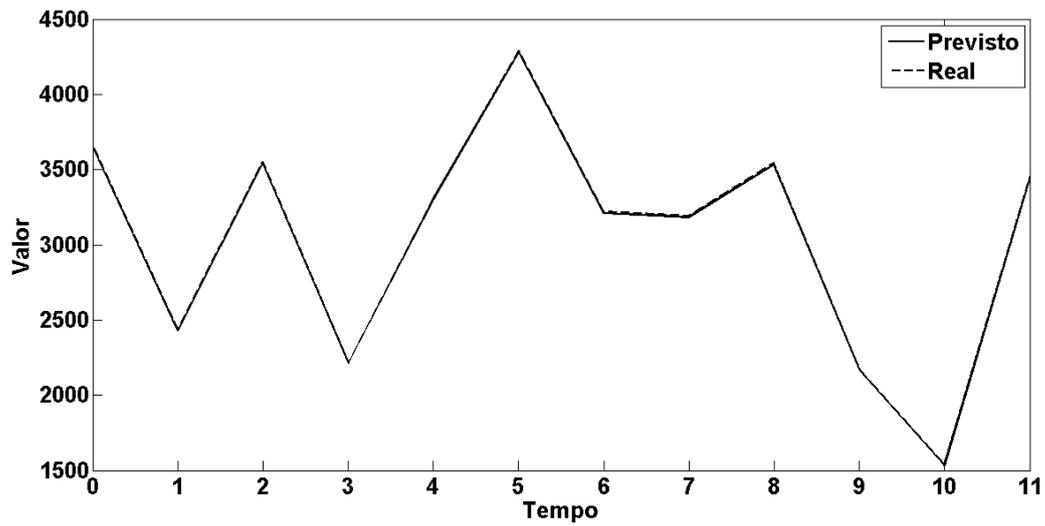


Figura 6.26 Gráfico previsto  $\times$  real para a série Pollution

com outras técnicas de previsão bastante conhecidas tais como SARIMA [2] e SVR [4]. Os resultados apresentados foram medidos em um conjunto de teste composto por 12 padrões. Em outro trabalho recente [19], foi proposto um sistema híbrido inteligente, formado pelo FPSO [21] e modelos MLP e SVR. Os resultados reportados foram similares aos obtidos pela SSVR.

Teste realizados para o mesmo conjunto de teste mostram que, ao utilizar o modelos SVR, a metodologia empregada nesta dissertação obteve resultados semelhantes aos reportados antes. Porém, ao utilizar as redes do tipo MLP, os resultados foram melhorados em 88% para o RMSE e em 87% para o MAPE. Para a obtenção dos referidos valores a base sofreu uma transformação logarítmica, pois pela análise do gráfico média  $\times$  desvio foi possível notar a existência de proporcionalidade entre a média das amostras e seus desvios, como é mostrado na Figura 6.27.

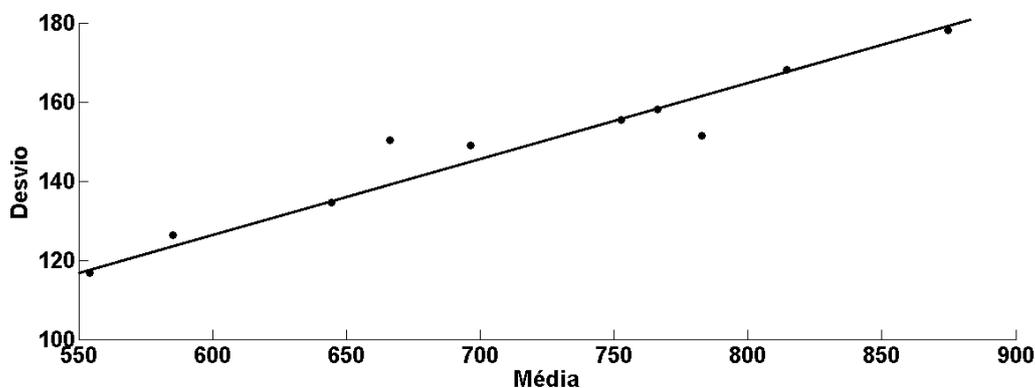


Figura 6.27 Gráfico desvio  $\times$  média na série Paper

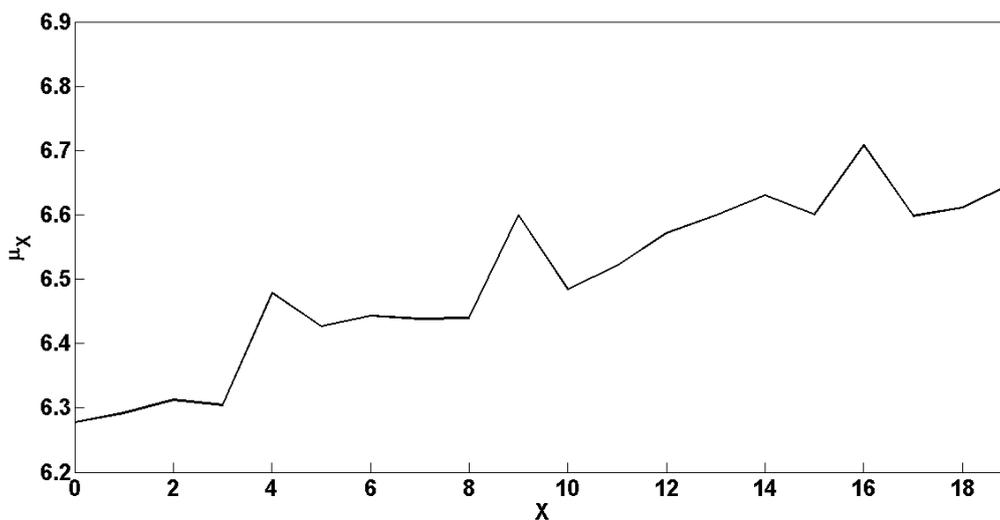
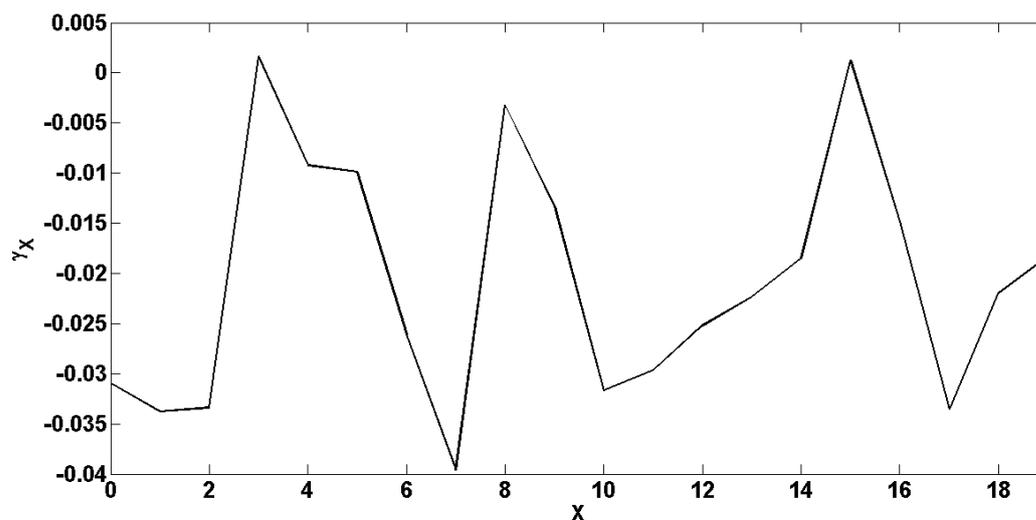


Figura 6.28 Gráfico da média amostral para a série Paper após transformação logarítmica

Após fazer a transformação logarítmica, a base demonstrou um comportamento não estacionário que pode ser confirmado pela análise dos gráficos de médias amostrais e autocovariâncias amostrais na Figura 6.28 e na Figura 6.29, respectivamente. Como a média amostral varia, há indícios de que a série não é estacionária. A variância da média amostral pode ser vista na Tabela 6.11. Também foi executado o teste KPSS e foi rejeitada com 99% de confiança a hipótese de estacionariedade. Devido a isso a série sofreu uma diferenciação, procedimento que foi suficiente para torná-la estacionária como mostrou o teste KPSS após esta diferenciação.



**Figura 6.29** Gráfico da autocovariância amostral para a série Paper após transformação logarítmica

Após diferenciar a série, foi realizada a seleção das características com o FPSO com um enxame de 10 partículas em um espaço de busca formado por 5 características. Os resultados obtidos foram melhores que os reportados em outros trabalhos [19, 25].

Média Amostral	Autocovariância Amostral
0,017	$1,5 \cdot 10^{-4}$

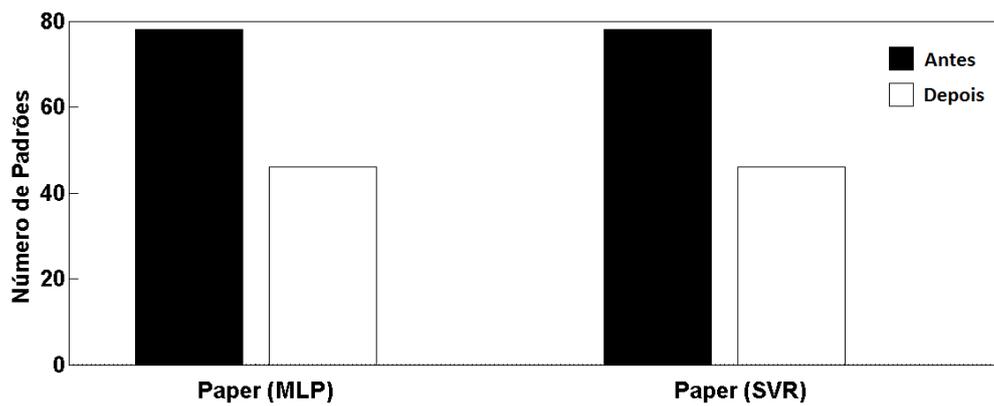
**Tabela 6.11** Variância da média amostral e da autocovariância para a série Paper após transformação logarítmica

Resultados anteriores conseguiram boa modelagem da série em questão, mas realizaram uma busca em espaço de características maior [19]. Já para os resultados produzidos nesta dissertação, o espaço de busca foi 75% menor, fato que contribuiu para a redução do esforço computacional.

A Figura 6.30 mostra que houve uma redução de 42% no tamanho do conjunto de treinamento com ganho de precisão tanto para metodologia com MLP quanto com SVR como modelo de previsão.

**Tabela 6.12** Resultados para a série Paper

Técnica(Parâmetros)	RMSE $\bar{x}(s)$	MAPE $\bar{x}(s)$
SSVR	50,96	0,0547
FSL( $S=20;F=16;\varepsilon = 0,001;C=2$ )	51,03(1,44)	0,047(0,002)
DSL ( $S=20,F=16,\varepsilon = 0,001;C=2$ )	62,80(2,2)	0,0655(0,0034)
DTSL*( $S=10;F=5;\varepsilon = 0,01;C=2$ )	49,43(1,44.10 <sup>-4</sup> )	0,0419(7,0.10 <sup>-8</sup> )
FM( $S=10;F=15;L=0,01;M=0,2;E=400$ )	69,54(1,38)	0,058(0,0019)
DFM( $S=20;F=15;L=0,1;M=0,2;E=400$ )	74,73(7,23)	0,0715(0,0081)
DTFM*( $S=10;F=5;L=0,01;M=0,02$ )	6,0(0,034)	0,0068(3,4.10 <sup>-6</sup> )

**Figura 6.30** Número de Padrões de treinamento antes e depois da seleção na Série Paper

Na Figura 6.31 são mostradas as séries prevista e real no conjunto de teste da série Paper, a metodologia proposta conseguiu modelar o comportamento da série original.

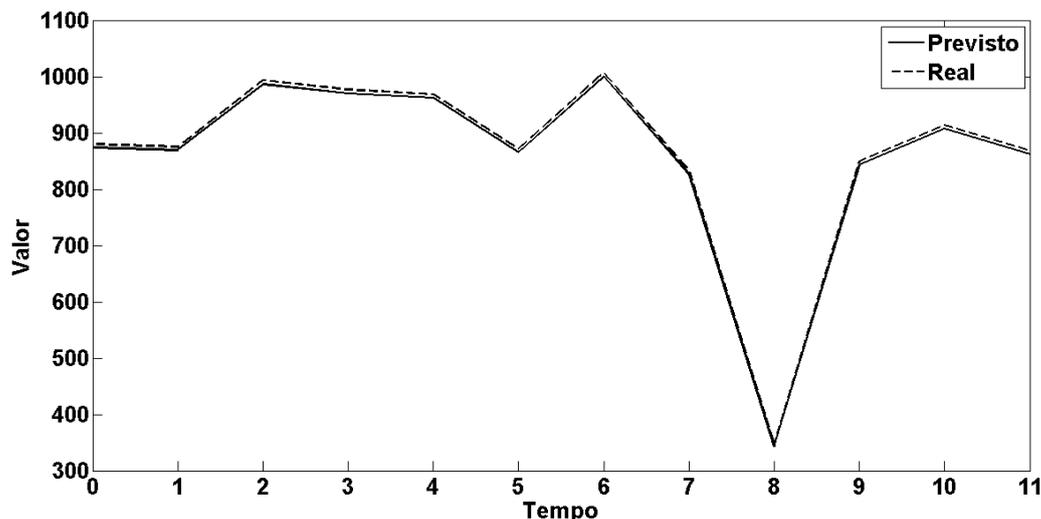


Figura 6.31 Gráfico previsto  $\times$  real para a série Paper

### 6.3.5 Experimentos com a série PETR4

Nos experimentos com a série PETR4 foram usadas as métricas MSE, MAPE e THEIL, pois as três foram utilizadas em outros trabalhos [19, 26] e deseja-se comparar os desempenhos. O conjunto de teste utilizado foi composto por 300 padrões. Ao analisar o gráfico média  $\times$  desvio não foi detectada relação de proporcionalidade entre as médias amostrais e o desvio padrão das amostras, por isso, não foi utilizada a transformação logarítmica. Como pode ser visto na Figura 6.32.

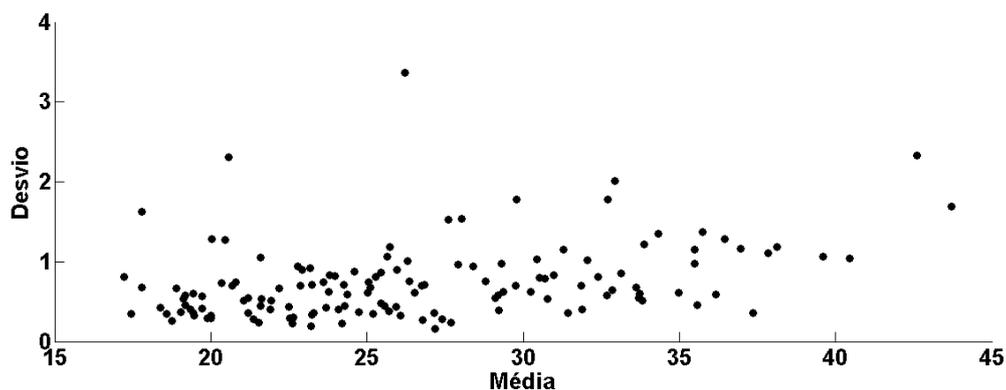
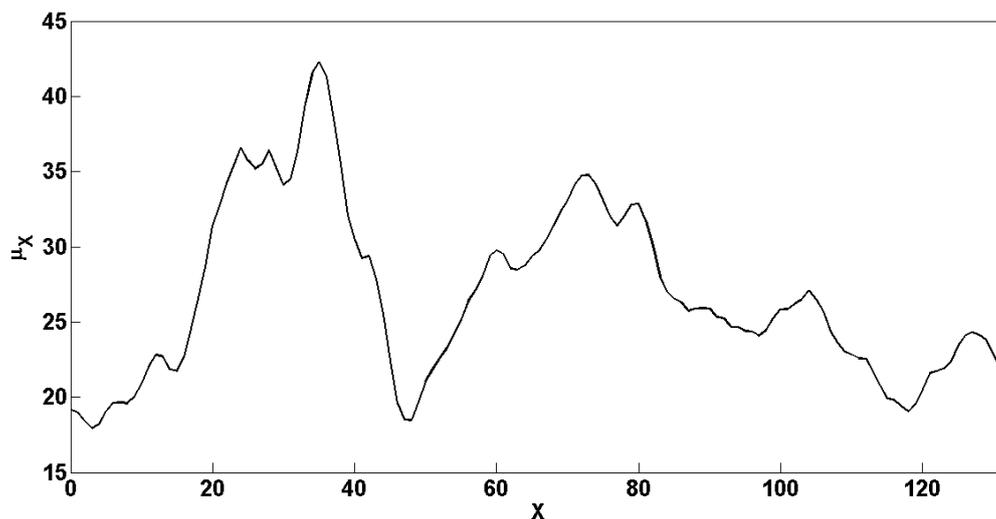


Figura 6.32 Gráfico média  $\times$  desvio para a série PETR4

Porém, ao analisar a Figura 6.33 e a Figura 6.34 pode-se notar que trata-se de uma série não estacionária e a Tabela 6.13 reforça essa hipótese. A execução do teste KPSS

também rejeitou a hipótese de estacionariedade com 99% de confiança. Sendo assim, foi necessário diferenciar uma vez a série PETR4 e isso foi suficiente para torná-la estacionária, fato que também foi verificado pela não rejeição da hipótese de estacionariedade na execução do KPSS após a diferenciação.



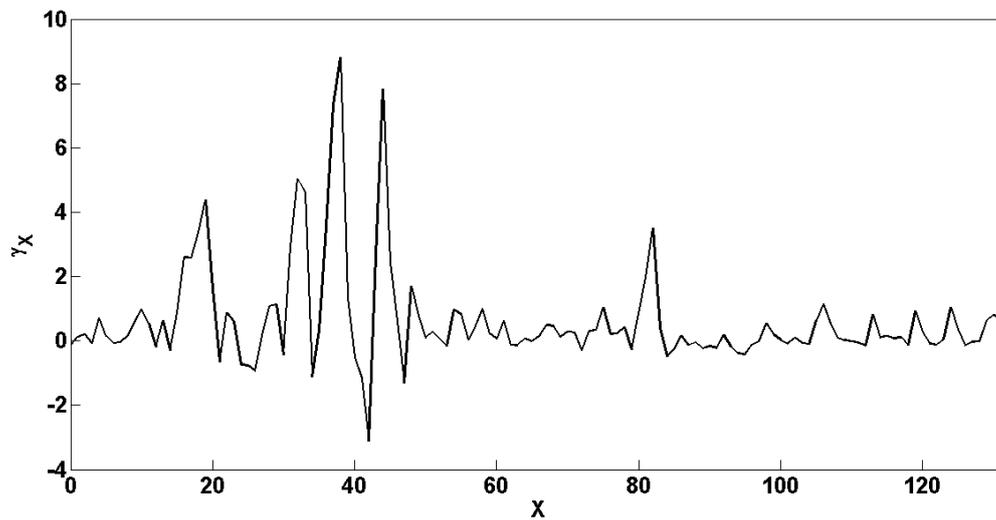
**Figura 6.33** Gráfico da média amostral para a série PETR4

Média Amostral	Autocovariância Amostral
33, 40	2, 49

**Tabela 6.13** Variância da média amostral e da autocovariância para a série PETR4

Na etapa de seleção do conjunto de treinamento, alguns tamanhos de janela foram aplicados para a PETR4, os valores testados podem ser verificados na Tabela 6.4. Após experimentos com cada uma das janelas, o tamanho considerado ideal foi escolhido através da abordagem mostrada na Figura 6.1. Ao utilizar a MLP e a SVR como modelos o conjunto de treinamento teve 699 padrões e 589 padrões respectivamente. Tanto para a MLP quando para a SVR houve redução desse conjunto e melhoria dos desempenhos. Ou seja, foram produzidos resultados melhores consumindo menos recursos computacionais que abordagens anteriores [19, 26]. Além disso o número de características selecionadas proporcionou a construção de modelos mais parcimoniosos.

Para as redes do tipo MLP, apenas 200 épocas foram executadas para o treinamento do modelo e a busca pelos *lags* foi realizada em um espaço de tamanho 10. A abordagem anterior [19] utilizou o dobro de épocas para o treinamento da rede e um espaço de 15 características. Com isso, pode-se concluir claramente a redução do esforço computacional com muito mais precisão nas previsões. Pode-se notar pela Tabela 6.14 os resultados aqui obtidos são melhores que os de outras abordagens [19, 26].

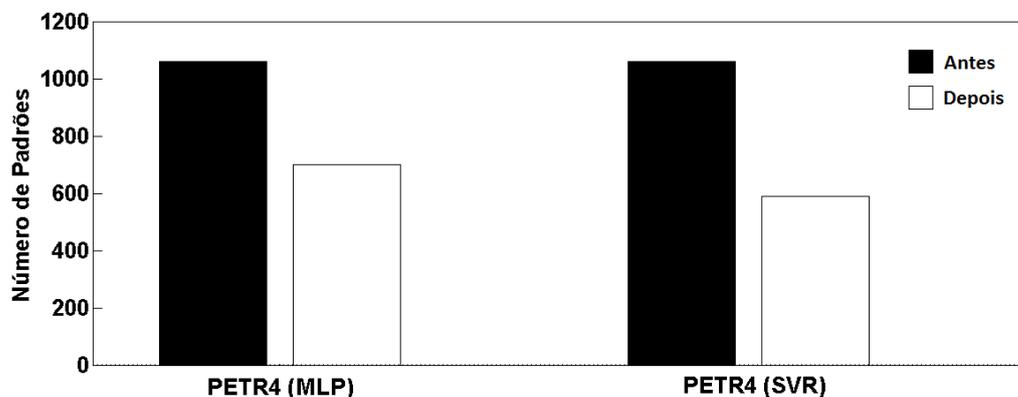


**Figura 6.34** Gráfico da autocovariância amostral para a série PETR4

**Tabela 6.14** Resultados para a série PETR4

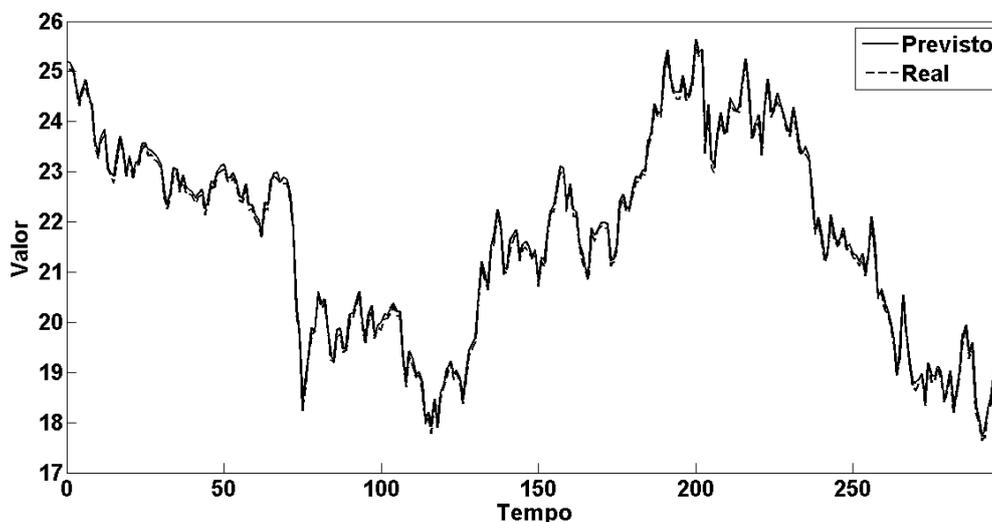
Técnica(Parâmetros)	MSE $\bar{x}(s)$	MAPE $\bar{x}(s)$	THEIL $\bar{x}(s)$
MLP-Committee	0,60(0,011)[26]	0,016(1,6.10 <sup>-4</sup> )	0,27(0,0035)
RBF-Committee	0,61(0,010)[26]	0,016(2,24.10 <sup>-4</sup> )	0,296(0,0038)
FSL( $S=20;F=15;\varepsilon = 0,001;C=1$ )	0,2(0,0026)	0,016(0,00007)	0,9(0,01)
DFSL( $S=20;F=10;\varepsilon = 0,001;\gamma = 0,5;C=1$ )	0,21(0,0156)	0,016(0,0007)	0,9(0,0816)
DTFSR*( $S=20;F=10;\varepsilon = 0,001;\gamma = 0,5;C=1$ )	0,01(0,0035)	0,004(6,6.10 <sup>-4</sup> )	0,05(0,0184)
FM( $S=20;F=10;L=0,01;M=0,05;E=400$ )	0,21(0,0024)	0,017(0,00008)	0,95(0,001)
DFM( $S=20;F=10;L=0,01;M=0,05;E=200$ )	0,19(0,0007)	0,016(0,00004)	0,94(0,005)
DTFM*( $S=20;F=10;L=0,01;M=0,05$ )	1,0.10 <sup>-2</sup> (0,000098)	0,004(0,00002)	0,05(0,0005)

A Figura 6.35 mostra uma comparação entre o número de padrões utilizados antes e depois de fazer a seleção do conjunto de treinamento. Houve melhora de precisão com redução de aproximadamente 45% no número de padrões no conjunto de treinamento.



**Figura 6.35** Número de Padrões de treinamento antes e depois da seleção na Série PETR4

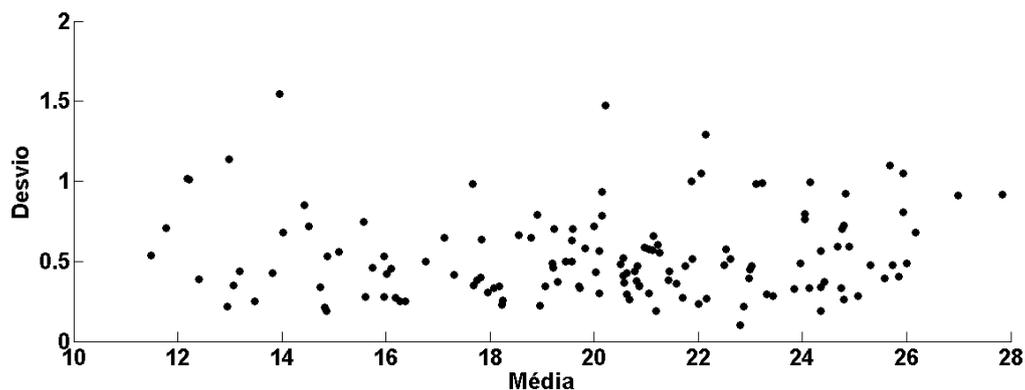
Pode-se verificar o comportamento da previsão em relação ao conjunto de teste da série PETR4 na Figura 6.36. Nota-se claramente a similaridade que existe entre o comportamento da série prevista e da série real.



**Figura 6.36** Gráfico previsto  $\times$  real na série PETR4

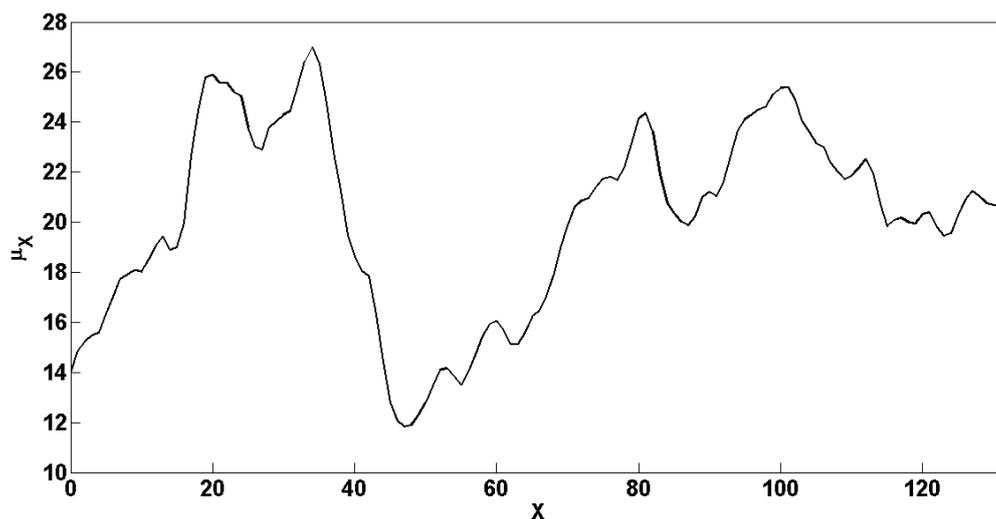
### 6.3.6 Experimentos com a série VALE5

Como de costume, a série VALE5 também foi alvo de previsões em outros trabalhos [19, 26] e por isso foi utilizada neste trabalho. As métricas de avaliação foram as mesmas utilizadas na série PETR4. O conjunto de teste foi composto por 300 padrões.



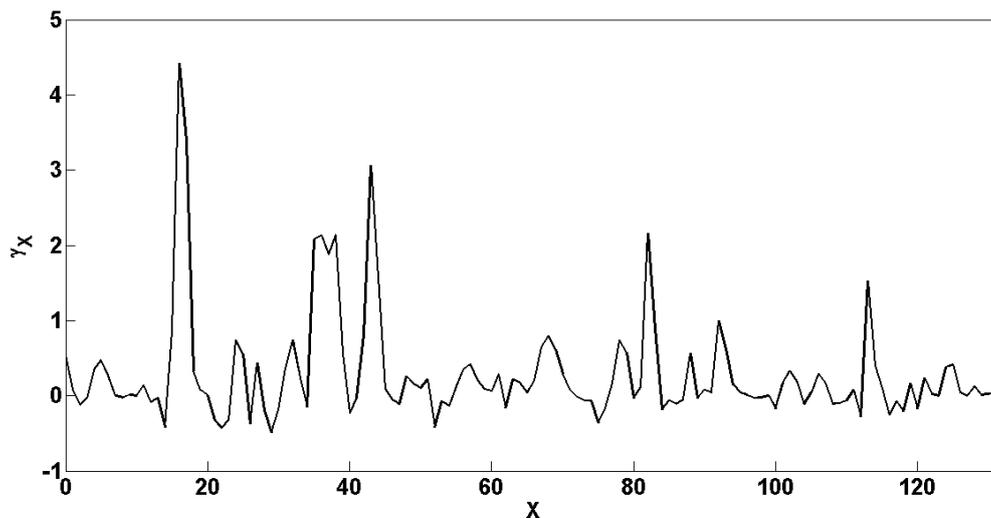
**Figura 6.37** Gráfico média  $\times$  desvio para a série VALE5

A Figura 6.37 mostra que não existe relação proporcional entre a média e o desvio das janelas da série. Por isso, não foi aplicada a transformação logarítmica nesta série. Há indícios de que série transformada não é estacionária, pois como se pode observado a partir da Figura 6.38 que mostra a variação da média amostral. A Figura 6.39 mostra que autocorrelação tem o mesmo comportamento da média amostral. O teste de estacionariedade KPSS rejeitou a hipótese de estacionariedade com 99% de confiança, fato que valida a análise gráfica e é um indício mais forte da não estacionariedade da série. Devido a isso, a série sofreu uma diferenciação. Após esta etapa, o teste KPSS foi executado novamente e não houve rejeição da hipótese de estacionariedade, sendo assim há fortes indícios de que a série passou a ser estacionária após uma diferenciação.



**Figura 6.38** Gráfico da média amostral para a série VALE5

A Tabela 6.15 serve para mostrar através da variância que tanto a média quanto a autocorrelação não são constantes.



**Figura 6.39** Gráfico da autocovariância amostral para a série VALE5

Média Amostral	Autocovariância Amostral
14,87	0,4720

**Tabela 6.15** Variância da média amostral e da autocovariância para a série VALE5

Na seleção do conjunto de treinamento, os tamanhos 699 e 590 foram definidos quando os modelos MLP e SVR foram usados, respectivamente. As características foram selecionadas pelo FPSO com um enxame de 20 partículas em um espaço de busca de tamanho 10, o qual foi menor que o espaço investigado em abordagem anterior [19].

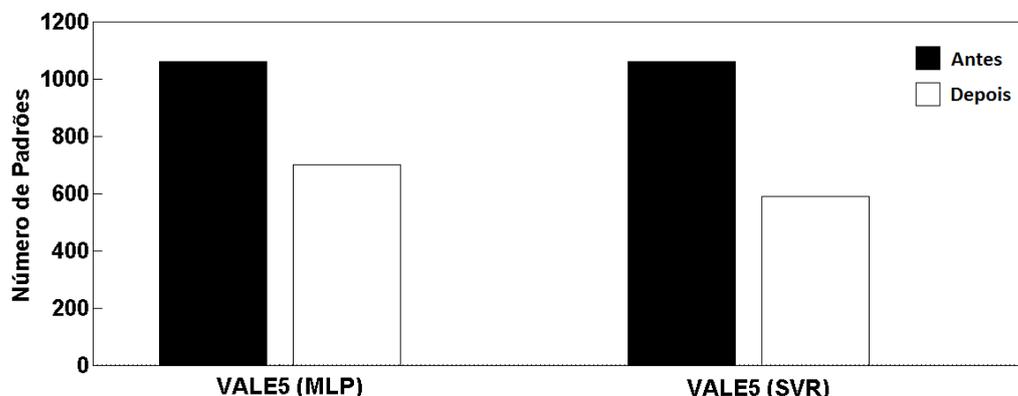
Pode-se observar pela Tabela 6.16 que a metodologia proposta obteve melhores resultados que os reportados em outros trabalhos [26]. Vale salientar que os resultados obtidos utilizaram menos dados para treinamento, fato que contribuiu para uma metodologia que exige menos recursos computacionais.

**Tabela 6.16** Resultados para a série VALE5

Técnica(Parâmetros)	MSE $\bar{x}(s)$	MAPE $\bar{x}(s)$	THEIL $\bar{x}(s)$
MLP-Committe	0,72(0,024)[26]	0,015(2,7.10 <sup>-4</sup> )	0,34(0,012)
RBF-Committe	0,59(0,013)[26]	0,015(2,5.10 <sup>-4</sup> )	0,265(0,007)
FSL( $S=20;F=10;\varepsilon = 0,001;C=2$ )	0,14(0,007)	0,0137(4,72.10 <sup>-4</sup> )	0,90(0,004)
DFSL( $S=20;F=10;\varepsilon = 0,001;C=1$ )	0,1388(0,0035)	0,0135(1,81.10 <sup>-4</sup> )	0,93(0,001)
DTFSL*( $S=15;F=10;\varepsilon = 0,001;C=1$ )	0,0018(7,8.10 <sup>-5</sup> )	0,0019(1,11.10 <sup>-4</sup> )	0,019(0,003)
FM( $S=20;F=10;L=0,01;M=0,05;E=400$ )	0,1310(0,0015)	0,013(1,0.10 <sup>-4</sup> )	0,81(1,2.10 <sup>-3</sup> )
DFM( $S=20;F=10;L=0,01;M=0,05;E=100$ )	0,1405(7,5.10 <sup>-4</sup> )	0,0137(4,8.10 <sup>-5</sup> )	0,93(5,35.10 <sup>-2</sup> )
DTFM*( $S=20;F=10;L=0,01;M=0,05$ )	0,0024(6,9.10 <sup>-5</sup> )	0,0024(3,1.10 <sup>-5</sup> )	0,0187(5,35.10 <sup>-4</sup> )

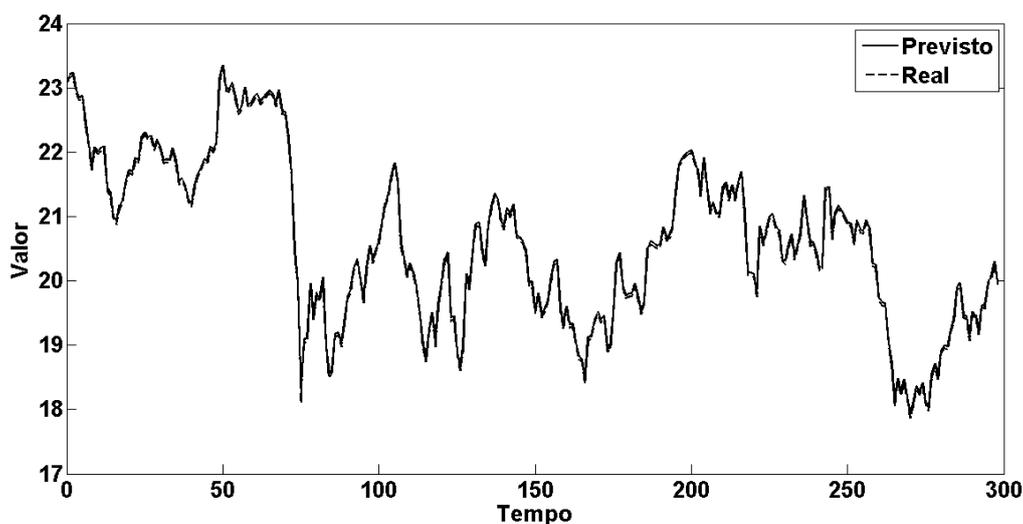
A Figura 6.40 mostra uma comparação entre o número de padrões utilizados antes e

depois de fazer a seleção do conjunto de treinamento. Houve melhora de precisão com redução de aproximadamente 45% no número de padrões no conjunto de treinamento.



**Figura 6.40** Número de Padrões de treinamento antes e depois da seleção na Série VALE5

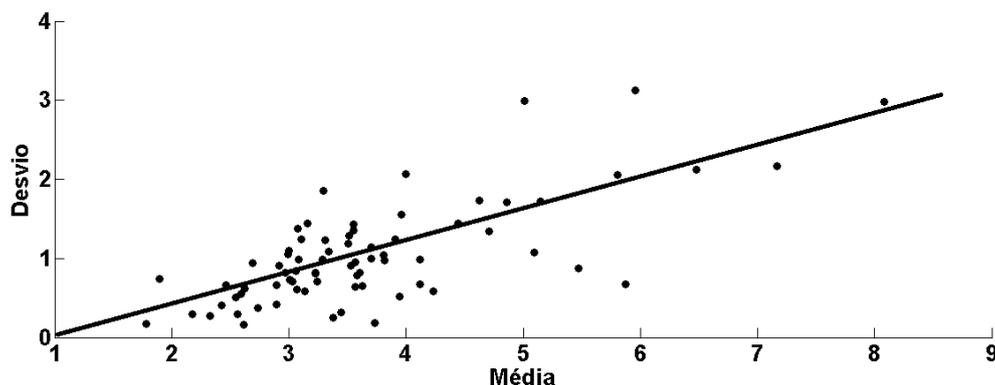
A Figura 6.41 mostra o comportamento da previsão e da série real, ambas no conjunto de teste. Nesta figura é possível notar que a metodologia empregada conseguiu uma boa aproximação da série prevista em relação à série real.



**Figura 6.41** Gráfico previsto  $\times$  real para a série VALE5

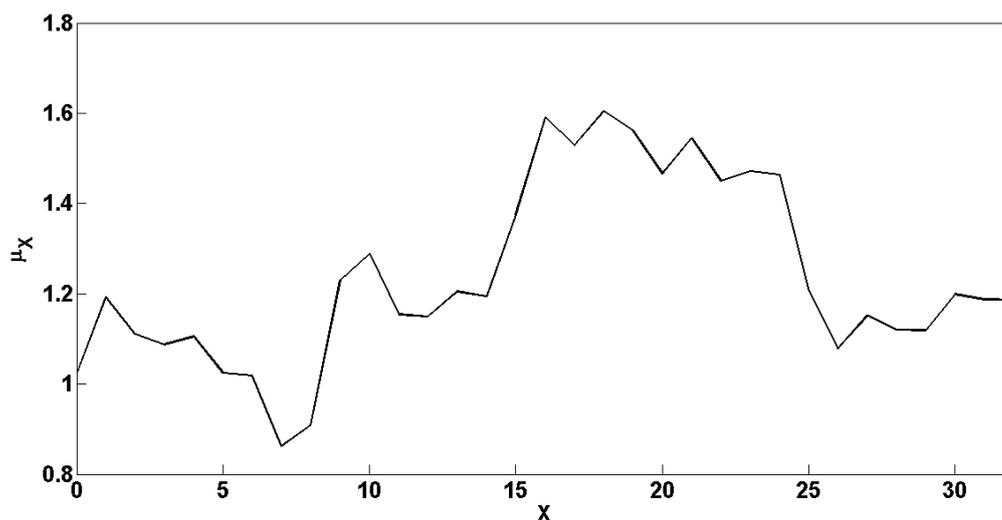
### 6.3.7 Experimentos com a série PCO

Os resultados obtidos com a série PCO são expressos em termos de MAPE tendo em vista que experimentos realizados pela metodologia *Temporal Memory Search + MLP* [18] também utilizaram essa métrica e deseja-se comparar os resultados. O conjunto de teste usado teve 92 observações.



**Figura 6.42** Gráfico média  $\times$  desvio para a série PCO

Seguindo o fluxo mostrado na Figura 6.11, a série necessitou passar pela transformação logarítmica conforme demonstra a Figura 6.42.

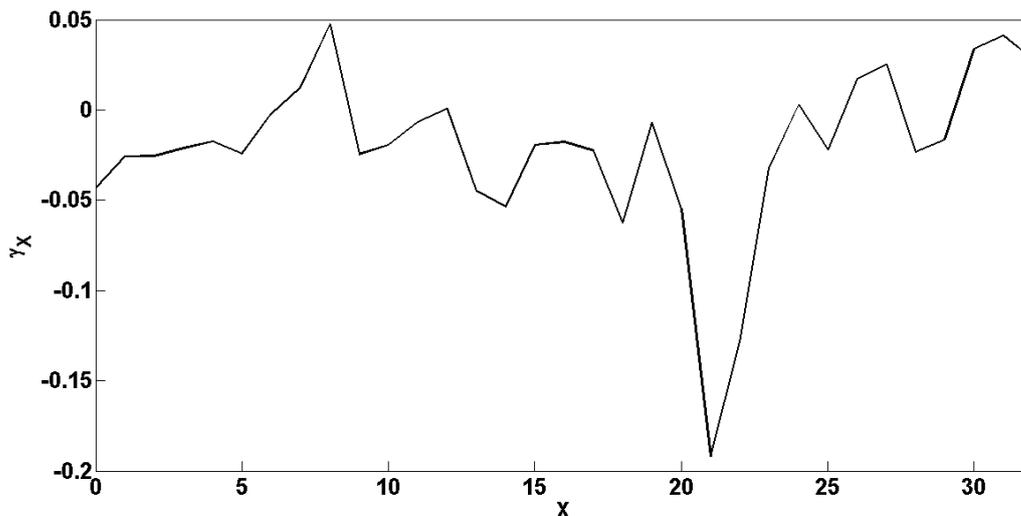


**Figura 6.43** Gráfico da média amostral para a série PCO após a transformação logarítmica

Através dos gráficos presentes na Figura 6.43 e na Figura 6.44 dão indícios de que a série não é estacionária. O teste de estacionariedade KPSS confirmou esse fato ao rejeitar a hipótese de estacionariedade com 99% de confiança. Sendo assim, uma diferenciação foi aplicada e, após isso, um novo teste KPSS não rejeitou a hipótese de estacionariedade, fato que fornece fortes indícios de que a série passou a ser estacionária.

O comportamento não estacionário pode ser comprovado pela Tabela 6.17, a qual mostra que tanto a variância da média amostral quanto da autocorrelação amostral estão distantes de zero, significando que essas medidas não são independentes da amostra.

Após a etapa de tratamento de dados, a seleção do conjunto de treinamento definiu 154 padrões para a rede MLP como modelo e 200 para a SVR como modelo. A partir



**Figura 6.44** Gráfico da autocorrelação amostral para a série PCO após a transformação logarítmica

Média Amostral	Autocovariância Amostral
0,0503	0,0022

**Tabela 6.17** Variância da média amostral e da autocovariância para a série PCO

desses conjuntos houve a seleção de características e se pode notar pela Tabela 6.18 que tanto para a MLP quanto para SVR os resultados foram superiores aos encontrados na literatura [18].

**Tabela 6.18** Resultados para a série PCO

Técnica(Parâmetros)	MAPE $\bar{x}(s)$
TMS + MLP [18]	0,2910(0,06)
FM( $S=20;F=15;L=0,01;M=0,2$ )	0,2783(0,0012)
DFM( $S=10;F=5;L=0,1;M=0,2,E=200$ )	0,26(0,0018)
DTFM*( $S=10;F=5;L=0,01;M=0,02,E=200$ )	0,064(4,2.10 <sup>-17</sup> )
FSR ( $S=20;F=15;\varepsilon = 0,001;C=1$ )	0,2580(0,0043)
DFSR( $S=20;F=10;\varepsilon = 0,001;C=1$ )	0,2718(0,0021)
DTFSR*( $S=20;F=15;\varepsilon = 0,0001;C=1$ )	0,1406(0,013)

Para os experimentos com a série PCO houve uma redução de aproximadamente 38% no número de padrões utilizados antes e depois de fazer a seleção do conjunto de treinamento como pode ser observado na Figura 6.45. Essa redução foi alcançada ao utilizar tanto a rede MLP quanto para a SVR como modelo de previsão.

A Figura 6.46 mostra o gráfico previsto  $\times$  real para a série PCO em um conjunto

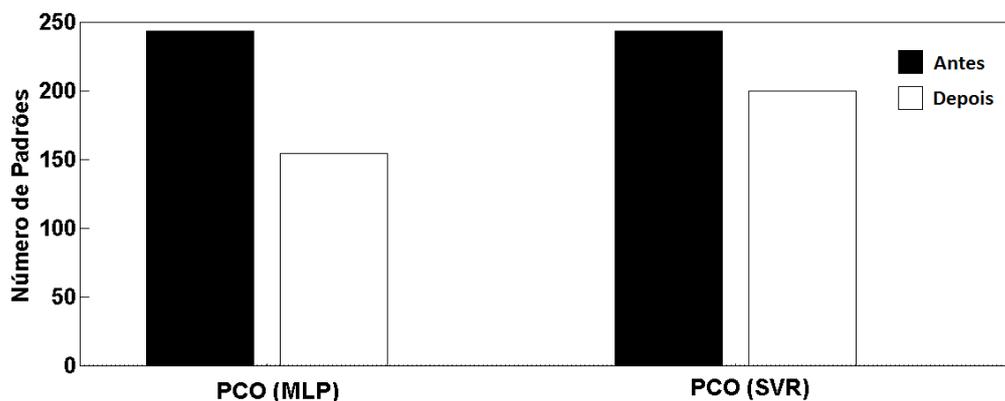


Figura 6.45 Número de Padrões de treinamento antes e depois da seleção na Série PCO

formado por 92 padrões. Nesta figura observa-se a forte similaridade que existe entre a série prevista e a série real.

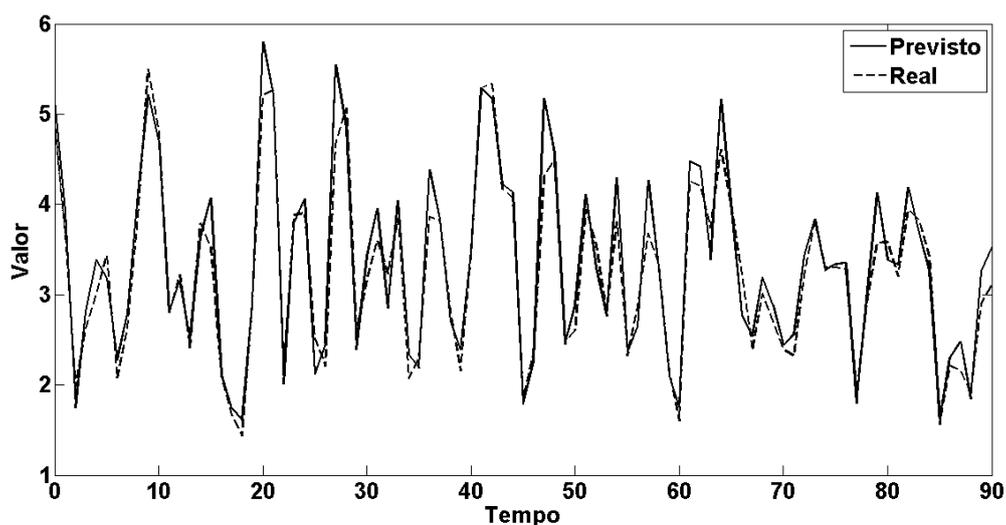
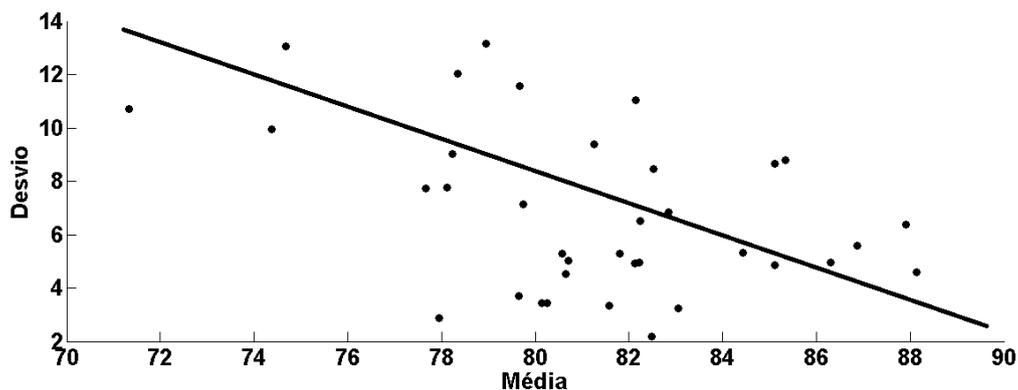


Figura 6.46 Gráfico previsto  $\times$  real para a série Poluição

### 6.3.8 Experimentos com a série Umidade

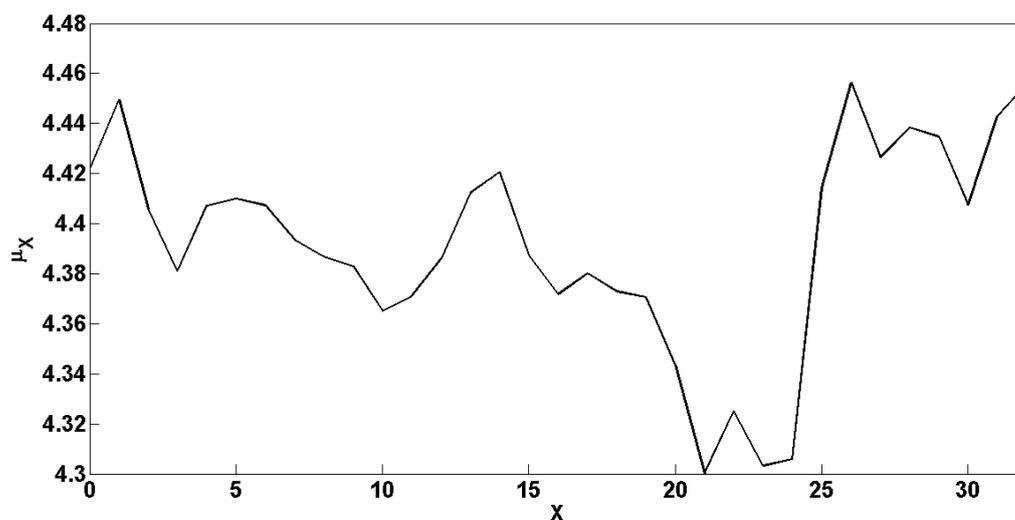
Para a série umidade, também foi utilizada a medida MAPE para comparar com os resultados obtidos pela *Temporal Memory Search + MLP* [18] e um conjunto de teste com 92 padrões. Nesta série, também foi aplicada a transformação logarítmica pois há relação de proporcionalidade entre a média das janelas e o desvio padrão das mesmas, conforme mostra a Figura 6.47.

O teste KPSS rejeitou a hipótese de estacionariedade com 99% de confiança, com isso tem um indício forte de que a série não é estacionária. Além disso, a média amostral,



**Figura 6.47** Gráfico média  $\times$  desvio para a série Umidade

Figura 6.48, possui um valor que varia de acordo com a amostra selecionada. Já a autocorrelação amostral, Figura 6.49 teve variação próxima de zero. A Tabela 6.19 mostra a variância de cada uma das medidas mencionadas. Devido aos indícios de não estacionariedade, foi necessária a aplicação de uma diferença na série. Após essa diferenciação um novo teste KPSS foi executado e não houve rejeição da hipótese de estacionariedade da série.

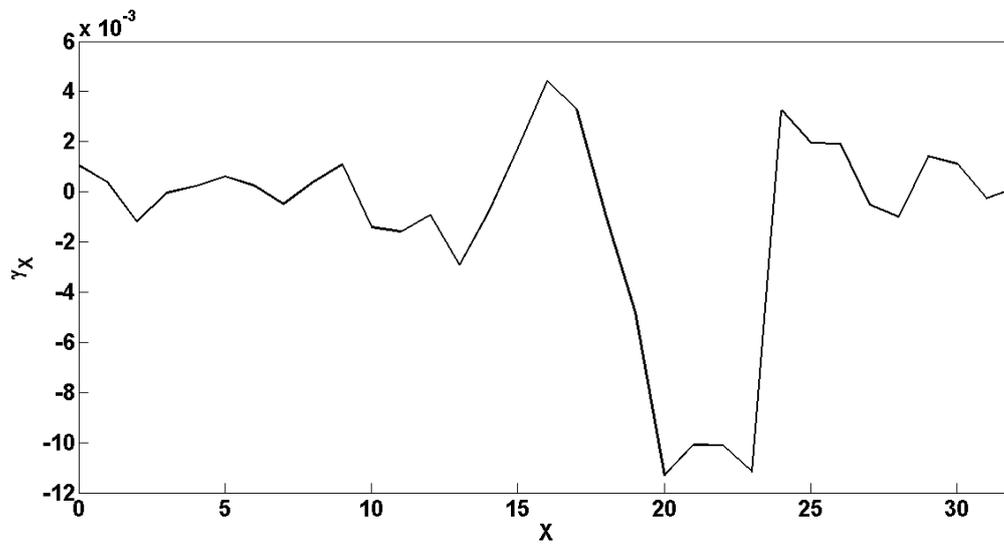


**Figura 6.48** Gráfico da média amostral para série Umidade

Na seleção do conjunto de treinamento foram definidos 221 padrões para treinar a MLP, já com a SVR esse conjunto foi composto por 201 padrões.

A Tabela 6.20 mostra que as previsões obtidas neste trabalho foram melhores que as produzidas por outras propostas [18] em termos de MAPE.

A Figura 6.50 mostra uma comparação entre o número de padrões utilizados antes e depois de fazer a seleção do conjunto de treinamento. Houve uma redução de aproxi-



**Figura 6.49** Gráfico da autocorrelação amostral para a série Umidade

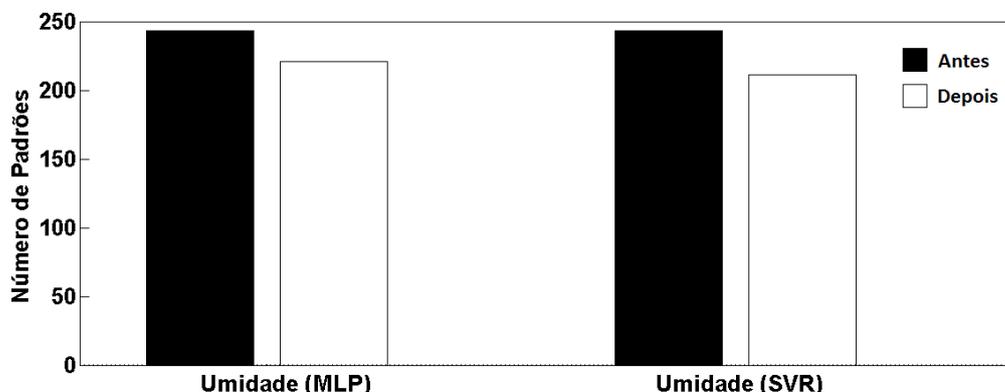
Média Amostral	Autocovariância Amostral
0,0026	$2,8 \cdot 10^{-5}$

**Tabela 6.19** Variância da média amostral e da autocovariância para a série Umidade

**Tabela 6.20** Resultados para a série Umidade

Técnica(Parâmetros)	MAPE $\bar{x}(s)$
TMS+MLP [18]	0,0645(0,021)
FSR( $S=15;F=15;\varepsilon = 0,001;C=1$ )	0,0679( $7,7 \cdot 10^{-4}$ )
DFSR( $S=15;F=15;\varepsilon = 0,001;C=1$ )	0,067( $9,76 \cdot 10^{-4}$ )
DTFSR*( $S=15;F=15;\varepsilon = 0,001;C=1$ )	0,039(0,0022)
FM ( $S=20,F=20;L=0,01;M=0,02$ )	0,0679(0,0011)
DFM( $S=10;F=5;L=0,1;M=0,2$ )	0,0629( $4,27 \cdot 10^{-17}$ )
DTFM*( $S=10,F=5;L=0,01;M=0,02$ )	0,0106(0,0017)

madamente 13% no número de padrões com o ganho de precisão tanto para MLP quanto para SVR como modelo de previsão.



**Figura 6.50** Número de Padrões de treinamento antes e depois da seleção na Série Umidade

A Figura 6.51 mostra o gráfico previsto  $\times$  para a série umidade em um conjunto formado por 92 padrões e nota-se a grande similaridade de comportamento entre os dados reais e a série prevista.

## 6.4 TESTES DE HIPÓTESE

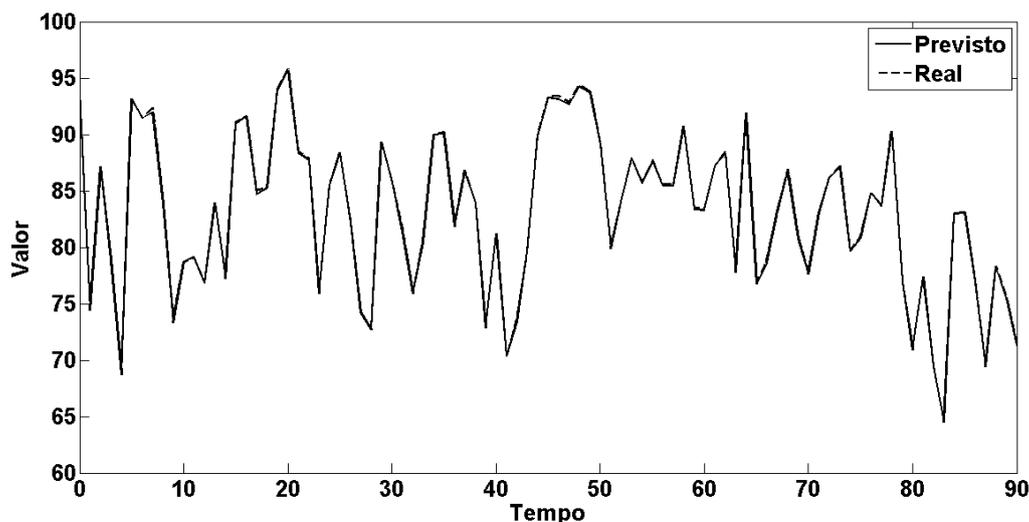
Um teste de hipótese é uma técnica de inferência estatística que é utilizada para verificar a veracidade de uma afirmação. Todo teste é composto por uma hipótese nula  $H_0$  e uma alternativa  $H_1$ . O objetivo é negar  $H_0$  com o maior grau de confiança possível.

Experimentos com diferentes arquiteturas foram realizados neste trabalho, cada arquitetura é enumerada abaixo:

1. A1: Seleção de características com FPSO e um modelo de previsão.
2. A2: Seleção do conjunto de treinamento, seleção de características com FPSO e um modelo de previsão.
3. A3: Tratamento dos dados, seleção do conjunto de treinamento e um modelo de previsão.

A arquitetura A3 obteve os melhores resultados para todos os experimentos, devido à grande diferença de resultados obtida pela mesma não foi necessário aplicar testes de hipótese para compará-la com as outras arquiteturas. Sendo assim, apenas foram realizados testes para comparar os resultados obtidos por A1 e A2 para verificar se a seleção do conjunto de treinamento causa melhoria na previsão dos modelos.

Os testes foram formulados com a hipótese nula  $H_0$  supondo que os resultados obtidos sem seleção do conjunto de treinamento, em termos da média  $\mu_1$ , são iguais aos obtidos com a seleção do conjunto de treinamento, cuja média é  $\mu_2$ . Já a hipótese alternativa,  $H_1$



**Figura 6.51** Gráfico previsto  $\times$  real para a série Umidade

supõe que existe diferença entre as médias dos resultados. O nível de significância de um teste mede a probabilidade de rejeitar a hipótese nula sendo ela verdadeira. Essa medida deve ser a menor possível e é representada pela letra grega  $\alpha$ . O valor de  $\alpha$  adotado neste trabalho foi 5%. Em outras palavras os testes de hipóteses realizados possuem a seguinte formulação:

$$\begin{cases} H_0 : \mu_1 = \mu_2, \\ H_1 : \mu_1 \neq \mu_2, \end{cases} \quad (6.2)$$

O teste de *Wilcoxon* [52] conhecido também como *Mann-Whitney* foi aplicado. Esse tipo de teste é não paramétrico e não faz suposição sobre as distribuições das quais as amostras de resultados foram retiradas. Foram coletadas 30 amostras de experimentos por série temporal e por modelo de previsão empregado. A métrica de avaliação escolhida foi o MAPE por ser uma medida percentual, portanto mais justa na comparação dos resultados.

Pela Tabela 6.21, são mostradas as médias e desvios padrão do MAPE nas amostras coletadas. Dos 13 testes executados, houve rejeição de  $H_0$  em 10 deles. Ou seja, em 10 amostras não existe igualdade de resultados entre as arquiteturas com e sem seleção do conjunto de treinamento. Entre esses, a metodologia com seleção do Treinamento foi melhor em 7. Sendo assim, a conclusão final dos teste de hipótese é a seguinte:

- 3 Testes não rejeitaram a igualdade, apesar das metodologias serem consideradas iguais em termos de MAPE. A redução do conjunto de treinamento alcançou resultados equivalentes com menos padrões de treinamento, portanto com menos esforço computacional.

Série	MAPE ( $\bar{x}(s)$ )	MAPE ( $\bar{x}(s)$ ) <b>Com seleção</b>	Resultado
ESTSP1 MLP	0,0535(2,11.10 <sup>-17</sup> )	0,0551(7,4.10 <sup>-4</sup> )	Não rejeita $H_0$
ESTSP1 SVR	0,0694(5,87.10 <sup>-4</sup> )	0,0569(7,8.10 <sup>-4</sup> )	Rejeita $H_0$
ESTSP2 MLP	0,0677(0,0057)	0,0613(0,0017)	Rejeita $H_0$
Pollution MLP	0,197(0,018)	0,198(0,031)	Rejeita $H_0$
Pollution SVR	0,26(0,025)	0,19(0,0013)	Rejeita $H_0$
PETR4 MLP	0,01663(8,17.10 <sup>-5</sup> )	0,0156(3,6.10 <sup>-5</sup> )	Rejeita $H_0$
PETR4 SVR	0,016(6,9.10 <sup>-5</sup> )	0,016(6,7.10 <sup>-4</sup> )	Não rejeita $H_0$
VALE5 MLP	0,014(0,0001)	0,0137(1,0.10 <sup>-5</sup> )	Rejeita $H_0$
VALE5 SVR	0,0131(1.10 <sup>-4</sup> )	0,0135(2,0.10 <sup>-4</sup> )	Não rejeita $H_0$
Poluição MLP	0,2783(0,0012)	0,0156(2,0.10 <sup>-4</sup> )	Rejeita $H_0$
Poluição SVR	0,0156(0,0001)	0,0161(7,0.10 <sup>-4</sup> )	Rejeita $H_0$
Umidade MLP	0,0679(0,0011)	0,0629(1,0.10 <sup>-5</sup> )	Rejeita $H_0$
Umidade SVR	0,0679(8,0.10 <sup>-4</sup> )	0,0671(0,001)	Rejeita $H_0$

**Tabela 6.21** Resultados dos testes de hipótese

- 10 Testes foram rejeitados, sendo 7 deles superiores em termo de MAPE quando foi utilizada a seleção do treinamento, os outros 3 mostraram leve vantagem para a metodologia sem seleção do conjunto de treino.

Por fim, há um custo adicional em escolher o conjunto de treinamento, porém é válido aplicar a metodologia de seleção do conjunto de treinamento. Pois mesmo nos resultados com desempenho inferior, a inferioridade não foi relevante, sendo esta compensada pelo ganho de tempo na execução das previsões.

## 6.5 COMENTÁRIOS FINAIS

Este capítulo detalhou expôs a origem de cada série temporal utilizada e mostrou as propriedades de cada uma delas. Dissertou sobre a metodologia de previsão empregada neste trabalho e os passos necessários para sua execução. Por fim, mostrou os resultados e os comparou com outros reportados anteriormente.

Testes de hipótese foram desenvolvidos e chegou-se à conclusão que a seleção do conjunto de treinamento é uma etapa válida na previsão de séries temporais, pois além de melhorar a precisão dos modelos reduz o esforço computacional para realizá-la.

# CONCLUSÃO

Este trabalho dissertou sobre algumas pesquisas na área de previsão de séries temporais. Foi discutida a importância de se fazer uma previsão correta. Se feita com precisão, uma previsão pode ser de grande utilidade em decisões que envolvam os valores futuros da série prevista.

Também foram apresentados alguns conceitos importantes da área previsão e mostradas algumas das principais técnicas estatísticas utilizadas para essa finalidade, bem como suas vantagens e desvantagens. Foram revisados também conceitos relativos às redes do tipo MLP e máquinas de vetores de suporte para regressão. Discutiu-se também duas formas de tratamento de dados e as situações nas quais elas devem ser utilizadas.

A seleção correta dos *lags* é uma tarefa vital para uma previsão correta, por isso, foram revisadas algumas formas de realizar essa tarefa. Quatro algoritmos de otimização baseados em enxames de partículas serviram como base para o funcionamento do FPSO, por esse motivo foram explicados brevemente o funcionamento de cada um deles para o entendimento do FPSO, que foi o utilizado na seleção das características para esse trabalho.

Outro ponto que foi discutido se referiu a análise de similaridade entre séries temporais. Pois com base nessa análise é que foi construída a abordagem de seleção do conjunto de treinamento proposta neste trabalho. Duas formas de analisar similaridade foram discutidas. Uma dessas maneiras foi o DTW, porém foi mostrado através de um exemplo que esse algoritmo possui complexidade quadrática. Por esse motivo, foi explicado o funcionamento do *FastDTW* na tarefa de aproximar o DTW com complexidade linear.

A metodologia desenvolvida neste trabalho foi composta por três pontos chave: tratamento dos dados, seleção do conjunto de treinamento e seleção de características. Sendo assim, cada ponto foi discutido em detalhes e mostrado um panorama completo da metodologia de previsão proposta.

Os modelos de previsão utilizados foram SVR e redes do tipo MLP, não houve otimização de seus parâmetros. Os mesmos foram determinados por meio de experimentos.

O desempenho da metodologia de previsão proposta foi medido por quatro métricas bem conhecidas na literatura: MSE, RMSE, MAPE e THEIL. O fato de ter escolhido essas métricas deve-se às mesmas também terem sido utilizadas em outros trabalhos, facilitando a comparação dos resultados.

Experimentos foram realizados com as seguintes arquiteturas: FPSO/ Modelo de Previsão, Seleção do conjunto de treinamento/ FPSO/ Modelo de previsão, por último a metodologia proposta nesta dissertação, Tratamento dos dados/ Seleção do Conjunto de Treinamento/ FPSO/ Modelo de previsão. A arquitetura que apresentou melhores resultados foi a última. Os resultados reportados mostraram também que para algumas

séries apenas a seleção das características foi suficiente para melhorar os resultados reportados por outros trabalhos, mas que através da seleção do conjunto de treinamento e do tratamento dos dados foi possível obter resultados ainda melhores para as oito séries temporais investigadas. A metodologia proposta mostrou-se bastante efetiva ao produzir previsões com alto grau de precisão tanto com a MLP quanto com a SVR e, além disso, reduzindo o esforço computacional em relação a outras metodologias [19, 26].

## 7.1 CONTRIBUIÇÕES

1. Investigação do algoritmo de otimização *Frankenstein's Particle Swarm Optimization* no problema de seleção características para séries temporais - Não há registros de outros trabalhos que utilizaram o FPSO para a seleção de características na previsão de séries temporais [19]. Através da utilização do FPSO foram obtidos resultados superiores para as oito séries temporais testadas em relação a resultados reportados na literatura [25, 16, 26, 18]
2. Proposição de uma abordagem de seleção do conjunto de treinamento baseada na similaridade entre janelas da série temporal alvo da previsão - Algoritmos de otimização geralmente são custosos, uma maneira de reduzir esse custo é selecionar o conjunto de treinamento sobre o qual o algoritmo de otimização vai atuar. A abordagem de seleção empregada é inovadora e foi capaz de atingir o seu objetivo sem causar perda de desempenho nas previsões, ao contrário, a partir dela foi possível alcançar previsões ainda mais corretas. Testes de hipótese confirmaram que a esta etapa foi importante para a melhoria das previsões.
3. Construção de uma metodologia de previsão - A metodologia de previsão construída pode ser completamente automática, tornando mais fácil a atividade de previsão por parte de um usuário que não possua conhecimento suficiente sobre séries temporais, redes neurais e algoritmos de otimização.

Através das contribuições enumeradas foi possível encontrar previsões mais corretas ao mesmo tempo em que foram gerados modelos mais parcimoniosos e com menos dados de treinamento o que sem dúvida contribuiu para a redução do esforço computacional. Os resultados obtidos foram superiores aos reportados na literatura [25, 26, 16, 18]. Parte desses resultados contribuiu para a publicação na conferência (*International Joint Conference on Neural Networks*) realizada em 2011 [19].

## 7.2 LIMITAÇÕES DO TRABALHO

A metodologia proposta trouxe melhorias para a área de previsão de séries temporais, porém ainda apresenta algumas limitações que podem ser eliminadas em trabalhos futuros, as limitações deste trabalhos são expostas abaixo:

- Neste trabalho foram investigados apenas dois modelos de previsão. Vale a pena investigar outros modelos de previsão tais como redes RBF [3] e redes recorrentes [3].

- Outros métodos de otimização podem ser investigados, tais como evolução diferencial [53].
- Os parâmetros dos modelos não foram otimizados, foram definidos através de experimentos. Por isso, é importante que o usuário do sistema conheça os modelos e algoritmos utilizados.
- As análises da necessidade de aplicar a transformação de Box-Cox [31] e de tornar a série estacionária foram feitas através de gráficos, tornando o sistema parcialmente automático.

### 7.3 TRABALHOS FUTUROS

Várias melhorias podem ser realizadas na metodologia, dentre as quais podem ser destacadas:

- A análise da necessidade de tratamento nos dados foi realizada com base em gráficos, mas esse processo pode ser completamente automatizado. Essa é uma contribuição futura que ajudaria na automatização do processo de previsão.
- Utilizar um mecanismo que otimize a escolha do tamanho da janela utilizado pelo *FastDTW* na busca por um subconjunto de treinamento.
- Investigar um mecanismo que otimize o número inicial de *lags* candidatos para que o FPSO encontre a partir desse o conjunto de características que otimize as previsões.
- Propôr uma forma mais eficiente de configuração dos modelos de previsão, atualmente a configuração é feita através de experimentos com várias configurações. Uma contribuição interessante seria automatizar esse processo.
- Estender a técnica de seleção do conjunto de treinamento para lidar com séries temporais muito maiores que as utilizadas neste trabalho.
- Testar a metodologia em um número maior de séries temporais.
- Propôr uma ferramenta de previsão de séries temporais para ser utilizadas em situações reais dando suporte às decisões.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] P. J. Brockwell and R. A. Davis. *Introduction to Time Series and Forecasting*. Springer Verlag, 1996.
- [2] G.E.P. Box and G.M. Jenkins. *Time Series Analysis: Forecasting and Control*. Prentice Hall, 1994.
- [3] S. Haykin. *Neural Networks: a Comprehensive Foundation*. Prentice Hall, 2 edition, 1999.
- [4] A.J. Smola and B. Schölkopf. A tutorial on support vector regression. Technical report, University of London, 1998.
- [5] A. P. Braga, A. C. P. L. F. Carvalho, and Ludermir T. B. *Redes Neurais Artificiais - Teoria e aplicações*. LTC, 2007.
- [6] Kourentzes N. and Crone S. F. Frequency independent automatic input variable selection for neural networks for forecasting. *World Congress on Computational Intelligence*, pages 1923–1936, 2010.
- [7] G.Q. Zhang, B.E. Patuwo, and M.Y. Hu. Forecasting with artificial neural networks: the state of the art. *International Journal of Forecasting*, pages 35–62, 1998.
- [8] Paulo S. G. de M. Neto, G. G. Petry, L. J. A. Rodrigues, and T. A. E. Ferreira. Combining artificial neural network and particle swarm system for time series forecasting. *International Joint Conference on Neural Networks*, pages 2230–2237, 2009.
- [9] Juan Peralta, Xiaodong Li, German Gutierrez, and Araceli Sanchis. Time series forecasting by evolving artificial neural networks using genetic algorithms and differential evolution. *World Congress on Computational Intelligence*, 2010.
- [10] R. R. Andrawis, A. F. Atiya, and H. El-Shishiny. Forecast combinations of computational intelligence and linear models for the nn5 time series forecasting competition. *International Journal of Forecasting*, pages 672–688, 2011.
- [11] Christiane Lemke and B. Gabrys. Meta-learning for time series forecasting and forecast combination. *Neurocomputing 73*, page 20062016, 2010.
- [12] S. B. Taieb, G. Bontempi, A. F. Atiya, and A. Sorjamaa. A review and comparison of strategies for multi-step ahead time series forecasting based on the nn5 forecasting competition. *Expert Systems with application 40*, pages 7067–7083, 2012.

- [13] B. C. R. Prudêncio and T. B. Ludermir. Forecast combinations of computational intelligence and linear models for the nn5 time series forecasting competition. *Neurocomputing* 61, pages 121–137, 2004.
- [14] G. P. Zhang. An investigation of neural networks for linear time-series forecasting. *Computers & Operations Research* 28, page 11831202, 2001.
- [15] G. P. Zhang, B. E. Patuwo, and M. Y. Hu. A simulation study of artificial neural networks for nonlinear time-series forecasting. *Computers & Operations Research* 28, page 381396, 2001.
- [16] N. Kourentzes and S.F. Crone. Feature selection for time series prediction - a combined filter and wrapper approach for neural networks. *Neurocomputing* 73, pages 1923–1936, 2010.
- [17] Ron Kohavi and George H. John. Wrappers for feature subset selection. *Artificial Intelligence* 97, pages 273–324, 1997.
- [18] I. Valença, T. Ludermir, and M. Valença. Hybrid systems to select variables for time series forecasting using mlp and search algorithms. *Brazilian Symposium on Neural Networks*, 2010.
- [19] G.H.T. Ribeiro, P. S. G Mattos, G.D.C. Cavalcanti, and Ing Ren Tsang. Lag selection for time series forecasting using particle swarm optimization. *International Joint Conference on Neural Networks*, pages 2437–2444, 2011.
- [20] G.K. Jha, P. Thulasiraman, and Thulasiram R.K. Pso based neural network for time series forecasting. *International Joint Conference on Neural Networks*, pages 1422–1427, 2009.
- [21] M.A. Montes de Oca, T. Stutzle, M. Birattari, and M. Dorigo. Frankenstein’s pso: A composite particle swarm optimization algorithm. *IEEE Transactions on Evolutionary Computation*, pages 1120–1132, 2009.
- [22] A. Chotirat and E. K. Ratanamahatana. Making time-series classification more accurate using learned constraints. Technical report, University of California - Riverside.
- [23] Young-Seon Jeong, K. Jeong Myong, and A. O. Olufemi. Weighted dynamic time warping for time series classification. *Pattern Recognition*, pages 2231–2240, 2011.
- [24] Stan Salvador and Philip Chan. Toward accurate dynamic time warping in linear time and space. *Intell. Data Anal.*, 2007.
- [25] Ping-Feng Pai, Kuo-Ping Lin, Chi-Shen Lin, and Ping-Teng Chang. Time series forecasting by a seasonal support vector regression model. *Expert Systems with Applications* 37, pages 4261–4265, 2010.

- [26] M. C. A. Neto, G. H. T. Ribeiro, V. M. O. Alves, G. D. C. Cavalcanti, and Ing Ren Tsang. Improving financial time series prediction using exogenous series and neural networks committees. *International Joint Conference on Neural Networks*, pages 2345–2352, 2010.
- [27] P. A. Moretin and C. M. C. Toloí. *Análise de Séries Temporais*. Editora Blucher, 2006.
- [28] G.P Zhang. Avoiding pitfalls in neural network research. *IEEE Systems, Man, and Cybernetics Society*, pages 3–16, 2007.
- [29] Zhang G. P. Time series forecasting using a hybrid arima and neural network model. *Neurocomputing 50*, page 159175, 2001.
- [30] V. N. Vapnik. *The nature of statistical learning theory*. Springer-Verlag New York, Inc., 1995.
- [31] R. M. Sakia. The box-cox transformation technique: a review. Technical report, Sokoine University of Agriculture, 1992.
- [32] G. E. P. BOX and D. R. COX. An analysis of tranformations. *Journal of the Royal Statistical Society*, 1964.
- [33] S. F. da Silva. *Seleção de características por meio de algoritmos genéticos para aprimoramento de rankings e de modelos de classificação*. PhD thesis, USP - São Carlos, 2011.
- [34] A. P. Engelbrecht. *Fundamentals of Computational swarm intelligence*. Wiley, 2005.
- [35] J. Kennedy and R. Eberhart. Particle swarm optimization. *IEEE International Conference on Neural Networks*, pages 1942–1948, 1995.
- [36] R. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pages 39–43, 1995.
- [37] M. Clerc and J. Kennedy. The particle swarm - explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation 6*, pages 58–73, 2002.
- [38] Y. Shi and R. Eberhart. A modified particle swarm optimizer. *IEEE International Conference on Evolutionary Computation*, pages 69–73, 1998.
- [39] R. Mendes, J. Kennedy, and J. Neves. The fully informed particle swarm: Simpler, maybe better. *IEEE Transactions on Evolutionary Computation 8*, pages 204–210, 2004.
- [40] Y. Shi and R.C. Eberhart. Empirical study of particle swarm optimization. *Evolutionary Computation*, 1999.

- [41] S. Janson and M. Middendorf. A hierarchical particle swarm optimizer and its adaptive variant. *IEEE Transactions on Systems, Man, and Cybernetics* 35, pages 1272–1282, 2005.
- [42] R. Bellman. *Dynamic Programming*. Princeton University Press, 1957.
- [43] Java. <http://www.java.com>, 2012. Online; acessado em 29-Julho-2012.
- [44] Matlab. <http://www.mathworks.com/products/matlab>, 2012. Online; acessado em 29-Julho-2012.
- [45] Postgres. <http://www.postgresql.org.br>, 2012. Online; acessado em 29-Julho-2012.
- [46] Weka. <http://www.cs.waikato.ac.nz/ml/weka>, 2012. Online; acessado em 29-Julho-2012.
- [47] R Project. <http://www.r-project.org/>, 2012. Online; acessado em 29-Julho-2012.
- [48] ESTSP Datasets. <http://research.ics.aalto.fi/eiml/datasets.shtml>, 2012. Online; acessado em 29-Julho-2012.
- [49] DATAMARKET. <http://datamarket.com>, 2012. Online; acessado em 29-Julho-2012.
- [50] Yahoo Finanças. <http://br.financas.yahoo.com>, 2012. Online; acessado em 29-Julho-2012.
- [51] Pedro Augusto Morettin. <http://www.ime.usp.br/~pam/st.html>, 2012. Online; acessado em 29-Julho-2012.
- [52] W. de O. Bussab and Morettin P. A. *Estatística Básica*. Editora Saraiva, 2010.
- [53] K. Price, R. M. Storn, and J. A. Lampinen. *Differential Evolution - A practical Approach to Global Optimization*. Springer, 2005.
- [54] J. G. Gooijer and R. J. Hyndman. 25 years of time series forecasting. *International Journal of Forecasting*, 2006.
- [55] Site de petrobrás. <http://www.petrobras.com.br>. Último acesso em 12/07/2012.
- [56] D. Basak, S. Pal, and D. C. Patranabis. Support vector regression. *Neural Information Processing - Letters and Reviews* 10, pages 203–224, 2007.