

**UNIVERSIDADE FEDERAL DE MINAS GERAIS  
PROGRAMA DE PÓS-GRADUAÇÃO EM SANEAMENTO,  
MEIO AMBIENTE E RECURSOS HÍDRICOS**

**Gustavo de Souza Groppo**

**PREVISÃO DE DEMANDA DE CURTO PRAZO EM SISTEMAS DE  
ABASTECIMENTO DE ÁGUA EMPREGANDO MODELOS ESTATÍSTICOS, DE  
APRENDIZADO DE MÁQUINA E DE VARREDURA**

**Belo Horizonte**

**2019**

**Gustavo de Souza Groppo**

**PREVISÃO DE DEMANDA DE CURTO PRAZO EM SISTEMAS DE  
ABASTECIMENTO DE ÁGUA EMPREGANDO MODELOS ESTATÍSTICOS, DE  
APRENDIZADO DE MÁQUINA E DE VARREDURA**

Tese apresentada ao Programa de Pós-graduação em Saneamento, Meio Ambiente e Recursos Hídricos da Universidade Federal de Minas Gerais, como requisito parcial à obtenção do título de Doutor em Saneamento, Meio Ambiente e Recursos Hídricos.

Área de concentração: Saneamento

Linha de pesquisa: Sistemas de Recursos Hídricos

Orientador: Marcelo Libânio

Coorientador: Marcelo Azevedo Costa

Belo Horizonte

Escola de Engenharia da UFMG

2019

G876p

Groppo, Gustavo de Souza.

Previsão de demanda de curto prazo em sistemas de abastecimento de água empregando modelos estatísticos, de aprendizado de máquina e de varredura [recurso eletrônico] / Gustavo de Souza Groppo.- 2019.

1 recurso online (xvii, 230 f. : il., color.) : pdf.

Orientador: Marcelo Libânio.

Coorientador: Marcelo Azevedo Costa.

Tese (doutorado) – Universidade Federal de Minas Gerais, Escola de Engenharia.

Anexos: f. 192-230.

Bibliografia: f. 166-190.

Exigências do sistema: Adobe Acrobat Reader.

1. Engenharia sanitária- Teses. 2. Saneamento - Teses.  
3. Abastecimento de água - Teses. 4. Aprendizado do computador - Teses.  
5. Métodos estatísticos - Teses. I. Libânio, Marcelo, 1960-. II. Costa, Marcelo Azevedo. III. Universidade Federal de Minas Gerais. Escola de Engenharia. IV. Título.

CDU: 628(043)



UNIVERSIDADE FEDERAL DE MINAS GERAIS  
Escola de Engenharia  
Programa de Pós-Graduação em Saneamento, Meio Ambiente e Recursos Hídricos  
Avenida Antônio Carlos, 6627 - 4º andar - 31270-901 - Belo Horizonte - BRASIL  
Telefax: 55 (31) 3409-1882 - [posgrad@desa.ufmg.br](mailto:posgrad@desa.ufmg.br)  
<http://www.smarh.eng.ufmg.br>

FOLHA DE APROVAÇÃO

"PREVISÃO DE DEMANDA DE CURTO PRAZO EM SISTEMAS DE ABASTECIMENTO DE ÁGUA EMPREGANDO MODELOS ESTATÍSTICOS, DE APRENDIZADO DE MÁQUINA E DE VARREDURA"

GUSTAVO DE SOUZA GROPPPO

Tese defendida e aprovada pela banca examinadora constituída pelos Senhores:

Prof. MARCELO LIBÂNIO

Prof. MARCELO AZEVEDO COSTA

Prof. VÊBER AFONSO FIGUEIREDO COSTA

Prof. FREDERICO KEIZO ODAN

Prof. GLÁSTON JULIANO PRATES MOREIRA

Prof. OLAVO CORREA PEDROLLO

Aprovada pelo Colegiado do PG SMARH

Versão Final aprovada por

Prof. Taciana Toledo de Almeida Albuquerque  
Subcoordenadora

Prof. Marcelo Libânio  
Orientador

Belo Horizonte, 25 de novembro de 2019.

## AGRADECIMENTOS

A Deus pela vida, presença e amor que me fortalecem e proporcionam fé. Graças a Ele venço mais essa etapa. A caminhada foi muito árdua, mas em cada derrota tenho certeza que uma vitória virá em seguida, pois Ele está sempre comigo.

A minha mulher Roberta Pádua Moares, pelo apoio incondicional nesse momento tão importante e difícil da minha vida. Como é bom ter você ao meu lado e, principalmente, saber que você estará nas horas alegres e tristes. Essa vitória também é sua! Te amo!

Aos meus pais Ernesto Fábio (*in memoriam*) e Neuza, minha irmã Giselle, aos meus filhos Gabriel, Davi e a minha enteada Maria Laura.

Ao professor Dr. Marcelo Libânio, pela confiança depositada em mim. Agradeço por ter aceitado me orientar mesmo sem nunca ter tido contato comigo e nem me conhecer.

Ao professor Dr. Marcelo Azevedo Costa, pela coorientação, pelo grande apoio na parte de modelagem e na definição do método proposto empregado no presente estudo. Pelos ensinamentos, pelo apoio operacional e pela paciência durante todo o processo desta pesquisa. Serei eternamente grato.

A UFMG, principalmente ao programa SMARH, por me conceder a oportunidade de realizar este doutorado, toda a equipe de professores que de alguma forma me apoiaram durante as disciplinas e toda a equipe administrativa do departamento.

Ao meu grande amigo Valter de Souza Lucas Júnior, pelo incentivo e apoio antes e após a entrada no doutorado. Ao amigo Leandro Brioschi Mineti, grande cientista de dados, pelo apoio na modelagem e pela disponibilidade em transmitir o seu conhecimento.

Aos colegas do SMARH Raphael Pedrollo de Paes, Adrielly Fonseca Fialho Ferreira, Ana Flávia Souza Foureaux, Marco Tulio da Silva Faria e a Janaína Antonino Pinto. Foi um prazer conviver com vocês ao longo desta empreitada. Aos colegas de disciplina, do departamento de engenharia elétrica, Caio Silva e Phillippe Leroy.

A todos os não aqui citados, mas não menos importantes, que contribuíram de alguma forma para a conclusão dessa etapa. O meu grande obrigado!

*“Eu fiz em casa, mas sei que não é minha”.*

Egberto Gismonti

## RESUMO

Na operação de Sistemas de Abastecimento de Água (SAA), um dos fatores críticos do abastecimento é a manutenção do equilíbrio entre a oferta e a demanda de água aos usuários. A manutenção deste equilíbrio é feita através de ações operacionais sendo a previsão de demanda de curto prazo um fator de grande importância para esse gerenciamento.

Ao longo dos anos, inúmeros métodos foram desenvolvidos e testados. Os modelos lineares têm sido amplamente utilizados, porque eles são fáceis de desenvolver e implementar, além de simples de entender e interpretar. No entanto, os dados de demanda de água têm diferentes graus de não-linearidade, que não podem ser adequadamente tratados pelos modelos lineares. A revisão de literatura destaca que os modelos mais bem-sucedidos são baseados em abordagens de *soft computing*, como redes neurais, sistemas *fuzzy*, computação evolucionária, máquinas de vetores de suporte além dos modelos híbridos.

Neste trabalho pesquisamos a viabilidade da utilização do método de varredura temporal *Dynamic Time Scan Forecasting* (DTSF) para prever a demanda de curto prazo (horário) em sistemas de abastecimento de água por meio da comparação, em termos de eficiência e do custo computacional (CC), de diversas alternativas univariadas conhecidas. Os métodos estatísticos e neuronais empregados foram: Box-Jenkins (SARIMA), alisamento exponencial (ETS), modelo híbrido usando o filtro *Seasonal and Trend decomposition using Loess* (STL) e o ETS (STL-ETS), modelo *Trigonometric Box-Cox transformation, ARMA erros, Trend and Seasonal components* (TBATS), além dos modelos neuronais híbridos rede neural autorregressiva não-linear com *bootstrap* (BNNAR), *Extreme Learning Machine* (ELM) e o *Naive Bayes* (NB).

Os dados empregados no presente estudo referem-se a uma zona de abastecimento (ZA) localizada na região centro sul de uma capital do Sudeste brasileiro e atende uma população de aproximadamente 230 mil pessoas, podendo ser comparada com uma cidade de médio porte.

Objetivando avaliar os métodos empregaram-se as seguintes métricas: raiz do erro quadrático médio (RMSE), erro absoluto médio (MAE), erro percentual absoluto médio (MAPE), *Model Fitting* (MF), além do CC. Os resultados obtidos demonstram que tanto no verão quanto no outono não houve um modelo que superasse os outros em termos de precisão de previsão. Todavia, para os meses de junho a novembro (inverno e primavera), o método ELM apresentou

os melhores resultados. Os dois métodos neuronais apresentaram as melhores acurácias, contudo, foram os que apresentaram os maiores CC ao empregar a validação cruzada de origem deslizando com recalibração.

Em relação à estatística CC, o algoritmo DTSF foi extremamente rápido quando comparado aos outros métodos estudados para todos os meses do ano, usando uma fração do tempo. Os resultados obtidos demonstraram que o método proposto fornece valores de previsão similares ou melhorados em comparação com métodos de *soft computing* e estatísticos empregados, mas usando uma fração do tempo de computação.

A grande vantagem deste método, dado que é um método orientado a dados, é que quanto maior a quantidade de dados empregada melhor será a sua capacidade de generalização e, em um ambiente conhecido como a era do *big data*, este obterá vantagem frente aos outros métodos.

**PALAVRAS-CHAVE:** *Dynamic Time Scan Forecasting (DTSF); previsão demanda de água; métodos estatísticos; machine learning; soft computing; custo computacional.*

## ABSTRACT

In the operation of Water Supply Systems (SAA), one of the critical factors of supply is maintaining the balance between water supply and demand to users. The maintenance of this balance is made through operational actions and the forecast of short term demand is a major factor for this management.

Over the years, numerous methods have been developed and tested. Linear models have been widely used because they are easy to develop and implement, and simple to understand and interpret. However, water demand data have different degrees of nonlinearity, which cannot be adequately addressed by linear models. The literature review highlights that the most successful models are based on soft computing approaches such as neural networks, fuzzy systems, evolutionary computing, support vector machines, and hybrid models.

In this paper we investigate the feasibility of using the Dynamic Time Scan Forecasting (DTSF) time scan method to predict short-term (hourly) demand in water supply systems by comparing efficiency and computational cost (CC), with several known univariate alternatives. The statistical and neuronal methods employed were: Box-Jenkins (SARIMA), Exponential Smoothing (ETS), Hybrid model using Seasonal and Trend decomposition using Loess filter (STL) and ETS (STL-ETS), Trigonometric Box-Cox transformation model, ARMA errors, Trend and Seasonal components (TBATS), in addition to hybrid neuronal models with bootstrap nonlinear autoregressive neural network (BNNAR), Extreme Learning Machine (ELM) and Naive Bayes (NB).

The data used in this study refer to a supply zone (ZA) located in the south central region of a capital of south eastern Brazil and serves a population of approximately 230,000 people, and can be compared with a medium-sized city.

In order to evaluate the methods, the following metrics were employed: root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), Model Fitting (MF), in addition to WC. The results show that in both summer and fall there was no model that outperformed the others in terms of forecast accuracy. However, for the months from June to November (winter and spring), the ELM method showed the best results. The two neuronal methods showed the best accuracy, however, the greatest CC when employing cross-validation of sliding origin with recalibration.

Regarding the CC statistic, the DTSF algorithm was extremely fast compared to the other methods studied for all months of the year, using a fraction of the time. The results obtained showed that the proposed method provides similar or improved forecast values compared to the soft computing and statistical methods employed, but using a fraction of the computation time.

The great advantage of this method, as it is a data-driven method, is that the more data you use, the better your generalizability will be, and in an environment known as the big data era, it will take advantage of others methods.

**KEYWORDS:** *Dynamic Time Scan Forecasting (DTSF); water demand forecast; Statistical methods; machine learning; soft computing; computational cost.*

## LISTA DE FIGURAS

<b>Figura 4.1</b> - Métodos quantitativos e qualitativos e alguns de seus modelos.....	27
<b>Figura 4.2</b> - Comparação entre neurônio biológico e artificial.....	51
<b>Figura 4.3</b> - Modelo geral do neurônio artificial.....	52
<b>Figura 4.4</b> - Rede Neural <i>Multilayer Perceptron</i> (MLP).....	54
<b>Figura 4.5</b> - (a) $\hat{f}_1(x)$ é uma MLP com dois neurônios na camada escondida; (b) $\hat{f}_2(x)$ é uma MLP com seis neurônios na camada escondida; e (c) $\hat{f}_3(x)$ é uma MLP com 20 neurônios na camada escondida.....	56
<b>Figura 4.6</b> - Dilema entre a capacidade de generalização da rede e a minimização do erro..	57
<b>Figura 4.7</b> - Representação gráfica dos máximos e mínimos de uma função objetivo.....	57
<b>Figura 4.8</b> - Parada antecipada de treinamento de uma rede MLP.....	59
<b>Figura 4.9</b> - Exibição esquemática do <i>Holdout Cross Validation</i> . Um conjunto de $n$ observações são divididas aleatoriamente em um conjunto de treinamento (em azul, contendo as observações 3, 5 e 9, entre outras) e um conjunto de validação (em cinza e contendo a observação 21, entre outras). O método de aprendizado estatístico é adequado ao conjunto de treinamento e seu desempenho é avaliado no conjunto de validação.....	60
<b>Figura 4.10</b> - Exibição esquemática do LOOCV. Um conjunto de $n$ pontos de dados é repetidamente dividido em um conjunto de treinamento (em azul) contendo todas, exceto uma observação, e um conjunto de validação que contém apenas essa observação (em cinza). O teste do erro é então estimado pela média dos $n$ MSE resultantes. O primeiro conjunto de treinamento contém todas as observações menos a primeira, o segundo conjunto de treinamento contém todas as observações menos a segunda e assim por diante.....	60
<b>Figura 4.11</b> - Exibição esquemática do <i>k-fold CV</i> . Um conjunto de $n$ observações é dividido aleatoriamente em $k$ grupos mutuamente exclusivos. Cada um desses $1/k$ atua como um conjunto de validação (em cinza) e o restante como um conjunto de treinamento (em azul). O erro de teste é estimado pela média do MSE das $k$ estimativas resultantes .....	61
<b>Figura 4.12</b> - Validação cruzada de séries temporais, onde cada conjunto de treino (azul) contém uma observação a mais que o conjunto anterior, e conseqüentemente cada conjunto de teste (cinza) tem menos uma observações que o anterior.....	62
<b>Figura 4.13</b> - Validação cruzada de séries temporais com base em previsões de um passo. Os pontos azuis representam os conjuntos de treino, os cinzas de teste e os pontos em branco são ignorados .....	62
<b>Figura 4.14</b> - Validação cruzada usando janela rolante.....	63

<b>Figura 4.15</b> - Rede neural autorregressiva não-linear (NNAR) .....	66
<b>Figura 4.16</b> - <i>Single-hidden layer feedforward network</i> (SLFN) .....	68
<b>Figura 4.17</b> - Abordagem de filtro no qual a seleção de atributos é independente do algoritmo de indução.....	75
<b>Figura 4.18</b> - Abordagem do invólucro para seleção de atributos.....	77
<b>Figura 4.19</b> - Abordagem embutida ou embarcada para seleção de atributos.....	78
<b>Figura 4.20</b> - Classificação de métodos baseados em dados nos domínios de tempo, frequência e tempo-frequência .....	80
<b>Figura 4.21</b> - Fluxo do Filtro Neural Iterativo (INF).....	82
<b>Figura 4.22</b> - Exemplo ilustrativo de ganho de desempenho usando um comitê de “M” máquinas em predição de séries temporais.....	83
<b>Figura 4.23</b> - Geração de conjuntos de treinamento utilizando <i>bagging</i> .....	89
<b>Figura 4.24</b> - Geração de conjuntos de treinamento utilizando <i>boosting</i> .....	90
<b>Figura 4.25</b> - Diagrama do procedimento de reamostragem <i>bootstrap</i> .....	91
<b>Figura 4.26</b> - Diagrama MBB.....	91
<b>Figura 4.27</b> - Representação de um processo estocástico e seu conjunto de trajetórias.....	93
<b>Figura 4.28</b> - Representação ilustrativa dos componentes tendência e sazonalidade.....	95
<b>Figura 4.29</b> - Decomposição de uma série temporal empregando STL.....	102
<b>Figura 4.30</b> - Perfil de previsão.....	105
<b>Figura 4.31</b> -Filtro linear, com entrada $\epsilon_t$ , função de transferência $\Psi(B)$ e saída $Z_t$ .....	111
<b>Figura 4.32</b> - Decomposição empregando TBATS para uma série temporal de ligações de um call center com frequência a cada 5 minutos.....	121
<b>Figura 4.33</b> - Dada as séries temporais X e Y, para cada ponto de X, o cálculo da distância é realizado apenas uma única vez para apenas um ponto em Y.....	124
<b>Figura 4.34</b> - Alinhamento de séries com diferentes tamanhos empregando a DE.....	125
<b>Figura 4.35</b> - Alinhamento de séries com diferentes tamanhos empregando a DTW.....	125
<b>Figura 4.36</b> - Ilustração do procedimento de varredura da série de tempo DTSF.....	127
<b>Figura 4.37</b> - Exemplo de DTSF usando séries temporais de demanda de água.....	129
<b>Figura 4.38</b> - Previsão de estimativas usando as janelas mais semelhantes, selecionadas usando DTSF.....	130
<b>Figura 5.1</b> - Exibição esquemática da recalibração de origem rolante.....	133

<b>Figura 5.2</b> - Fluxograma proposto para análise e comparação de todos os métodos a serem avaliados.....	134
<b>Figura 6.1</b> - Série histórica de água demandada desde 00:00 de 01 de janeiro de 2015 às 24:00 de 31 de dezembro de 2016.....	138
<b>Figura 6.2</b> - <i>Boxplot</i> dos dados para as 24 horas do dia empregando a vazão (L/s) no período de 00:00 de 01 de janeiro de 2015 às 24:00 de 31 de dezembro de 2016.....	139
<b>Figura 6.3</b> - <i>Boxplot</i> dos dados para os dias da semana empregando a vazão (L/s) no período de 00:00 de 01 de janeiro de 2015 às 24:00 de 31 de dezembro de 2016.....	140
<b>Figura 6.4</b> - <i>Boxplot</i> dos dados para os meses empregando a vazão (L/s) no período de 00:00 de 01 de janeiro de 2015 às 24:00 de 31 de dezembro de 2016.....	140
<b>Figura 6.5</b> - Histograma e densidade de probabilidade da demanda de água da ZA 18 em litros por segundo.....	141
<b>Figura 6.6</b> - Custo computacional do DTSTF <i>versus</i> demais métodos empregados ao longo de 2016.....	156
<b>Figura 6.7</b> - Custo computacional do DTSTF <i>versus</i> os métodos SARIMA, ETS e STL-ETS empregados ao longo de 2016.....	157
<b>Figura 6.8</b> - <i>Bloxplot</i> da mediana dos erros de previsão para as próximas 24 horas, dos modelos avaliados (01 de janeiro a 31 de dezembro de 2016).....	158
<b>Figura 6.9</b> - Evolução da mediana dos erros de previsão, para as próximas 24 horas, de todos os modelos avaliados.....	160
<b>Figura 6.10</b> - Evolução da mediana dos RMSE, para as próximas 24 horas, após aplicar uma média móvel de 7 dias para todos os modelos avaliados.....	161
<b>Figura 6.11</b> - Acurácia dos modelos em prever ao longo dos meses empregando o <i>Model Fitting</i> (MF) no período de 01 de janeiro a 31 de dezembro de 2016.....	163

## LISTA DE TABELAS

<b>Tabela 4.1</b> - Fatores que impactam a demanda urbana de água.....	29
<b>Tabela 4.2</b> - Relação entre nível de planejamento de uma concessionária de água, problemas de decisão e os atributos de previsão.....	29
<b>Tabela 4.3</b> - Métodos de previsão de demanda de acordo com literatura referenciada.....	44
<b>Tabela 4.4</b> - Tipos de busca, critério e avaliação do filtro, invólucro e embarcado dos subconjuntos de variáveis.....	75
<b>Tabela 4.5</b> - Comparação de métodos de seleção de características.....	79
<b>Tabela 4.6</b> - Escolha dos parâmetros do STL, descrição e estimação.....	101
<b>Tabela 4.7</b> - Família de modelos ETS com diferentes tipos de sazonalidade e tendência...	104
<b>Tabela 5.1</b> - Exemplo de matriz de erros para os dias e horizonte de previsão.....	133
<b>Tabela 6.1</b> - Medidas descritivas da série histórica de demanda.....	139
<b>Tabela 6.2</b> - Modelos de previsão empregando SARIMA e suas respectivas métricas (RMSE, MAE e MAPE) para os meses de previsão .....	143
<b>Tabela 6.3</b> - Modelos de previsão empregando alisamento exponencial (ETS) e suas respectivas métricas (RMSE, MAE e MAPE) para os meses de previsão .....	145
<b>Tabela 6.4</b> - Modelos de previsão empregando o STL-ETS e suas respectivas métricas (RMSE, MAE e MAPE) para os meses de previsão .....	146
<b>Tabela 6.5</b> - Modelos de previsão empregando redes neurais autorregressivas não-lineares com <i>bootstrap</i> e suas respectivas métricas (RMSE, MAE e MAPE).....	148
<b>Tabela 6.6</b> - Modelos de previsão ELM, o número de defasagens empregadas e suas respectivas métricas (RMSE, MAE e MAPE).....	149
<b>Tabela 6.7</b> - Modelos de previsão empregando o TBATS e suas respectivas métricas (RMSE, MAE e MAPE).....	151
<b>Tabela 6.8</b> - Modelos de previsão empregando o DTSF e suas respectivas métricas (RMSE, MAE e MAPE) para os meses de previsão .....	152
<b>Tabela 6.9</b> – Avaliação dos métodos de previsão empregando as métricas Raiz do Erro Quadrático Médio (RMSE), Erro Absoluto Médio (MAE), Erro Percentual Médio (MAPE), Complexidade Computacional (CC) para um horizonte de previsão horário .....	155
<b>Tabela 6.10</b> – Número e percentual dos melhores ajustes de previsão de cada um dos modelos para as próximas 24 horas ao longo do ano de 2016, após aplicar média móvel de uma semana.....	162

<b>Tabela 6.11</b> - Performance de previsão empregando <i>Model Fitting</i> (MF) de 01 de janeiro a 31 de dezembro de 2016.....	163
--	-----

## LISTA DE ABREVIATURAS, SIGLAS E SÍMBOLOS

AARE	Erro relativo absoluto médio
ACO	Otimização colônia de formigas
ACSPO	Algoritmo de otimização adaptativa de enxame de partículas de caos
ADALINE	<i>Adaptative Linear Element</i>
AFS	<i>Adaptative Fourier series</i>
AG	Algoritmos genéticos
AIC	Critério de informação Akaike
AICc	Critério de informação Akaike com correção
AMALGAM	Método de otimização multialgoritmo geneticamente adaptativo
ANFIS	Sistema de inferência <i>Fuzzy</i> neuro adaptativa
AR	Modelo autorregressivo
ARE	Erro relativo médio
ARIMA	Modelo autorregressivo integrado de médias móveis
ARMA	Modelo autoregressivo de médias móveis
B	<i>Bootstrap</i>
B-ELM	Modelo híbrido aprendizado extremo da máquina com <i>bootstrap</i>
BIC	Critério de informação bayesiana
B-MLP	Modelo híbrido rede neural artificial com <i>bootstrap</i>
B-NNAR	Modelo híbrido rede neural autorregressiva não-linear com <i>bootstrap</i>
CC	Custo computacional ou complexidade computacional
CCNN	Rede neural de correlação em cascata
CNN	Rede neural convolucional
CID	Distância invariante complexa
CID-DTW	Distância invariante complexa <i>Dynamic Time Warpping</i>
DAAE	Departamento Autônomo de Água e Esgoto
DAN2	Rede neural artificial dinâmica
DAN2-H	Modelo híbrido rede neural artificial dinâmica
DE	Distância Euclidiana
DMA	Áreas de medidas distritais
DTSF	<i>Dynamic Time Scan Forecasting</i>
DTW	<i>Dynamic Time Warping</i>
EDBD	Extended-Delta-Bar-Delta
E-FFNN	Rede neural evolutiva
ELM	Aprendizado extremo da máquina
EMD	Método de decomposição em modo empírico
EMD-MLP	Modelo híbrido que combina decomposição em modo empírico e rede <i>Multilayer Perceptron</i>
EPANET 2	Simulador hidráulico desenvolvido pela <i>U.S. Environmental Protection Agency</i> (EPA)
E-RNA	Redes neurais artificiais evolutivas
ESNN	Rede neural de estado echo
<i>Et</i>	Componente ciclo
ETA	Estação de tratamento de água
ETS	Métodos de alisamento exponencial
ETS (A,Ad,A)	Métodos de alisamento exponencial com erro aditivo, tendência amortecida e sazonalidade aditiva
FA	Floresta aleatória

FAC	Função de autocorrelação
FACP	Função de autocorrelação parcial
FACV	Função de autocovariância
FCM	<i>C-mean fuzzy</i>
fdp	Função de densidade de probabilidade
FFNN	Rede neural <i>feedforward</i>
FIS	Sistema de inferência Fuzzy
FKE	Filtro <i>kalman</i> estendido
FRNN	Rede neural totalmente recorrente
FTS	<i>Fuzzy Takagi-Sugeno</i>
GARCH	Heterocedasticidade condicional autorregressiva generalizada
GRASP	Procedimentos ágil de pesquisa adaptativa aleatória
GRNN	Rede neural de regressão generalizada
GSAA	Algoritmo genético de recozimento simulado
HCV	<i>Holdout Cross Validation</i>
HMC	Cadeia de <i>Markov</i> homogêneas
HW	<i>Holt-winters</i>
I	Filtro de integração
I.C.	Intervalo de confiança
<i>i.i.d.</i>	Independente e identicamente distribuídas
IMF	Funções de modo intrínseco
INF	Filtro neural iterativo
JB	Teste de normalidade de Jarque-Bera
K	Função do núcleo Kernel
K-Fold	<i>K-fold Cross Validation</i>
K-NN	<i>K-Nearest Neighbors</i>
KPSS	Teste KPSS
L-B	Teste de autocorrelação de Ljung-Box
LFT	Função de transferência linear
L.I.	Limite inferior do intervalo de confiança
LM	Multiplicador de Lagrange
LOESS	Regressão localmente ponderada
LOOCV	<i>Leave-One-Out Cross Validation</i>
L.S.	Limite superior do intervalo de confiança
LS-SVM	Máquina de vetor suporte de mínimos quadrados
MA	Componente de média móvel
MAE	Erro absoluto médio
MAPE	Erro percentual absoluto médio
MARS	Regressão multivariada adaptativa <i>splines</i>
MASE	Erro médio absoluto escalonado
MASS	Mueen's algoritmo de pesquisa de similaridade
MaxARE	Erro relativo absoluto máximo
MBB	<i>Moving block bootstrap</i>
MC	Cadeia de <i>Markov</i>
Md	Mediana
MF	<i>Model fitting</i>
MFIS	Inferência <i>Mamdani Fuzzy</i>
MLP	<i>Multilayer perceptron</i>
Moram I	Estatística de Moran I
MSE	Erro quadrático médio

MS-RVR	Regressão vetorial de relevância multiescala
NB	<i>Naive Bayes</i>
NHMC	Cadeia de <i>Markov</i> não homogêneas
NNAR	Rede neural autorregressiva não-linear
NRMSE	Erro de raiz quadrada média normalizada
NSI	<i>Nash-sutcliffe</i>
PDA	Módulo de previsão de demanda
Pdv	Desvio percentual no pico
pE-FFNN	Redes neurais evolutivas em paralelo
PG	Programação genética
PI	Índice de persistência
PIB	Produto interno bruto
PNN	Rede neural probabilística
PPR	Regressão não paramétrica <i>projection pursuit</i>
Q-Q	Gráfico de probabilidade Quantil-Quantil
QuickProp	<i>Quick propagation</i>
R	Coefficiente de correlação
R <sup>2</sup>	Coefficiente de determinação
RBNN	Rede neural com função de base radial
rE-FFNN	Redes neurais evolutivas recursiva
RLF	Regressão linear <i>Fuzzy</i>
RLM	Regressão linear múltipla
RMBH	Região metropolitana de Belo Horizonte
RMSE	Raiz do erro quadrático médio
RMSP	Região metropolitana de São Paulo
RNA	Redes neurais artificiais
RNLM	Regressão não-linear múltipla
RNN	Redes neurais recorrentes
RPROP	<i>Resilient propagation</i>
RRMSE	Raiz do erro relativo ao quadrado médio
RVR	Regressão vetorial de relevância
RW	Passeio aleatório
RWD	Passeio aleatório com tendência
SAA	Sistemas de abastecimento de Água
SAR	Modelo espacial autoregressivo
SARIMA	Modelo autorregressivo, sazonal, integrado de média móvel
SARMA	Modelo espacial autoregressivo de média móvel
SC	<i>Soft computing</i>
SEM	Modelo de erro espacial
SEP	Erro de Previsão Padrão
SLFN	<i>Single-hidden Layer Feedforward Network</i>
SMA	Média móvel sazonal
sMAPE	Erro percentual absoluto médio simétrico
SOM	Mapa auto organizado
SPD	Sistema de previsão de demanda
ST	Séries temporais
<i>St</i>	Componente sazonal
STL	<i>Seasonal and Trend decomposition using Loess</i>
STL-ETS	Modelo híbrido que combina STL-ETS
SVD	Decomposição do valor singular

SVR	Regressão vetor suporte
TBATS	<i>Trigonometric Box-Cox transformation ARMA erros, Trend and Seasonal componentes</i>
TSC	Classificação de séries temporais
TS-GRNN	Modelo híbrido série temporal e rede neural de regressão generalizada
<i>Tt</i>	Componente tendência
VAR	Vetor autoregressivo
W	<i>Wavelet</i>
WB-MLP	Modelo híbrido rede neural artificial com <i>wavelet</i> e <i>bootstrap</i>
W-ELM	Modelo híbrido aprendizado extremo da máquina com <i>wavelet</i>
W-MLP	Modelo híbrido rede neural artificial com <i>wavelet</i>
YWS	<i>Yorkshire Water Services</i>
ZA	Zona de abastecimento

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO.....</b>	<b>21</b>
<b>2</b>	<b>OBJETIVOS.....</b>	<b>24</b>
2.1	OBJETIVO GERAL.....	24
2.2	OBJETIVOS ESPECÍFICOS .....	24
<b>3</b>	<b>JUSTIFICATIVA .....</b>	<b>25</b>
<b>4</b>	<b>REVISÃO DA LITERATURA .....</b>	<b>26</b>
4.1	PREVISÃO DE DEMANDA DE ÁGUA .....	26
4.2	MODELOS DE PREVISÃO DE DEMANDA .....	30
4.2.1	<i>Redes neurais</i> .....	30
4.2.2	<i>Modelos Fuzzy e neuro-Fuzzy</i> .....	35
4.2.3	<i>Máquinas de vetores suporte</i> .....	36
4.2.4	<i>Computação evolucionária</i> .....	38
4.2.5	<i>Modelos espaciais</i> .....	40
4.2.6	<i>Modelos estatísticos</i> .....	41
4.2.7	<i>Discussão acerca da revisão bibliográfica dos modelos de previsão de demanda</i> .....	48
4.3	REDES NEURAIS .....	50
4.3.1	<i>Conceitos gerais</i> .....	50
4.3.2	<i>A rede neural Multilayer Perceptron (MLP)</i> .....	53
4.3.3	<i>Regras de aprendizagem RNA</i> .....	54
4.3.4	<i>Capacidade de generalização</i> .....	56
4.3.5	<i>Parada antecipada (early stopping)</i> .....	58
4.3.6	<i>Métodos de separação de amostra para fazer a validação cruzada</i> .....	59
4.3.7	<i>Dilema viés-variância</i> .....	64
4.3.8	<i>Rede neural autorregressiva não-linear (NNAR)</i> .....	65
4.4	EXTREME LEARNING MACHINE (ELM).....	67
4.4.1	<i>Conceito geral</i> .....	67
4.4.2	<i>Algoritmo extreme learning machine</i> .....	70
4.5	SELEÇÃO DE MODELOS .....	73
4.6	SELEÇÃO VARIÁVEIS E/OU ATRIBUTOS (FILTROS, INVÓLUCROS E EMBARCADOS) .....	74
4.7	ENSEMBLE .....	82
4.7.1	<i>Conceito geral</i> .....	82
4.7.2	<i>Fundamento teórico para o uso de um comitê</i> .....	85
4.7.3	<i>Bagging</i> .....	87
4.7.4	<i>Boosting</i> .....	89
4.7.5	<i>Bootstrap</i> .....	90
4.8	SÉRIES TEMPORAIS .....	92
4.9	DECOMPOSIÇÃO DE SÉRIES TEMPORAIS .....	95
4.9.1	<i>A Seasonal Trend decomposition procedures based on Loess (STL)</i> .....	96
4.10	MODELO NO CONTEXTO ESPAÇO TEMPO .....	102
4.10.1	<i>Modelo de alisamento exponencial</i> .....	102
4.10.2	<i>Formulação dos modelos de alisamento exponencial</i> .....	103
4.11	METODOLOGIA BOX E JENKINS .....	110
4.11.1	<i>Conceito geral</i> .....	110
4.11.2	<i>Modelos estacionários</i> .....	110
4.11.3	<i>Modelos autorregressivos AR(p)</i> .....	111
4.11.4	<i>Modelos de médias móveis MA(q)</i> .....	112
4.11.5	<i>Modelo autorregressivos de médias móveis ARMA(p,q)</i> .....	112
4.11.6	<i>Modelo ARMA integrados ARIMA(p,d,q)</i> .....	114
4.11.7	<i>Etapas da metodologia Box e Jenkins</i> .....	115
4.11.8	<i>Crerios de comparação e seleção de modelos</i> .....	117
4.12	TRIGONOMETRIC BOX-COX TRANSFORMATION ERROS, TREND AND SEASONAL COMPONENTS (TBATS). 119	
4.13	NAIVE BAYES.....	122

4.14	MÉTODOS DE SIMILARIDADE BASEADO EM DISTÂNCIA .....	123
4.14.1	<i>Distância Euclidiana (DE) e Dynamic Time Warping (DTW)</i> .....	123
4.15	DYNAMIC TIME SCAN FORECASTING (DTSF) .....	125
4.15.1	<i>Background histórico</i> .....	125
4.15.2	<i>Formulação do Dynamic Time Scan Forecasting (DTSF)</i> .....	126
<b>5</b>	<b>MATERIAL E MÉTODOS .....</b>	<b>131</b>
5.1	MÉTRICAS DE AVALIAÇÃO .....	135
5.2	INTERVALO DE CONFIANÇA (I.C.) EMPÍRICO.....	137
<b>6</b>	<b>RESULTADOS E DISCUSSÃO .....</b>	<b>138</b>
6.1	ESTUDO DE CASO .....	138
6.2	APLICAÇÃO DOS MÉTODOS DE PREVISÃO .....	141
6.2.1	<i>Box e Jenkins</i> .....	141
6.2.2	<i>Alisamento Exponencial (ETS)</i> .....	144
6.2.3	<i>Modelo híbrido STL-ETS</i> .....	146
6.2.4	<i>Redes Neurais não-lineares autorregressivas com Bootstrap (BNNAR)</i> .....	147
6.2.5	<i>Extreme learning machine (ELM)</i> .....	148
6.2.6	<i>Trigonometric Box-Cox transformation erros, Trend and Seasonal components (TBATS)</i> .....	150
6.2.7	<i>Dynamic Time Scan Forecasting (DTSF)</i> .....	152
6.2.8	<i>Análise comparativa dos modelos empregando as diversas técnicas</i> .....	153
<b>7</b>	<b>CONCLUSÕES .....</b>	<b>164</b>
<b>8</b>	<b>RECOMENDAÇÕES E TRABALHOS FUTUROS .....</b>	<b>166</b>
	<b>REFERÊNCIAS .....</b>	<b>167</b>
	<b>ANEXO I.....</b>	<b>193</b>
	<b>ANEXO II .....</b>	<b>196</b>
	<b>ANEXO III.....</b>	<b>199</b>

# 1 INTRODUÇÃO

*“Aprender é a única coisa que a mente nunca se cansa, nunca tem medo e nunca se arrepende”.*

Leonardo da Vinci

A crescente demanda de água e o risco eminente de desabastecimento estão associados ao aumento da população urbana e do consumo *per capita*. Dados da Organização Meteorológica Mundial mostram que o consumo mundial de água aumentou mais de seis vezes em menos de um século, mais do que o dobro das taxas de crescimento da população (FREITAS e SANTOS, 1999).

Uma grande proporção da população mundial está passando por estresse hídrico (VÖRÖSMARTY *et al.*, 2000). A disponibilidade de água em quantidade e qualidade representa um grande desafio, principalmente nas grandes metrópoles. Muitas metrópoles têm crescido desordenadamente, as coberturas vegetais são removidas e os solos impermeabilizados causando alterações hidrológicas e meteorológicas, tais como o aumento da temperatura do ar e de riscos de enchentes. Além disso, devido ao crescimento urbano, a disponibilidade de água está cada vez menor, seja devido à poluição ou à devastação de mananciais, e as fontes disponíveis para captação e abastecimento estão cada vez mais distantes dos centros consumidores, tornando a exploração cada vez mais dispendiosa.

Na operação de Sistemas de Abastecimento de Água (SAA), um dos fatores críticos do abastecimento é a manutenção do equilíbrio entre a oferta e a demanda de água aos usuários. A manutenção deste equilíbrio é feita através de ações operacionais, muitas delas aplicando conceitos e ferramentas de previsão. A previsão do consumo é imprescindível em vários processos dentro de uma empresa de saneamento, principalmente nos setores de projetos de novos sistemas e de ampliação dos sistemas existentes, e nos setores de produção de água (captação, tratamento, reservação e distribuição).

Nos processos de produção e de distribuição, durante a operação de um SAA, a previsão do consumo é feita com base em dados históricos de demanda que dependem de diversos fatores, tais como o tamanho da cidade, o consumo *per capita*, a renda, o número médio de habitantes por residência, a precipitação, a temperatura, a altitude, o preço da água, a hora do dia, o dia da

semana, as estações do ano (ARBUÉS *et al.*, 2003; RUJIS *et al.*, 2008; SCHEICH e HILLENBRAND, 2009; NAUGES e WHITTINGTON, 2010; BABEL *et al.*, 2014; MARIA ANDRÉ e CARVALHO, 2014; ROMANO e KAPELAN, 2014; ROMANO *et al.*, 2016). O consumo de água também é afetado pelas “perdas” devido a vazamentos (BORGES, 2003).

São inúmeras as vantagens de se obter um modelo de previsão de demanda de água para otimização da distribuição, tais como a identificação rápida de vazamentos, ganho na qualidade da água distribuída e planejamento otimizado da operação da estação de tratamento de água. Entretanto, a otimização desta operação depende basicamente das previsões precisas das demandas horárias e do planejamento das operações de bombeamento durante os períodos de baixas tarifas de energia elétrica (GHIASSI *et al.*, 2008). Este conjunto de informações também é essencial na composição de estratégias de expansão de longo prazo, pois auxilia em questões de planejamento empresarial, tais como analisar a tendência do consumo de água, medir os esforços das políticas de conservação, a definição de tarifas e a previsão de receitas, além de reduzir gastos em energia elétrica e insumos de tratamento.

A fim de solucionar o problema da oferta de água, devem ser criados planos de ação que envolvam a combinação das seguintes características: otimização dos sistemas de abastecimento por meio do seu controle e o desenvolvimento de estratégias de expansão de infraestrutura sustentável e de baixo custo (ADAMOWSKI, 2008).

Segundo McDONALD *et al.* (2014), apesar de toda essa infraestrutura de água urbana existente, muitas cidades permanecem sob estresse hídrico: um quarto ( $25 \pm 4\%$ ), ou seja,  $381 \pm 55$  milhões de pessoas têm fontes de água que estão sob estresse. Este cenário demonstra que a escassez hídrica para abastecimento urbano atinge inúmeras cidades, particularmente as regiões metropolitanas, e essa situação não foi diferente nos estados do Sudeste Brasileiro. Como exemplos, pode-se citar a crise hídrica na Região Metropolitana de São Paulo (RMSP) entre os anos de 2014 e 2016 e na Região Metropolitana de Belo Horizonte (RMBH) em 2015, onde o volume de água disponível ficou abaixo da média histórica gerando problemas de abastecimento. Sendo assim, o gerenciamento da água nestas regiões, e em tantas outras, requer um sistema de informação de prognóstico com o objetivo de orientar as tomadas de decisão. Nesse sentido, emergem algumas questões, tais como: quais os impactos das condições meteorológicas no consumo? Quais as características do consumo regional? Em que dias da semana a demanda é maior?

Portanto, o principal objetivo desta tese é avaliar a capacidade de predição da demanda urbana no curto prazo de sistemas de abastecimento de água por meio de modelos univariados, tais como: *Naive Bayes* (NB), alisamento exponencial (ETS), Box-Jenkins (SARIMA), *Trigonometric Box-Cox transformation*, *ARMA erros*, *Trend and Seasonal components* (TBATS), rede neural autorregressiva não-linear (NNAR), *Extreme Learning Machine* (ELM) e *Dynamic Time Scan Forecasting* (DTSF). Espera-se que o método proposto DTSF tenha um desempenho semelhante ou superior aos métodos comumente usados na área do saneamento.

## 2 OBJETIVOS

### 2.1 *Objetivo geral*

Pesquisar a viabilidade da utilização do método de varredura *Dynamic Time Scan Forecasting* (DTSF) para prever a demanda de curto prazo em sistemas urbanos de abastecimento de água por meio da comparação, em termos de eficiência e do custo computacional, com as principais alternativas univariadas conhecidas: *Naive Bayes* (NB), Box-Jenkins (SARIMA), alisamento exponencial (ETS), modelo híbrido usando *Seasonal and Trend decomposition using Loess* (STL) e o ETS (STL-ETS), *Trigonometric Box-Cox transformation*, *ARMA erros*, *Trend and Seasonal components* (TBATS), rede neural autorregressiva não-linear com *bootstrap* (BNNAR) e *Extreme Learning Machine* (ELM).

### 2.2 *Objetivos específicos*

Especificamente pretende-se:

- Aplicar o método proposto DTSF, visto que não há na literatura emprego de tal metodologia na previsão de demanda urbana de água;
- Verificar o desempenho dos modelos univariados em realizar prognósticos na previsão de demanda de água para sistema de abastecimento de água; e
- Comparar o desempenho dos modelos construídos, tanto na capacidade de generalização quanto no custo computacional demandado.

### 3 JUSTIFICATIVA

A previsão de demanda de curto e longo prazo é um fator de grande importância para o gerenciamento do consumo de água, principalmente quando se pretende criar ou expandir um sistema de abastecimento urbano. Existem diversos métodos de previsão de demanda que podem ser amplamente classificados em lineares e não-lineares (ZHANG, 2001). Os modelos lineares empregam a análise univariada de séries temporais como, por exemplo, autorregressivo integrado de médias móveis (ARIMA) e os modelos de regressão linear. Entre os modelos não-lineares destacam-se os de regressão não-lineares, modelos de lógica *Fuzzy*, modelos de redes neurais artificiais (RNA), modelos híbridos baseados nas RNA e métodos de dissimilaridade Distância Euclidiana (DE) e o *Dynamic Time Warping* (DTW).

Os modelos lineares têm sido amplamente utilizados, porque eles são fáceis de desenvolver e implementar, além de simples de entender e interpretar. No entanto, os dados de demanda de água têm diferentes graus de não-linearidade, que não podem ser adequadamente tratados pelos modelos lineares. A principal motivação para o uso do ELM e do NNAR ocorre devido à sua capacidade de modelar comportamentos não-lineares, e do algoritmo DTW, pela sua qualidade como medida de distância adequada para a classificação de séries temporais por dissimilaridade e pelo baixíssimo custo computacional.

O emprego deste algoritmo, originalmente apresentado no contexto de previsão de demanda de vento (COSTA *et al.*, 2019), decorre devido a sua eficiência na medida de similaridade tempo/série, ainda não aplicado no contexto de previsão de demanda de água. Espera-se que o método de varredura temporal tenha um ganho preditivo e computacional quando comparado com os outros modelos comumente usados na área do saneamento. Por esta razão, estudar a previsão de demanda de água e o desempenho do algoritmo DTW são as principais motivações nesta tese.

## 4 REVISÃO BIBLIOGRÁFICA

### 4.1 *Previsão de demanda de água*

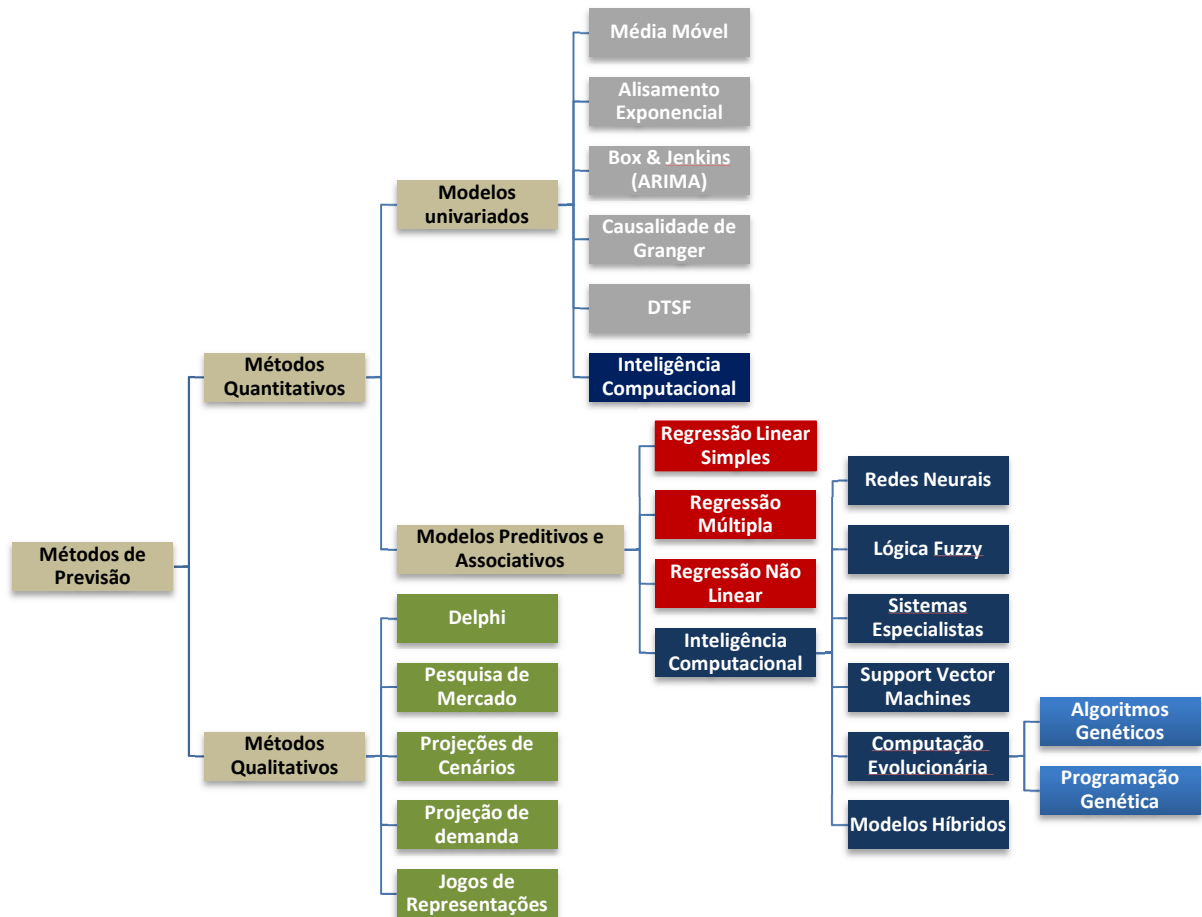
A previsão de demanda é de fundamental importância no mundo corporativo, auxiliando os gestores na tomada de decisão, como, por exemplo, os recursos necessários ao negócio. A escolha do método a ser utilizado depende da natureza do bem ou serviço e de vários fatores, tais como a disponibilidade de dados históricos, horizonte de previsão (curto e longo prazo), precisão, orçamento disponível e padrões dos dados (tendência e sazonalidade). As técnicas de previsão são classificadas em qualitativas, baseadas em estimativas e opiniões de especialistas, e, em quantitativas, baseadas em modelos matemáticos.

As técnicas quantitativas se dividem em dois tipos de modelos. Os modelos de séries temporais que utilizam apenas dados históricos de demandas como base para a determinação de padrões que podem se repetir no futuro e os associativos, que têm como objetivo relacionar a variável dependente com outras variáveis, tais como, consumo *per capita*, renda, número médio de habitantes por residência, índice de perdas<sup>1</sup>, idade média da população, precipitação, temperatura, altitude, preço, produto interno bruto (PIB) e inflação.

Na Figura 1 apresentam-se as técnicas dos métodos quantitativos e qualitativos, e alguns de seus modelos.

---

<sup>1</sup> O índice de perdas usualmente é governado por diversos fatores tais como: comprimento da rede de distribuição, pressões estáticas e dinâmicas na rede e percentual de macro e micromedicação.



**Figura 4.1** – Métodos quantitativos e qualitativos e alguns de seus modelos (adaptado de TRAUTWEIN JÚNIOR, 2004).

Atualmente, existe uma grande preocupação por parte dos gestores com o planejamento do SAA, de forma a atender à demanda de água a custos operacionais menores, ao mesmo tempo em que, na área de projetos, tem se intensificado a avaliação do uso da infraestrutura existente e das estratégias de expansão (plano diretor e estudo de concepção/projeto de construção).

Os planos diretores visam contemplar uma expectativa futura de longo prazo para a definição dos planos de investimentos, vinculando as demandas de água às tendências de crescimento vegetativo da população abastecida, ponderado sobre as zonas de expansão, considerando as variáveis socioeconômicas e climáticas que alteram o comportamento de consumo no longo prazo.

As diretrizes definidas nos planos diretores norteiam os estudos de concepção e projetos de construção de obras civis e hidráulicas que garantam o abastecimento, e para isso há de se considerar o consumo médio do setor, podendo, assim, elaborar projetos “personalizados” para cada zona de abastecimento. À medida que os anos passam essas obras precisam ser readequadas, objetivando atender o novo consumo demandado, sendo orientado pelo perfil médio de consumo diário e dos respectivos coeficientes de variação do consumo do setor a ser estudado.

De acordo com Falkenberg (2005), o conhecimento antecipado do perfil de demanda de água, ao longo do dia, permitirá o correto planejamento de operação do respectivo SAA, possibilitando a otimização dos custos de produção e de distribuição, mantendo-se a qualidade do serviço e do produto para o consumo. A correta operação do sistema reduzirá os custos com produtos químicos, com as perdas físicas nas redes por vazamento e com energia elétrica, evitando a operação das bombas em horários mais onerosos. Já a melhoria da qualidade do serviço será alcançada à medida que se evitará a falta de água em regiões mais críticas nos horários de picos de consumo, além de possibilitar uma detecção mais eficaz de vazamentos na rede de distribuição. Ao realizar previsões para as próximas horas ou próximos dias, os operadores do SAA serão munidos de informações e obterão respostas instantâneas a picos prolongados e flutuações acentuadas de demanda, ocasionadas por possíveis ocorrências de vazamentos na rede, agilizando a manutenção e reduzindo as perdas ocasionadas. No tocante à melhoria na qualidade da água, essa será garantida pela manutenção dos volumes adequados dos reservatórios possibilitando a eliminação de superdosagem de produtos químicos, principalmente de cloro, nas estações de tratamento de água (ETA), evitando a perda de qualidade da água causada pelo elevado tempo entre a produção e o consumo.

Existem vários fatores que afetam a quantidade de água consumida e os mais importantes são: condições climáticas como temperatura, precipitação e umidade relativa; natureza da cidade; pressão na rede; perdas no sistema; preço e estrutura tarifária (residencial, comercial, industrial e público); sistema de fornecimento e cobrança (hidrometração); renda; porte da família e da residência; tamanho do espaço ao ar livre (ARBUÉS *et al.*, 2003; TSUTIYA, 2006; WENTZ e GOBER, 2007; RUJIS *et al.*, 2008; SCHEICH e HILLENBRAND, 2009; NAUGES e WHITTINGTON, 2010; MARIA ANDRÉ e CARVALHO, 2014; ROMANO *et al.*, 2016).

Trautwein Júnior (2004) relacionou os fatores que impactam a demanda de água de acordo com o horizonte de previsão, apresentados na Tabela 4.1.

**Tabela 4.1** – Fatores que impactam a demanda urbana de água.

<b>Periodicidade de previsão</b>	<b>Fatores que impactam a demanda</b>
Horário	Período do dia
Horário e diário	Temperatura, umidade ou precipitação, dia da semana
Semanal e mensal	Épocas do ano
Mensal, semestral e anual	Variações do número de ligações por ampliações do sistema ou mudanças nos limites das zonas de abastecimento

**Fonte:** Adaptado de Trautwein Júnior (2004)

A Tabela 4.2 mostra a categorização do horizonte de previsão por nível de planejamento, resumindo o problema da decisão por parte dos executivos e técnicos dos SAA e a respectiva periodicidade de previsão.

**Tabela 4.2** – Relação entre nível de planejamento de uma concessionária de água, problemas de decisão e atributos de previsão.

<b>Nível de Planejamento</b>	<b>Problema de decisão</b>	<b>Horizonte de previsão</b>	<b>Periodicidade de previsão</b>
Operacional	Gerenciamento do sistema de operação	Curto prazo (menor que um ano)	Horário, diário, semanal e mensal
Tático	Previsão de receitas, planos de investimentos e sistema de melhoria	Médio prazo (1 a 10 anos)	Mensal e anual
Estratégico	Capacidade de expansão	Longo prazo (mais que 10 anos)	Anual

**Fonte:** Donkor *et al.* (2014)

## 4.2 Modelos de previsão de demanda

### 4.2.1 Redes Neurais

Trautwein Júnior (2004) realizou um estudo da previsão da demanda de água usando dois métodos: alisamento exponencial e redes neurais artificiais (RNA). Empregou-se a rede *Multilayer Perceptron* (MLP) e a rede com função de base radial (RBNN). A análise baseou-se em dados de curtíssimo prazo (horizontes de quinze minutos), para duas zonas de abastecimento com diferentes características de consumo. Os resultados da pesquisa demonstraram a eficiência da previsão de demanda ao longo de um período de horas. No entanto, eles também demonstraram a inexistência de um modelo ideal, entre os modelos analisados, aplicável a todos os SAA.

Falkenberg (2005) desenvolveu diferentes modelos de previsão de demanda de água no curto prazo utilizando diferentes técnicas, tais como MLP, regressão linear múltipla (RLM) e modelos do tipo *Box e Jenkins*. Os modelos foram aplicados para três bairros de Ponta Grossa com perfis distintos de consumo (Uvaranas – residencial; centro histórico – comercial; e distrito industrial – misto). Para cada uma das regiões foram propostos e comparados seis modelos MLP, três modelos RLM, sete modelos *Box e Jenkins* e dois modelos híbridos. O procedimento empregado foi a técnica de previsão com realimentação, no qual o primeiro passo foi prever o consumo da hora seguinte, de forma que este valor será entrada de consumo passado para previsão de consumo duas horas adiante, e assim sucessivamente. Porém, este método não é indicado para previsões de muitos passos devido ao erro que se acumula a cada passo de previsão. Foram utilizados três tipos de estruturas de dados para comporem a camada de entrada da rede neural, definidos como RNh, RNd1 e RNd2 (nos quais, h é a hora, d1 é a primeira diferença e d2 é a segunda diferença). Para cada uma das três estruturas propostas, Falkenberg utilizou duas redes neurais: uma rede neural artificial tipo MLP com uma camada escondida e função de ativação linear, e uma MLP com uma camada de neurônios ocultos ativados por uma função tangente hiperbólica. Em ambas se utilizou o algoritmo *Levenberg-Marquardt*. Assim como Trautwein Júnior (2004), Falkenberg (2005) concluiu que não existe um único modelo para a solução de todos os casos. Cada região deve ser estudada separadamente, buscando agregar os pontos fortes de cada modelo em uma combinação de modelos ou na escolha de um modelo mais adequado para uma dada ocasião.

Adamowski (2008) realizou previsões dos picos diários de demanda de água para Ottawa (Canadá). A análise foi realizada utilizando dados de pico de consumo de água e variáveis meteorológicas (precipitação e temperatura máxima), para os meses de verão, ao longo de um período de dez anos. Eles desenvolveram 39 modelos de regressão linear múltipla, nove modelos de séries temporais (ARIMA) e 39 modelos de redes neurais MLP, e compararam os desempenhos relativos entre eles. Os melhores resultados foram obtidos quando a demanda diária máxima de água para o dia anterior, a temperatura máxima para o dia corrente e anterior e a precipitação para os últimos cinco dias foram utilizados como dados de entrada. Verificou-se também que a demanda diária de pico se correlaciona melhor com a ocorrência de precipitação do que o próprio volume precipitado. Além disso, atribuir uma ponderação de dias anteriores a “sem chuva”, não resultam em modelos mais precisos.

Ghiassi *et al.* (2008) desenvolveram um modelo de rede neural artificial dinâmica (DAN2) para prever a demanda de água em uma cidade da Califórnia (Estados Unidos da América). Este modelo dinâmico é um caso especial da arquitetura da rede neural *feedforward* (FFNN). O modelo desenvolvido apresentou melhor desempenho que os modelos ARIMA e de redes neurais de retropropagação, provando ser mais efetivo na previsão da demanda de água. Os autores observaram que a inclusão de informações meteorológicas em modelos de previsão aumenta a precisão. No entanto, mesmo ao usar apenas dados de demanda de água, os modelos DAN2 fornecem excelentes ajustes. Os resultados obtidos para as previsões mensais, semanais e diárias foram altamente precisos, bem como os modelos horários. Estes resultados demonstram uma excelente eficácia para a DAN2 na previsão da demanda de água urbana para todos os horizontes de tempo. Segundo os pesquisadores, a otimização das operações pode resultar em economias substanciais de 25% a 30% nos custos operacionais, devido à redução de custos com eletricidade e insumos de tratamento. Resultado semelhante foi obtido por Odan (2013).

Firat *et al.* (2009b) estimaram a previsão mensal de demanda de água na cidade de Izmir (Turquia) usando várias técnicas de redes neurais, como redes neurais *feedforward* (FFNN), de regressão generalizada (GRNN<sup>2</sup>) e RBNN. Eles usaram vários fatores socioeconômicos e climáticos que afetam o consumo (consumo médio mensal de água, população, número de domicílios, produto interno bruto, temperatura média mensal, precipitação mensal total,

---

<sup>2</sup> GRNN também chamada de Regressão de Kernel ou Estimador Nadaraya-Watson, e implementada pelo algoritmo proposto por Specht (1991)

umidade média mensal e inflação). O conjunto de dados foi dividido em dois subconjuntos (treinamento e teste). Os modelos que obtiveram os melhores ajustes também foram comparados ao modelo RLM. Os resultados obtidos indicaram que o GRNN supera todos os outros métodos na modelagem do consumo mensal de água.

Posteriormente, Firat *et al.* (2010) também estimaram a previsão mensal de demanda de água da cidade de Izmir (Turquia), empregando várias técnicas de RNA, incluindo rede neural de correlação em cascata (CCNN), GRNN e FFNN. Seis modelos de previsão foram construídos. A melhor estrutura de entrada de ajuste foi investigada comparando as técnicas empregadas. O modelo M5 (com defasagens de cinco meses) mostrou-se mais eficiente que os demais modelos. Ao comparar os resultados usando as três técnicas de RNA, o desempenho do modelo M5-CCNN foi ligeiramente melhor que os demais.

Adamowksi e Karapataki (2010) analisaram a demanda de água empregando dados de pico semanal e as variáveis meteorológicas temperatura máxima semanal e precipitação total para duas regiões distintas na cidade de Nicósia (Chipre). Os autores desenvolveram e compararam o desempenho relativo de 20 modelos RLM e 60 modelos MLP usando três diferentes algoritmos de aprendizagem (*Levenberg-Marquardt*, *Resilientes Backpropagation* e *Gradiente Conjugado Modelos Powell Beale*). Para as duas regiões analisadas, o método empregando o algoritmo de *Levenberg-Marquardt* apresentou a previsão mais precisa da demanda de pico semanal.

Azadeh *et al.* (2012) apresentaram uma abordagem híbrida, usando RNA e regressão linear *Fuzzy* (RLF), para melhorar a previsão da demanda de água. Segundo os pesquisadores, essa abordagem pode ser facilmente aplicada em ambientes incertos ou complexos, dada a sua flexibilidade. Sua proposta de abordagem híbrida foi aplicada para prever a demanda diária de água em Teerã (Irã). As variáveis utilizadas foram a temperatura máxima diária, a temperatura máxima prevista para o dia seguinte, o índice de precipitação e a demanda em dias quentes e dias frios. Os resultados indicaram que a RNA superou a RLF nos dias quentes devido à sua capacidade de lidar com a complexidade e a não-linearidade. No entanto, tanto RNA e RLF foram ideais em dias frios.

Adamowski *et al.* (2012) realizaram previsões diárias de demanda de água para os meses de verão na cidade canadense de Montreal. Os autores empregaram um método híbrido baseado

em transformações discretas *wavelet* (W) acoplados com redes neurais MLP usando o algoritmo *Levenberg-Marquardt*. Os modelos híbridos W-MLP foram comparados aos modelos MLP, RLM, regressão não linear múltipla (RNLM) e ARIMA, para prever um passo à frente. Os resultados indicam que os modelos W-MLP foram mais robustos que todos os outros modelos, sugerindo o potencial promissor deste método para prever a demanda de água urbana. Segundo os autores, a precisão do modelo de previsão W-MLP pode ser útil no gerenciamento, planejamento e avaliação de sistemas existentes, iniciativas de conservação, análise de condições de seca e políticas de precificação de água.

Odan e Reis (2012) objetivaram identificar um modelo que melhor se adequasse aos dados de consumo horário para uma determinada zona de abastecimento no município de Araraquara (Brasil). Antes de utilizar os valores observados, utilizou-se o procedimento de pré-processamento de dados perdidos resultantes de falhas de registros, ou a presença de valores maiores ou menores que o dobro do valor absoluto do desvio padrão. O método de pré-processamento utilizado foi a análise bayesiana de componentes principais desenvolvido por Oba *et al.* (2003). Este método é baseado em uma regressão de componentes principais, estimativa bayesiana e um algoritmo de maximização de expectativa repetitiva. O método utiliza um algoritmo bayesiano variante iterativo para estimar a distribuição posterior dos parâmetros do modelo e os dados defeituosos, até a sua convergência. Segundo os pesquisadores, a técnica de pré-processamento utilizada tem bons resultados, mesmo quando faltam 40% dos dados, excedendo assim o desempenho dos modelos baseados em *K-Nearest Neighbors* (K-NN) e em decomposição de valor singular (SVD). Após a finalização do pré-processamento, foi utilizada a análise de correlação para identificar as variáveis de entrada (temperatura, umidade relativa, consumo de tempo e nível do reservatório). Observou-se que a inclusão de variáveis climáticas (temperatura e umidade relativa) melhorou a previsão de demanda. A variável consumo no tempo foi considerada na análise de correlação, devido ao consumo cíclico de água ao longo do dia. Este estudo abordou o problema da predição de um SAA em tempo real usando modelos RNA, DAN2 e híbridos. Os modelos híbridos utilizaram o erro produzido pela predição da série de Fourier como entrada, e os resultados da utilização desses modelos mostraram-se promissores. O modelo híbrido DAN2-H apresentou os melhores resultados, tanto para as previsões horárias quanto para as diárias.

Santos e Pereira Filho (2014) realizaram um estudo para prever a demanda de água na região metropolitana de São Paulo (RSPM), utilizando o consumo de água do Sistema Cantareira e

variáveis meteorológicas e socioambientais, empregando redes neurais artificiais de três camadas. Segundo os autores, tempo, clima e efeitos antrópicos tendem a resultar em variações as quais podem contribuir para melhoria da previsão de demanda. Portanto, eles podem ser usados para otimizar o tratamento e distribuição de água com antecipação de até um dia. Oito modelos de configuração foram desenvolvidos. Desses oito modelos, o melhor desempenho foi obtido para a média de 12 horas das variáveis de entrada. Este modelo proporcionou melhor treinamento e desempenho prognóstico do que o modelo de regressão linear múltipla.

Segundo Al-Zahrani e Abo-Monasar (2015), os fatores climáticos desempenham um papel fundamental na previsão da demanda de água no curto prazo, uma vez que influenciam diretamente o consumo de água. O estudo foi conduzido na cidade de Al-Khobar (Arábia Saudita), e empregou os parâmetros climáticos de umidade e temperatura (mínimo, médio e máximo), intensidade de chuvas, ocorrência de chuvas e velocidade do vento associada ao consumo diário de água. Neste estudo, o potencial de modelos híbridos, para a previsão diária de demanda de água, foi investigado por meio do acoplamento de séries temporais (TS) a redes neurais artificiais (TS-GRNN). Os resultados indicam que os modelos híbridos TS-GRNN fornecem melhores previsões quando comparados aos modelos de MLP ou TS isolados. Segundo os autores, os resultados indicam que a temperatura é o mais importante preditor meteorológico no treinamento da rede neural. A umidade, a velocidade do vento e a ocorrência de chuva também se mostraram importantes, mas não podem ser usadas sem temperatura. Por outro lado, a intensidade da chuva é o parâmetro que menos contribui para a capacidade do modelo de prever a demanda de água, durante o processo de treinamento das MLP.

Recentemente, Tiwari e Adamowski (2015) realizaram previsões semanais e mensais da demanda de água na cidade de Calgary (Canadá). O modelo utilizado em seu estudo foi o híbrido *wavelet* e *bootstrap* MLP (WB-MLP). O uso deste método teve como objetivo melhorar a precisão e a confiabilidade da previsão de demanda, incorporando a capacidade de processamento *wavelet* e a análise de *bootstrap* usando redes neurais artificiais. Este modelo foi então comparado ao padrão MLP, MLP baseado em reamostragem *bootstrap* (B-MLP) e modelos baseados em W-MLP. Para predição de picos semanais e mensais, os modelos híbridos WB-MLP e W-MLP mostraram-se mais precisos quando comparados aos modelos B-MLP e MLP. Os resultados das previsões usando os modelos híbridos WB-MLP e W-MLP foram muito eficazes na previsão dos picos de demanda de água. Isso indica que a análise *wavelet* melhorou significativamente o desempenho do modelo, enquanto a técnica *bootstrap* melhorou

a confiabilidade das previsões. Outro ponto destacado pelos pesquisadores é a eficácia da metodologia em situações onde a disponibilidade de dados é limitada.

Tiwari *et al.* (2016) empregaram a modelagem recém-desenvolvida *extreme learning machines* (ELM), isoladamente ou em conjunto com a análise *wavelet* ou *bootstrap*, para prever a demanda diária de água na cidade de Calgary, no Canadá. Posteriormente, os resultados foram avaliados e comparados aos obtidos com modelos baseados em uma MLP. Os modelos híbridos B-ELM e B-MLP forneceram precisão semelhante nas previsões nos dias de pico. No entanto, os modelos W-MLP e W-ELM forneceram maior precisão, com o modelo W-ELM superando o modelo W-MLP. A superioridade do modelo W-ELM sobre os modelos W-MLP ou B-MLP demonstra a importância da transformação de *wavelet* na modelagem de previsão de demanda de água urbana. Isso destaca a capacidade da transformação *wavelet* de decompor séries temporais com comportamento não-estacionário em componentes discretos, destacando os padrões e tendências cíclicos.

#### 4.2.2 Modelos Fuzzy e neuro-Fuzzy

Altunkaynak *et al.* (2005) usaram o método *Fuzzy Takagi-Sugeno* (FTS) para prever a demanda mensal de água em Istambul (Turquia). O modelo consiste em utilizar os valores de consumo de água dos últimos três meses como variáveis independentes. Ou seja, a demanda atual de água é uma função das flutuações da demanda nos últimos três meses. O método sistema de inferência *Fuzzy* neuro adaptativa (ANFIS) foi utilizado para determinar os parâmetros do modelo. A estatística do erro quadrado médio (MSE) para diferentes configurações do modelo foi usada para selecionar o mais eficaz. Os autores argumentam que este modelo é mais amplamente utilizado na previsão de demanda de água que os modelos Markov ou ARIMA, comumente disponíveis para modelagem e previsão estocástica. Uma das vantagens de usar o modelo FTS, comparado aos modelos ARIMA, é que ele não se baseia nas suposições estacionárias e de ergodicidade<sup>3</sup>. Por fim, esse modelo também ajuda a fazer previsões com menos de 10% de erro relativo.

---

<sup>3</sup> Para maiores detalhes ver Birkhoff (1931)

Firat *et al.* (2009a) compararam dois tipos de sistema de inferência *Fuzzy* (FIS) para prever as séries temporais de demanda de água urbana da cidade de Izmir, na Turquia. Os sistemas de inferência difusa usados incluem um sistema ANFIS e um sistema de inferência *Mamdani Fuzzy* (MFIS). Os desempenhos dos modelos ANFIS e MFIS foram analisados nas etapas de treinamento e teste. Para avaliar o melhor modelo de previsão, os desempenhos dos dois modelos, tanto nas etapas de treinamento como nos testes de demanda, foram comparados com os valores observados. Todos os níveis de estatísticas de limiar empregados no estudo demonstraram a maior precisão do modelo M5 em relação ao modelo M5-MFIS. O modelo M5 compreende um sistema de inferência *Fuzzy* com atrasos de cinco meses. Portanto, os resultados mostraram que o método ANFIS é superior ao método MFIS para previsão de séries de demanda mensal e pode ser aplicado com sucesso para prever o consumo de água.

#### 4.2.3 Máquinas de vetores suporte

Herrera *et al.* (2010) descreveram e compararam vários métodos para prever a demanda de água em uma cidade do sudeste da Espanha. Os modelos utilizados foram: o modelo regressão vetor suporte (SVR), FFNN empregando o método de aprendizagem retropropagação do erro, o modelo de regressão não-paramétrico *projection pursuit* (PPR), o modelo de regressão multivariada adaptativa *splines* (MARS), proposto por Friedman (1991) e o modelo floresta aleatória (FA). Além desses modelos, os pesquisadores propuseram um modelo simples baseado no perfil de demanda, usando resultados ponderados da análise exploratória de dados (WPatt). Os resultados obtidos identificaram o SVR como o modelo mais preciso, seguido de perto pelos modelos MARS, PPR e FA.

Segundo Liu *et al.* (2013), os principais fatores que influenciam os padrões de demanda de água urbana são muitas vezes difíceis de identificar, empregando algoritmos tradicionais, devido a inúmeros fatores incertos e dificuldades na quantificação. A fim de contornar este problema, os pesquisadores propuseram um algoritmo de redução de atributos melhorado baseado na média cumulativa de ponderação *C-mean Fuzzy* (FCM). Este algoritmo foi usado para analisar os principais fatores que afetam o padrão de demanda diurna de água na cidade de Hangzhou (China). Os dados utilizados neste estudo incluíram temperatura diária mínima, média e máxima, precipitação diária, dia da semana ou fim de semana e padrão de demanda de água diurna. Mais tarde, os pesquisadores usaram SVR para avaliar a influência dos principais fatores

na previsão do padrão diurno de demanda de água. O modelo de melhor desempenho incluiu a temperatura mínima e máxima diária e o dia da semana. Segundo os pesquisadores, o algoritmo FCM mostrou-se um método eficaz e viável para discretizar<sup>4</sup> espacialmente o padrão hídrico diurno de demanda de água.

Devido à dificuldade de modelar séries temporais de demanda hídrica utilizando métodos estatísticos tradicionais, Shabri e Samsudin (2015) propuseram um modelo híbrido que combina a decomposição do modo empírico (EMD), método proposto por Huang *et al.* (1998), e o método da máquina de vetor de suporte pelo mínimo quadrado (LS-SVM) para prever o consumo de água na cidade de Batu Pahat (Malásia). O EMD foi usado para decompor a série de demandas não-lineares e não-estacionárias de água em vários componentes das funções do modo intrínseco (IMF) e um componente residual. Segundo Di *et al.* (2014), a vantagem desta técnica é que ela é muito adequada para este tipo de séries temporais. O algoritmo LS-SVM foi construído para prever esses componentes intrínsecos e residuais individualmente, que foram posteriormente agregados para produzir o valor final esperado. Os resultados empíricos indicam que o modelo proposto supera o modelo LS-SVM único, a MLP, sem o pré-processamento de EMD, e o modelo EMD-MLP.

Peña-Guzmán *et al.* (2016) usaram o método LS-SVM para prever a demanda mensal de água para categorias residenciais, industriais e comerciais em Bogotá (Colômbia). Para isso, utilizaram os parâmetros de demanda mensal de água, número de usuários e preço, para as categorias residencial, industrial e comercial. A cidade emprega um sistema de estratificação socioeconômica, de acordo com as leis nacionais sobre serviços públicos, onde as residências são classificadas em seis estratos. Eles usaram registros mensais de janeiro de 2004 a dezembro de 2014. Assim como em Ghiassi *et al.* (2008), os pesquisadores empregaram 80% dos dados de treinamento e 20% dos dados de teste. O modelo LS-SVM mostrou-se superior ao modelo de redes neurais, utilizando o método da retropropagação de aprendizagem do erro, para todas as categorias e estratos. Isto provou ser uma ferramenta eficaz para o planejamento e gestão da demanda de água, pois ajudou a identificar a necessidade de decisões administrativas para regular o consumo em diferentes estratos e para diferentes usos.

Segundo Herrera *et al.* (2010), o uso de SVR é uma das melhores opções de aprendizado de máquina para previsão de demanda de curto prazo. Portanto, Brentan *et al.* (2016) propuseram

---

<sup>4</sup> Para maiores detalhes ver Liu *et al.* (2013) p.25-26

um modelo híbrido *online* aplicando SVR e *Adaptative Fourier Series* (AFS) para melhorar a previsão. O estudo foi realizado no município de Franca (Brasil), no qual foram utilizados dados sobre a demanda de água por consumidores residenciais, temperatura, umidade, precipitação e velocidade do vento. Segundo estes autores o SVR é capaz de descrever a forma ou padrão geral da demanda diária. No entanto, não é capaz de capturar bem os extremos (máximos e mínimos) e, como resultado, a precisão diminuiu nos picos de demanda. Quando comparado, o modelo híbrido obteve um  $R^2$  de 0,974 e um RMSE 1,318, enquanto o modelo SVR teve um  $R^2$  de 0,745 e um RMSE 4,767.

#### 4.2.4 Computação evolucionária

Pulido-Calvo e Gutiérrez-Estrada (2009) propuseram um modelo híbrido utilizando redes neurais, inferência *Fuzzy* e algoritmos genéticos (AG) para prever a demanda diária de água no distrito de irrigação da Andaluzia no sul da Espanha. Modelos de RNA foram treinados usando o algoritmo *Extended-Delta-Bar-Delta* (EDBD) (Pulido-Calvo *et al.*, 2003) e, subsequentemente, recalibrados com uma variação do algoritmo de retropropagação do erro conhecido como *Levenberg-Marquardt* (Shepherd, 1997). De acordo com Wilamowski e Yu (2010), o algoritmo de *Levenberg-Marquardt* é atualmente um dos mais eficientes para o treinamento de redes neurais artificiais, especialmente quando envolvem longas séries temporais. Os resultados obtidos utilizando o modelo híbrido indicaram que ele é superior aos modelos de regressão autorregressiva, univariada e multivariada, provando ser uma ferramenta muito poderosa para o desenvolvimento de políticas adequadas sobre o consumo de água de irrigação.

Um modelo híbrido que combina o filtro *Kalman* estendido (FKE) e a programação genética (PG) foi proposto por Nasser *et al.* (2011) para prever a demanda mensal de água na cidade de Teerã (Irã). A principal vantagem da abordagem proposta é a possibilidade de obter o menor número possível de formulações matemáticas não-lineares e determinísticas, para a previsão mensal da demanda de água, através do método evolutivo. Os resultados obtidos com os modelos híbridos FKE-PG e PG mostraram um efeito perceptível na precisão da previsão.

Odan (2013) implementou o método de otimização multi algoritmo geneticamente adaptativo (AMALGAM) integrado ao simulador hidráulico (EPANET 2) e uma rede neural dinâmica

(DAN2). O estudo foi realizado para o Departamento Autônomo de Água e Esgoto (DAAE) da cidade de Araraquara (Brasil). O modelo foi aplicado em três setores diferentes (Eliana, Iguatemi e Martinez) e as estratégias operacionais resultantes produziram reduções de 14%, 13% e 30%, respectivamente, no custo do consumo de eletricidade. Este método de otimização provou ser uma ferramenta robusta e eficiente.

Bai *et al.* (2014) propuseram uma abordagem de regressão vetorial de relevância multiescala (MS-RVR) para prever a demanda diária de água na cidade de Chongqing (China). Esses pesquisadores empregaram a abordagem da transformada *wavelet* estacionária para decompor a série histórica da água em diferentes escalas. Os coeficientes da *wavelet*, de todas as escalas obtidas, foram usados para treinar um modelo de aprendizado de máquina usando o método de regressão vetorial de relevância (RVR). Posteriormente, os coeficientes estimados do RVR foram utilizados para gerar resultados de previsão através da transformada inversa *wavelet*. A fim de facilitar a predição de MS-RVR, as características do caos da série diária de abastecimento de água foram analisadas usando o algoritmo de otimização adaptativa de enxame de partículas de caos (ACPSO) para determinar a combinação ótima de variáveis de entrada do modelo RVR. Finalmente, os pesquisadores compararam os resultados do melhor modelo MS-RVR com dois métodos recentes propostos por Firat *et al.* (2010), denominados GRNN e FFNN, usando o mesmo conjunto de dados e os mesmos critérios de precisão. Os resultados mostraram que o método proposto MS-RVR é mais preciso.

Romano e Kapelan (2014) apresentaram uma metodologia inovadora para prever a demanda de água por até 24 horas, visando apoiar a gestão operacional em sistemas de distribuição de água em tempo quase real. A metodologia é baseada somente na análise de séries temporais de demanda de água (estimadas por análise de balanço de massa), utilizando redes neurais artificiais evolutivas (E-FFNN). As principais características do sistema de previsão de demanda (SPD) incluem adaptabilidade contínua a mudanças nos padrões de demanda de água, aplicabilidade genérica e transparência para diferentes sinais de demanda, redução drástica no esforço de especialistas humanos necessários na projeção de um modelo de FFNN e viabilidade da implementação da metodologia em um ambiente *on-line*. O SPD consiste em quatro componentes principais: (i) o módulo de pré-processamento de dados; (ii) o módulo de otimização de FFNN; (iii) o módulo de construção da FFNN; e (iv) o módulo de previsão de demanda de água (PDA). Para o sinal de demanda específico que está sendo analisado, o módulo de pré-processamento de dados prepara os dados brutos para facilitar/melhorar o

processo de construção dos modelos E-FFNN e, assim, obter uma PDA mais precisa. O SPD permite a aplicação de duas abordagens alternativas aos sistemas de distribuição de água. O primeiro modelo (pE-FFNN) utilizou muitos modelos de E-FFNN em paralelo para prever as demandas, separadamente, para diferentes momentos do dia. O segundo modelo (rE-FFNN) usou um horizonte de previsão por hora recursivamente para prever as demandas. Ambas as abordagens foram usadas e testadas para três áreas de medidas distritais (DMA) e uma zona de abastecimento (ZA) da *Yorkshire Water Services* (YWS) cobrindo partes significativas de duas cidades no condado de *Yorkshire*, no Reino Unido. De acordo com os pesquisadores, essa nova metodologia permite gerar previsões mais precisas, demonstrando assim o potencial de proporcionar melhorias substanciais ao estado da arte no gerenciamento de sistemas inteligentes de distribuição de água, em tempo quase real. Os desempenhos das previsões foram avaliados em termos de *Nash-Sutcliffe* (NSI), erro quadrático médio (MSE) e erro percentual absoluto médio (MAPE). Os resultados mostraram que, independentemente da abordagem utilizada, os modelos múltiplos de E-FFNN superaram ligeiramente o modelo único de E-FFNN em termos de precisão na previsão da demanda de água.

#### 4.2.5 Modelos espaciais

Maria André e Carvalho (2014) estimaram a função da demanda residencial de água na cidade de Fortaleza (Brasil), considerando o impacto potencial da inclusão de efeitos espaciais na modelagem, uma vez que a exclusão desses efeitos subestima o impacto da renda, do número de banheiros e dos preços marginais da demanda residencial de água. Primeiramente, esses autores estimaram um modelo econométrico da demanda de água sem levar em conta os efeitos espaciais. Este modelo econométrico foi calculado para as seguintes especificações: preço médio (modelo AP), preço marginal com diferença (modelo MP) e preço marginal com diferença usando o método de McFadden. Posteriormente, eles calcularam três modelos para verificar a inclusão de efeitos espaciais na demanda de água: modelo de erro espacial (SEM), modelo autorregressivo espacial (SAR) e modelo espacial autorregressivo média móvel (SARMA). As seguintes variáveis explicativas foram usadas: diferença de renda média e marginal, número de residentes do sexo masculino e feminino e número de banheiros, sob diferentes especificações espaciais. Os resultados sugerem que o modelo SARMA fornece os melhores resultados. No entanto, esses resultados contradizem os achados de Chang *et al.*

(2010) e House-Peters *et al.* (2010), que afirmam que a abordagem espacial fornece resultados mais precisos do que o modelo SARMA. Após estimar o modelo SARMA (tanto para o modelo AP quanto para o modelo McFadden), e corrigindo os efeitos diretos e indiretos dos parâmetros estimados, concluiu-se que o uso de uma abordagem espacial é mais vantajoso. Não incluir efeitos espaciais nas variáveis causou uma subestimação do efeito de todas as variáveis no modelo. Depois de incluir esses componentes espaciais, a elasticidade-preço da demanda nos modelos AP e McFadden aumentou 24,66% e 13,32%, respectivamente, afetando a demanda prevista.

#### 4.2.6 Modelos estatísticos

Caiado (2010) analisou o desempenho da previsão de demanda na Espanha empregando os modelos univariados alisamento exponencial de Holt-Winters (HW), ARIMA e heterocedasticidade condicional autorregressiva generalizada (GARCH), e os modelos híbridos HW-ARIMA, HW-GARCH, ARIMA-GARCH e HW-ARIMA-GARCH para múltiplos passos à frente. Os resultados obtidos indicam que a combinação das previsões pode ser bastante útil, especialmente no tocante a previsões de curto prazo. Contudo, o desempenho desta abordagem não é consistente ao longo dos sete dias da semana. Por outro lado, as previsões individuais de modelos de HW e GARCH podem melhorar a precisão das previsões em dias específicos da semana. Por estar amplamente dominada pelos ciclos sazonais diários, semanais e anuais, modelos univariados de previsões podem ser úteis para prever a demanda d'água no curto prazo quando são retratados os diversos ciclos periódicos e sazonais na especificação da modelagem.

O objetivo principal do estudo de Fullerton Jr. *et al.* (2016) foi analisar a dinâmica da demanda de água para a cidade de El Paso no estado do Texas (EUA), empregando várias metodologias de previsão, dentre as quais a função de transferência linear (LTF), que é uma extensão da metodologia descrita por Box e Jenkins (1976). O resultado do LTF foi superior quando se comparado com os modelos de vetores autorregressivos (VAR), passeio aleatório (RW) e passeio aleatório com tendência (RWD).

Arandia *et al.* (2016) apresentaram uma metodologia de previsão de demanda de água para o curto prazo (*offline* ou *on-line*) por meio da combinação autorregressiva sazonal integrada de médias móveis (SARIMA). Segundo Caiado (2010), este modelo não tem recebido muita

atenção para previsão de demanda de água apesar de suas qualidades de parcimônia e de facilidade na interpretação dos seus parâmetros em função explícita das formulações matemáticas (Box *et al.*, 2008). Para Arandia *et al.* (2016), o modo *offline* é mais adequado para operações de serviços públicos (tais como o dimensionamento de produção diária de água), enquanto o modo *on-line* pode ser mais apropriado para outras operações (tais como agendamento de bombas). No modo *offline*, o método emprega os modelos reestimados utilizando dados históricos mais recentes. Já no modo *on-line*, o método aplica o filtro Kalman com o objetivo de atualizar e otimizar os modelos empregando “alimentação” em tempo real de dados. Foram modelados três conjuntos de dados qualitativamente diferentes, sendo identificadas estruturas e a estimativa do tamanho da amostra de dados utilizados para treino. Estes modelos foram aplicados para prever demandas com 24 horas de antecedência para os modos *offline* e *on-line*. Posteriormente, os resultados foram analisados e comparados, demonstrando a aplicação do método na previsão da produção diária de água empregando modelos SARIMA. Ao contrário dos modelos de RNA, ou outros modelos conhecidos como “caixa-preta”, a metodologia SARIMA pode ser moldada em forma de “espaço-estado”, identificando as estruturas paramétricas mais apropriadas para demandas de água com resoluções temporais variando de sub-horário a diária.

Gagliardi *et al.* (2017) propuseram um método de previsão de demanda de água a curto prazo baseado no conceito estatístico da cadeia de Markov (MC), fornecendo estimativas para demandas futuras e as probabilidades de que a demanda prevista caia dentro da variabilidade esperada. Dois modelos foram propostos, um baseado em cadeias de Markov homogêneas (HMC) e um baseado em cadeias de Markov não homogêneas (NHMC). Estes modelos foram aplicados a três áreas de medição distritais (DMA) localizadas em *Yorkshire* (Reino Unido), a fim de prever as demandas de água de 1 h a 24 h mais tarde. Posteriormente, os resultados foram comparados com as previsões dos dois modelos utilizados como *benchmarks* (RNA, *Naive Bayes*). Os resultados mostraram que o modelo HMC fornece previsões de curto prazo mais precisas do que o NHMC. Ambos os modelos forneceram informações probabilísticas sobre previsão de demanda estocástica com esforço computacional reduzido, em comparação com a maioria dos métodos existentes. Esta informação não está prontamente disponível nos modelos de *benchmark* RNA ou *Naive Bayes*. No entanto, este pode ser obtido por meio de análise de pós-processamento usando simulações de Monte Carlo que são computacionalmente mais caras.

Na Tabela 4.3 sintetizam-se as informações listadas anteriormente acerca do emprego de modelos de previsão de demanda de água.

**Tabela 4.3** – Métodos de previsão de demanda de acordo com literatura referenciada

<b>Autores</b>	<b>Cidade/País</b>	<b>Variáveis empregadas</b>	<b>Horizonte de previsão</b>	<b>Métricas utilizadas</b>	<b>Métodos empregados*</b>
Adamowski (2008)	Ottawa/Canadá	Temperatura, precipitação, demanda	Diário	R <sup>2</sup> , AARE, maxARE	<b>Rede Neural Artificial (RNA)</b> Autorregressivo Integrado de Médias Móveis (ARIMA) Regressão Linear Múltipla (RLM)
Adamowski e Karapataki (2010)	Nicosia/Chipre	Temperatura, precipitação, demanda	Semanal	R <sup>2</sup> , RMSE, AARE, maxARE, PI	<b>Rede Neural Artificial (RNA)</b> Regressão Linear Múltipla (RLM)
Adamowski <i>et al.</i> (2012)	Montreal/Canadá	Temperatura, precipitação, demanda	Diário	R <sup>2</sup> , RMSE, RRMSE, NSI	<b>Rede Neural Artificial (RNA): W-RNA</b> Autorregressivo Integrado de Médias Móveis (ARIMA) Regressão Linear Múltipla (RLM) Regressão não Linear Múltipla (RNLM)
Adamowski <i>et al.</i> (2014)	Calgary/Canadá	Temperatura, precipitação, demanda	Semanal	R <sup>2</sup> , RMSE, MAE, Pdv	Rede Neural Artificial (RNA): B-RNA, W-RNA, <b>WB-RNA</b>
Altunkaynak <i>et al.</i> (2005)	Istanbul/Turquia	Demanda	Mensal	ARE, RMSE	Autorregressivo Integrado de Médias Móveis (ARIMA) Inferência Fuzzy: ANFIS, MFIS, <b>TS-Fuzzy</b>
Al Zahrani e Abo-Monasar (2015)	Al Kobar/Arabia Saudita	Temperatura, precipitação, humidade relativa, velocidade do vento, demanda	Diário	MAPE, R <sup>2</sup>	Rede Neural Artificial (RNA): GRNN, <b>TS-GRNN</b>
Azadeh <i>et al.</i> (2012)	Tehran/Irã	Temperatura, precipitação, demanda	Diário	MAPE	Rede Neural Artificial (RNA) Regressão Linear Fuzzy (FLR)
Bai <i>et al.</i> (2014)	Chongqing/China	Demanda	Diário	MAPE, NRMSE, R	Multi-Scale Relevance Vector Regression ( <b>MS-RVR</b> ), Rede Neural Artificial (RNA): GRNN, FFNN

<b>Autores</b>	<b>Cidade/País</b>	<b>Variáveis empregadas</b>	<b>Horizonte de previsão</b>	<b>Métricas utilizadas</b>	<b>Métodos empregados*</b>
Brentan <i>et al.</i> (2016)	Franca/Brasil	Temperatura, precipitação, humidade relativa, velocidade do vento,	Diário	RMSE, MAE, R <sup>2</sup>	Regressão Vetor Suporte (SVR), Regressão Vetor Suporte Série Fourier Adaptativa (SVR-AFS)
Caiado (2010)	Espanha	Demanda	Diário	MSE	Random Walk (RW), ARIMA, Holt Winters (HW), Generalizado Autorregressivo Condicional Heterocedasticidade (GARCH), HW-ARIMA, HW-GARCH, ARIMA-GARCH, HW-ARIMA-GARCH
Firat <i>et al.</i> (2009a)	Izmir/Turquia	Demanda	Mensal	NRMSE, AARE, TS	Inferência Fuzzy: ANFIS, MFIS
Firat <i>et al.</i> (2009b)	Izmir/Turquia	População, número de domicílios, produto nacional bruto, temperatura, precipitação, humidade, inflação, demanda	Mensal	NRMSE, R, NSI	Rede Neural Artificial (RNA): GRNN, FFNN, RBN
Firat <i>et al.</i> (2010)	Izmir/Turquia	Demanda	Mensal	NRMSE, R, NSI	Rede Neural Artificial (RNA): GRNN, CCNN, FFNN
Gagliardi <i>et al.</i> (2017)	Yorkshire/Reino Unido	Demanda	Horário	NSI	MLP, NB, HMC, NHMC
Ghiassi <i>et al.</i> (2008)	San Jose e cidades circunvizinhas de Campbell, Cupertino, Los Gatos, Monte Sereno e Saratoga / USA	Volume bombeado	Horário, diário, semanal, mensal	MAPE	Rede Neural Artificial (RNA): DAN2 Autorregressivo Integrado de Médias Móveis (ARIMA)

<b>Autores</b>	<b>Cidade/País</b>	<b>Variáveis empregadas</b>	<b>Horizonte de previsão</b>	<b>Métricas utilizadas</b>	<b>Métodos empregados*</b>
Herrera <i>et al.</i> (2010)	Espanha	Temperatura, velocidade do vento, pressão atmosférica, precipitação, demanda	Horário	RMSE, MAE, NSI	Rede Neural Artificial (RNA): FFNN PPR, MARS, <b>SVR</b> , RF, WPATT, MC
Liu <i>et al.</i> (2011)	Hangzhou/China	Temperatura diária mínima, máxima e média, índice diário de condições climáticas, precipitação, dia da semana e fim de semana, demanda	Horário	AARE	SVR, FCM-SVR
Maria André e Carvalho (2014)	Fortaleza/Brasil	Preço, preço marginal, preço médio, renda, número de banheiros, área de cobertura do jardim/terra, tipo de residência, sexo	Mensal	R <sup>2</sup> , LM, Moran I	Modelos espaciais (SEM, SAR, <b>SARMA</b> )
Nasseri, <i>et al.</i> (2011)	Tehran/Irã	Demanda	Mensal	R <sup>2</sup> , NMSE	Programação Genética (GP), EKFPD
Odan e Reis (2012)	Araraquara/Brasil	Temperatura, humidade relativa, demanda	Horário	MAE, R	Rede Neural Artificial (RNA): DAN2, <b>DAN2H</b>
Peña-Guzmán <i>et al.</i> (2016)	Bogotá/Colombia	Demanda, número de usuários, preço	Mensal	R <sup>2</sup> , RMSE, AARE	Rede Neural Artificial (RNA): FFNN-BP <b>LS-SVM</b>
Pulido-Calvo e Gutiérrez-Estrada (2009)	Córdoba/Espanha	Demanda	Diário	R <sup>2</sup> , R, RMS, SEP, NSI, PI	Rede Neural Artificial (RNA): FFNN-EDBD, FFNN-LM, <b>FFNN-EDBD FUZZY</b> , FFNN-EDBD FUZZY
Romano e Kapelan (2014)	Reino Unido	Demanda	Diário	MSE, MAPE, NSI	Rede Neural Artificial (RNA): E-FFNN; rE-FFNN; <b>pE-FFNN</b>
Santos e Pereira Filho (2014)	Grande São Paulo/Brasil	Temperatura, precipitação, humidade relativa, velocidade do vento, preço, demanda	Horário, diário, semanal	MAE, ME, RMSE, R <sup>2</sup> , R,	<b>Rede Neural Artificial (RNA)</b> Regressão Linear Múltipla (RLM)

<b>Autores</b>	<b>Cidade/País</b>	<b>Variáveis empregadas</b>	<b>Horizonte de previsão</b>	<b>Métricas utilizadas</b>	<b>Métodos empregados*</b>
Shabri e Samsudin (2015)	Batu Pahat/ Malásia	Demanda	Mensal	MAE, RMSE, R	Rede Neural Artificial (RNA): RNA, EDM-RNA LS-SVM, <b>EDM-LS-SVM</b>
Tiwari e Adamowski (2015)	Calgary/Canadá	Temperatura, precipitação, demanda	Diário, mensal	R <sup>2</sup> , RMSE, MAE, PI, Pdv	Rede Neural Artificial (RNA): B-RNA, <b>W-RNA, WB-RNA</b>
Tiwari <i>et al.</i> (2016)	Calgary/Canadá	Temperatura, precipitação, demanda	Diário	R <sup>2</sup> , RMSE, MAE, PI, Pdv	Rede Neural Artificial (RNA): B-RNA, W-RNA, WB-RNA Extreme Learning Machines: ELM, <b>W-ELM</b> , B-ELM

\* Métodos em negrito obtiveram as melhores performances (previsões mais robustas)

**Fonte:** Elaborado pelo autor

#### 4.2.7 Discussão acerca da revisão bibliográfica dos modelos de previsão de demanda

A revisão de literatura ilustra, no contexto da previsão da demanda de água, que vários métodos de *soft computing* (SC) foram estudados e aplicados para lidar com problemas de precisão, estocasticidade e não-linearidade (BONISSONE, 1997) presentes na série histórica de demanda de água. Isso faz com que a previsão da demanda de água seja uma tarefa muito árdua. Nesse sentido, os métodos lógica *Fuzzy*, computação neural, algoritmos genéticos, computação evolutiva, aprendizado de máquina e sistemas híbridos, que empregam tal complementaridade, contribuíram muito para os avanços metodológicos na previsão da demanda de água urbana. Cada um desses métodos fornece ferramentas adicionais, procurando resolver problemas complexos do mundo real.

No entanto, um olhar mais atento à literatura revisada e aos recentes avanços nos métodos de SC sugerem que ainda há espaço para melhorias em relação à previsão da demanda de água.

As redes neurais podem ser categorizadas em vários tipos. No entanto, existem poucos métodos usados para prever a demanda de água (GHALEHKhONDABI *et al.*, 2017). Uma extensa revisão da literatura revelou que apenas um pequeno número de arquiteturas foi empregado para prever a demanda de água urbana. A arquitetura mais utilizada na previsão de demanda de água urbana é o FFNN, também conhecido como MLP. Referências ao modelo RBNN, rede neural probabilística (PNN), entre outras também foram identificadas. No entanto, não foram encontrados estudos que utilizem a arquitetura de redes neurais recorrentes (RNN) como redes Elman, Jordan, Hopfield, *feedback global*, *echo state* (ESNN), *output feedback* e rede totalmente recorrente (FRNN).

Ainda, em relação às redes neurais, o número de camadas ocultas e os algoritmos utilizados no treinamento podem afetar o desempenho da RNA. Encontrar a melhor arquitetura pode ser uma tarefa árdua, e a FFNN pode nem sempre ser o melhor método ou fornecer os melhores resultados (HERRERA *et al.*, 2010). Outros tipos de algoritmos, como o ELM, foram pesquisados e desenvolvidos para otimizar as previsões baseadas em redes neurais (TIWARI *et al.*, 2016).

Esta revisão de literatura mostra que os estudos são mais focados no sistema operacional de gestão (curto prazo), de acordo com a classificação proposta por Donkor *et al.* (2014). Existem poucos estudos que abordam a previsão de médio e de longo prazo. Uma possível explicação

pode estar associada à inadequação da arquitetura básica da RNA, como a FFNN, para lidar com dados ruidosos (GHALEHKHONDABI *et al.*, 2017), limitando sua aplicação com padrões pouco complexos e linearmente inseparáveis. Existem também certos tipos de padrões nas séries temporais de demanda de água, que exigem uma grande necessidade de pré-processamento.

A fim de melhorar a precisão da previsão, os pesquisadores começaram a desenvolver modelos híbridos baseados no acoplamento de *wavelet* e *bootstrap*. Esses modelos foram mais robustos na previsão, comparados aos modelos RLM, RNLM e ARIMA. Isto indica que a transformada *wavelet* melhorou significativamente o desempenho dos modelos, destacando a *wavelet* de capacidade de processamento na decomposição de comportamento discreto e não estacionário de componentes de séries temporais (ADAMOWSKI, 2008; GHIASSI *et al.*, 2008), enquanto a técnica *bootstrap* melhorou a confiabilidade das previsões, sugerindo um potencial promissor deste método híbrido para prever a demanda de água urbana.

Outra linha de pesquisa que tem grande potencial de aplicação na previsão de demanda de água é a computação evolucionária. Poucos estudos foram desenvolvidos usando programação genética, como os trabalhos de Pulido-Calvo e Gutiérrez-Estrada (2009) e Nasser *et al.* (2011). Odan (2013) utilizou o método de otimização AMALGAM e, mais recentemente, Bai *et al.* (2014) utilizaram o ACPSO. Outras metaheurísticas, como o método de otimização colônia de formigas (ACO), procedimento ágil de pesquisa adaptativa aleatória (GRASP) e algoritmo genético de recozimento simulado (GSAA), também poderiam ser empregados.

Os avanços mais recentes de redes neurais, como as redes neurais convolucionais (CNN ou *ConvNet*), ainda não foram utilizados na previsão de demanda de água urbana, abrindo um espaço promissor a esse problema. Recentemente, Borovykh *et al.* (2017) desenvolveram uma rede neural profunda convolutiva para a previsão de séries temporais multivariadas com base na arquitetura recente *WaveNet*, desenvolvida por Oord *et al.* (2016). Ainda, segundo Qiu *et al.* (2014), os métodos de aprendizado profundo demonstraram ser altamente promissores em diversas áreas de previsão. Fawaz *et al.* (2019) estudaram o desempenho atual de última geração de algoritmos de aprendizado profundo para classificação de séries temporais (TSC), apresentando um estudo empírico das arquiteturas de redes neurais profundas mais recentes para TSC.

Outra arquitetura, proposta por Vaswani *et al.* (2017), conhecida como Transformer, baseia-se unicamente em mecanismos de atenção, dispensando totalmente as recorrências e convoluções. Os resultados obtidos por estes modelos são de qualidade superior, ao mesmo tempo em que são mais paralelizáveis e requerem significativamente menos tempo para treinar (VASWANI *et al.*, 2017). Empregando esta nova arquitetura Li e Moura (2019) propuseram o Forecaster para enfrentar os desafios dos dados espaciais e dependentes do tempo. Essa tarefa é difícil devido à sua dependência espacial complexa, dependência temporal de longo alcance, não estacionariedade e heterogeneidade dos dados. Ainda, segundo Li e Moura (2019), esta arquitetura proposta supera significativamente as linhas de base de última geração.

Segundo Yang e Alessandrini (2019), os análogos (previsão usando padrões semelhantes do passado) foram amplamente adotados pelas comunidades de meteorologia e energia renovável para aplicações preditivas. A correspondência de padrões exigida pelo conjunto analógico se beneficia de conjuntos de dados históricos cada vez maiores. Contudo, pesquisas analógicas de força bruta se tornam impraticáveis com conjuntos de dados de treinamento maiores. Visando superar esse desafio, estes autores apresentam um método rápido para encontrar análogos. Esse método é conhecido como algoritmo de Mueen para pesquisa de similaridade (MASS). O MASS tem inúmeras propriedades desejáveis, pois é exato, não paramétrico, escalável, paralelizável e, principalmente, livre da maldição da dimensionalidade (o CC é independente do comprimento da consulta). Verificou-se que o MASS é cerca de 100 vezes mais rápido que o algoritmo de força bruta, sendo adequado para todas as tarefas de correspondência de padrões com base na distância euclidiana. Outro trabalho que baseia-se em previsão analógica foi proposto por Chattopadhyay *et al.* (2019). Este emprega uma nova técnica de reconhecimento de padrões de aprendizado profundo (redes neurais em cápsulas, CapsNets) e estratégia de rotulagem automática baseada em impacto.

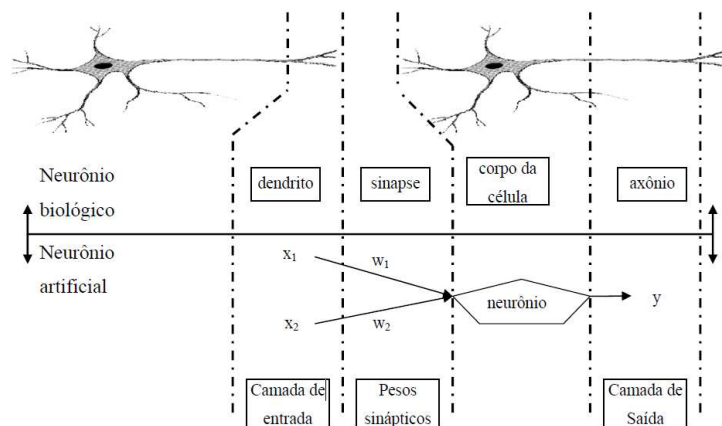
## **4.3 Redes Neurais**

### **4.3.1 Conceitos gerais**

As redes neurais artificiais (RNA) são modelos computacionais inspirados no comportamento fisiológico do neurônio. O primeiro passo significativo no desenvolvimento das redes neurais começa com o trabalho do psiquiatra e neuroanatomista Warren McCulloch e do matemático

Walter Pitt que, em 1943, publicaram o primeiro artigo sobre propriedades matemáticas das redes neurais intitulado: *A Logical Calculus of the Ideas Immanent in Nervous Activity*. As RNA são modelos matemáticos baseados no funcionamento do cérebro humano com capacidade para reconhecer, classificar, converter e aprender padrões.

Na figura 4.2 pode-se observar a comparação entre neurônio biológico e artificial, onde são realizados os seguintes cotejos: dendrito com a camada de entrada; a sinapse com os pesos sinápticos; o corpo da célula com o neurônio artificial; e o axônio com a camada de saída.



**Figura 4.2** – Comparação entre neurônio biológico e artificial (ODAN, 2013)

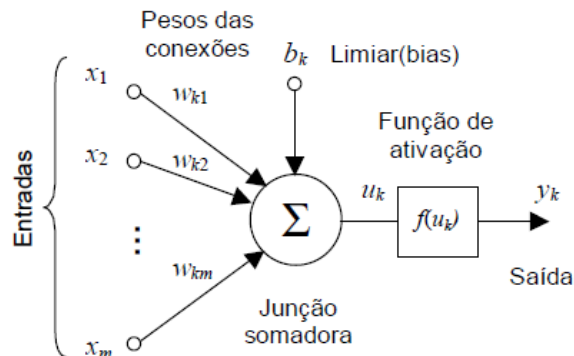
Segundo Haykin (2001), as RNA apresentam as seguintes propriedades e capacidades: não-linearidade; mapeamento de entrada e saída; adaptabilidade; tolerância a falhas; e capacidade de generalização.

Em virtude de sua versatilidade, essa modelagem é empregada em diversas áreas do conhecimento tais como o reconhecimento de padrões, a otimização e a previsão de séries temporais (BRAGA *et al.*, 1998).

A principal característica do neurônio artificial diz respeito ao aprendizado que ocorre através da experiência (HAYKIN, 2001). Outra característica, também de grande importância, é a força de conexão entre os neurônios, conhecida como pesos sinápticos, empregadas para armazenar o conhecimento adquirido. O processo de aprendizado consiste em calibrar os pesos sinápticos

e o viés<sup>5</sup>, de forma que a aplicação de um conjunto de entradas produza um conjunto de saídas desejadas. Este processo, na qual a rede é treinada, é denominado aprendizado supervisionado.

O neurônio artificial, representado pela Figura 4.3, é composto por três elementos básicos, quais sejam, os pesos das conexões ou pesos sinápticos, a junção somadora ou função aditiva e a função de ativação. Cada sinapse é definida como sendo um estímulo de entrada multiplicado pelo seu peso sináptico correspondente.



**Figura 4.3** – Modelo geral do neurônio artificial (Adaptado de HAYKIN, 2001)

em que:

$x_1, x_2, \dots, x_m$  são os sinais de entradas do neurônio (números reais ou binários);

$w_{k1}, w_{k2}, \dots, w_{km}$  são os pesos sinápticos (valores positivos ou negativos) que são os parâmetros livres a serem ajustados na fase de aprendizado;

$u_k$  é o resultado da função aditiva dos sinais das entradas, ponderados pelas respectivas sinapses;

$f(u_k)$  é a função de ativação. Esta função é um limitador de amplitude de um neurônio de saída e é geralmente fechada em um intervalo de  $[0,1]$  ou alternativamente  $[-1,1]$ ;

---

<sup>5</sup> Segundo Silva (2003), viés é uma entrada adicional de polarização aplicada a um dado neurônio  $k$  aumentando ou diminuindo o argumento da função de ativação, caso seja positivo ou negativo, respectivamente.

$b_k$  (viés) é o viés, que possui o efeito de aumentar ou diminuir a entrada da função de ativação, dependendo se o sinal é positivo ou negativo; e

$y_k$  é o sinal de saída do neurônio.

Em termos matemáticos, um neurônio  $k$  poderá ser escrito como:

$$u_k = \sum_{j=1}^m w_{kj} x_j + b_k \quad (4.3.1.1)$$

A função será calculada através da combinação linear dos pesos sinápticos e das entradas ou, equivalentemente, ao produto escalar do vetor de pesos  $w^T = [w_{k1}, w_{k2}, \dots, w_{km}]$  e do vetor de entradas  $x = [x_1, x_2, \dots, x_m]$ .

$$u_k = w^T x + b_k \quad (4.3.1.2)$$

A saída será dada aplicando-se uma função de ativação

$$y_k = f(u_k) = f\left(\sum_{j=1}^m w_{kj} x_j + b_k\right) \quad (4.3.1.3)$$

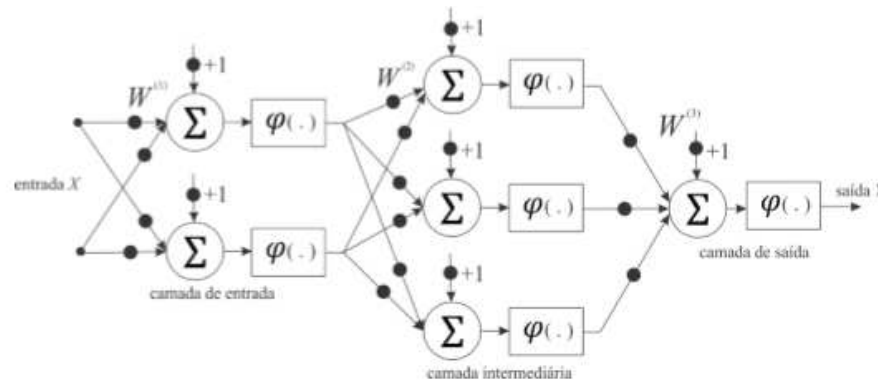
A função de ativação do neurônio, também conhecida como função restritiva, tem o propósito proporcionar uma função contínua derivável em todo domínio em um intervalo de  $[0,1]$  ou alternativamente  $[-1,1]$ . As funções de ativação podem ser lineares ou não-lineares, tais como função limiar ou degrau, função linear, função gaussiana, função sigmoide logística e tangente hiperbólica e função unidade linear retificadora mais conhecida como ReLU. As funções sigmoide são as mais comumente usadas na construção de RNA do tipo MLP, por serem contínuas, crescentes, diferenciáveis e não-lineares.

#### 4.3.2 Rede neural *Multilayer Perceptron* (MLP)

A rede *Multilayer Perceptron* é uma das principais arquiteturas de redes neurais, a qual utiliza diversos neurônios de McCulloch-Pitts dispostos em camadas. São amplamente usadas na

resolução de problemas complexos, especialmente no tocante à previsão, devido a sua capacidade de mapeamento de entrada-saída desejada.

Uma rede MLP pode ser composta por várias camadas. Na Figura 4.4 apresenta-se uma rede MLP com três camadas. A primeira camada é a camada onde a informação externa é recebida, a camada intermediária, também denominada de camada oculta, é onde ocorre o processamento e, por fim, a camada de saída é onde a solução do problema é obtida.



**Figura 4.4** – Rede neural *Multilayer Perceptron* (MLP) (Adaptado de HAYKIN, 2001)

A rede neural apresentada acima é do tipo *feedforward*, ou seja, o sinal de entrada  $X$  é combinado com os pesos das conexões e propagado através da rede. A saída de cada uma das camadas de entrada se torna a entrada da camada intermediária até gerar uma saída  $Y$ . Ao projetar uma MLP, é necessário determinar o número de nós de entrada, o número de camadas escondidas e seus respectivos nós ocultos e o número de nós de saída.

Segundo Cybenko (1989) uma rede, com uma camada intermediária, será capaz de implementar qualquer função contínua.

### 4.3.3 Regra de aprendizagem da RNA

A solução de uma tarefa, empregando RNA, passa por uma fase de aprendizagem. Essa etapa consiste em um processo de ajuste iterativo dos pesos sinápticos das conexões, denominado de algoritmo de treinamento, podendo ser classificado como não supervisionado ou supervisionado.

Um dos algoritmos mais utilizados para o treinamento de uma rede MLP é conhecido como algoritmo de retropropagação de erro (*error backpropagation*), desenvolvido por Rumelhart *et al.* (1986). Embora as equações possam ser derivadas, o algoritmo padrão de retropropagação não garante a convergência para a solução ideal, uma vez que a superfície de erro para redes multicamadas com funções de ativação não-linear não é tão simétrica e regular como a de um modelo de neurônio *Adaptive Linear Element* (ADALINE)<sup>6</sup>. Este algoritmo, descrito por Widrow e Hoff (1960), é conhecido como regra delta de aprendizado e tem uma grande importância na área de RNA, já que foi ele que deu origem ao algoritmo *backpropagation* para treinamento de MLP de múltiplas camadas. Este método utiliza o gradiente descendente no ajuste dos pesos sinápticos da rede baseado na minimização do erro quadrático médio.

Um problema relacionado ao algoritmo de retropropagação de erro diz respeito à obtenção da solução ideal ao chegar em regiões de mínimos locais. A forma irregular da superfície do erro é, de fato, a dificuldade principal no treinamento de redes multicamadas com funções de ativação não-lineares. As não-linearidades podem resultar em regiões onde o gradiente é nulo, levando o treinamento a ficar preso sem atingir a solução ideal. Visando superar essa dificuldade no treinamento de redes multicamadas e melhorar a velocidade de convergência, vários algoritmos foram desenvolvidos, tais como *Quick Propagation* (QuickProp) que busca aproximar a superfície do erro por uma parábola em função dos pesos de forma que o erro mínimo da parábola seja alcançado (FAHLMAN, 1988), *Resilient Propagation* (RPROP) que emprega o sinal do gradiente e não o seu valor para realizar a correção dos pesos (REIDMILLER e BRAUN, 1993), *Back-Propagation com Momentum* (VOGL *et al.*, 1988; JACOBS, 1988; FAHLMAN, 1989; SILVA e ALMEIDA, 1990; PHANSALKAR e SASTRY, 1994; HAGIWARA e SATO, 1995; KAMARTHI e PITTNER, 1999) e algoritmos de taxa de aprendizagem adaptativa (YU e CHEN, 1997). Os métodos de taxa adaptativa utilizam técnicas para o ajuste da taxa de aprendizado. Contudo, nenhum destes métodos garante uma convergência ao mínimo global.

---

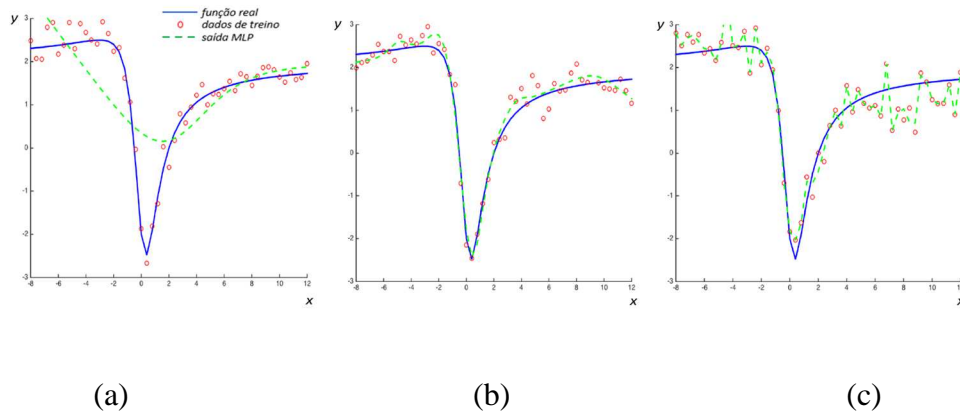
<sup>6</sup> Widrow e Hoff desenvolveram uma importante variação do algoritmo de aprendizado *Perceptron* por meio da aplicação do conceito de erro médio quadrático.

#### 4.3.4 Capacidade de generalização

A capacidade de generalização é a principal aptidão buscada nas tarefas que envolvem o aprendizado da máquina. Essa habilidade de generalizar refere-se à capacidade de um sistema de aprendizagem (a partir do aprendizado obtido após o treinamento) em conseguir apresentar saídas corretas para novas entradas (novas experiências) ainda não apresentadas ao modelo (BISHOP, 2006).

Um fator de grande importância na capacidade de generalização diz respeito à definição do número de neurônios a serem utilizados na RNA, sendo que essa quantidade decorrerá do problema a ser estudado. Esse fator deve ser escolhido com muito cuidado, pois influencia diretamente na capacidade de aprendizagem e de generalização da rede. As situações indesejadas podem ser geradas pela escolha de uma quantidade reduzida ou excessiva de neurônios, gerando os problemas de subajuste (*underfitting*) e superajuste (*overfitting*), que estão diretamente ligados à capacidade de generalização do modelo proposto.

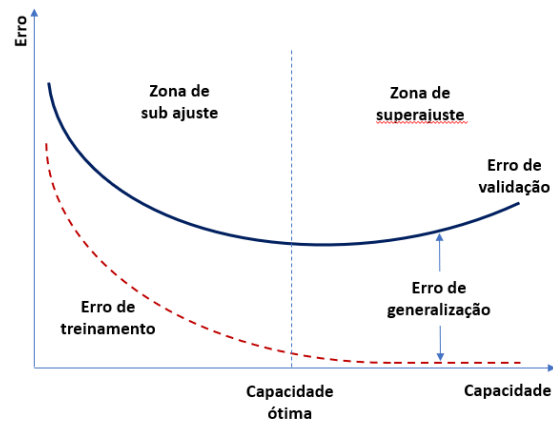
A Figura 4.5 apresenta exemplos de subajuste (a), uma boa generalização (b) e uma situação de superajuste (c) para um problema dado pela função  $Y = \frac{(X-2)(2X+1)}{1+X^2}$ .



**Figura 4.5** - (a)  $\hat{f}_1(x)$  é uma MLP com dois neurônios na camada escondida; (b)  $\hat{f}_2(x)$  é uma MLP com seis neurônios na camada escondida; e (c)  $\hat{f}_3(x)$  é uma MLP com 20 neurônios na camada escondida. (Elaborado pelo autor)

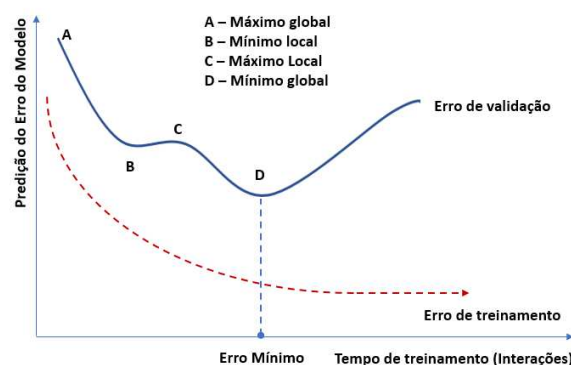
Outro fator diz respeito ao dilema entre a capacidade de generalização da rede e a minimização do erro, apresentado na Figura 4.6. Observa-se, nesta figura, que o erro de teste, utilizado como uma aproximação da capacidade de generalização, decresce com o erro de treinamento somente

até um certo ponto, a partir do qual menores erros de treinamento implicarão em maiores erros de teste (MACKAY, 2003). Quando uma rede aprende mais do que deveria durante o período de treinamento, perderá a sua capacidade de generalização. Para evitar o superajuste, deve-se treinar a rede até que o erro de validação atinja o seu mínimo. Neste ponto, o erro de treinamento não é mínimo, entretanto é o menor possível para os padrões de validação.



**Figura 4.6** – Dilema entre a capacidade de generalização da rede e a minimização do erro. (Elaborado pelo autor)

A curva do erro de treinamento tem um comportamento monotônico, isto é, à medida que as iterações progredem o erro do conjunto de treinamento cai. Contudo, a curva de validação nem sempre terá um comportamento monotonicamente decrescente até o seu ponto de mínimo, podendo existir mínimos locais, dificultando o melhor momento de interrupção, como apresentado na Figura 4.7.



**Figura 4.7** – Representação do dilema entre capacidade de generalização da rede e a minimização do erro. (Elaborado pelo autor)

#### 4.3.5 Parada antecipada (*early stopping*)<sup>7</sup>

Uma das dificuldades do uso de RNA consiste em identificar o melhor ponto de parada de treinamento. A parada antecipada do treinamento de uma rede antes do seu término, denominada parada antecipada (WEIGEND *et al.*, 1990; PRECHELT, 1998), tem como objetivo obter redes com boa capacidade de generalização. O método consiste em acompanhar a evolução do aprendizado nas curvas correspondentes aos subconjuntos de dados de treinamento e de validação.

No processo de treinamento de redes MLP no qual os dados apresentam ruídos, pode-se notar que, à medida que o número de iterações aumenta, o erro de validação decresce até um ponto de mínimo e, a partir deste ponto, começa a crescer novamente apesar do erro de treinamento continuar decrescendo, como ilustrado pela Figura 4.6. Este processo pode ser dividido em duas partes. A primeira parte vai do início do treinamento até o ponto de mínimo da curva de erro de validação. Nesta parte a rede se adapta somente às principais características dos dados, aprendendo a função geradora. Já a segunda fase, na medida em que o tempo de treinamento avança, o ruído também começa a ser modelado.

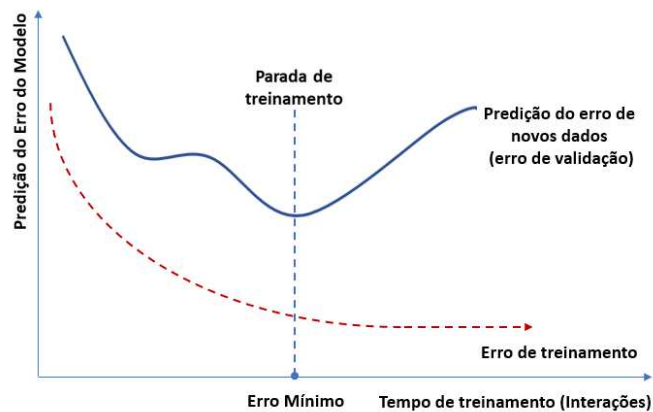
Na Figura 4.8 apresenta-se um exemplo no qual será determinada a parada do treinamento de uma MLP usando os erros do grupo de validação.

Para identificar o ponto de parada de aprendizado, buscando obter a melhor generalização da rede neural, uma alternativa será utilizar a técnica da regra de parada antecipada com base na validação cruzada (HECHT-NIELSEN, 1990; SARLE, 1995). De acordo com Goutte (1997) a validação cruzada é melhor do que a parada antecipada, principalmente nos casos de pequeno conjunto de dados.

O critério exato utilizado para a validação cruzada com base em uma parada precoce, no entanto, é escolhido de forma “*ad hoc*” pela maioria dos pesquisadores ou o treinamento é interrompido interativamente (PRECHELT, 1998).

---

<sup>7</sup> Para maiores detalhes ver PRECHELET, L. Automatic early stopping using cross validation: quantifying the criteria. **Appeared in Neural Network**, 1998.



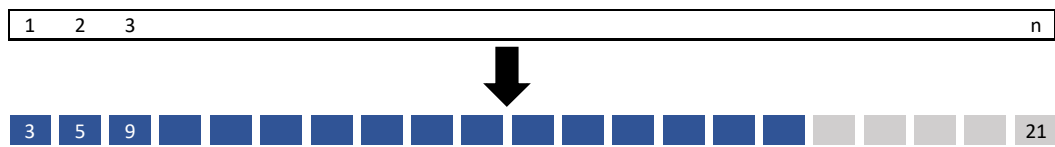
**Figura 4.8** – Parada antecipada do treinamento de uma rede MLP. (Elaborado pelo autor)

#### 4.3.6 Métodos de separação de amostras para fazer a validação cruzada

A história do desenvolvimento da validação cruzada está documentada em Stone (1974). A sua formalização se dá pela partição da base de dados em subconjuntos mutuamente exclusivos (dados de treino e de teste), e posteriormente empregados para avaliar a capacidade de predição. Numa primeira etapa um determinado algoritmo de indução é aplicado a base de treinamento, obtendo um modelo “treinado”, que representa, de certa forma, o conhecimento extraído. Posteriormente, numa segunda etapa, o modelo obtido é aplicado ao subconjunto da base de teste.

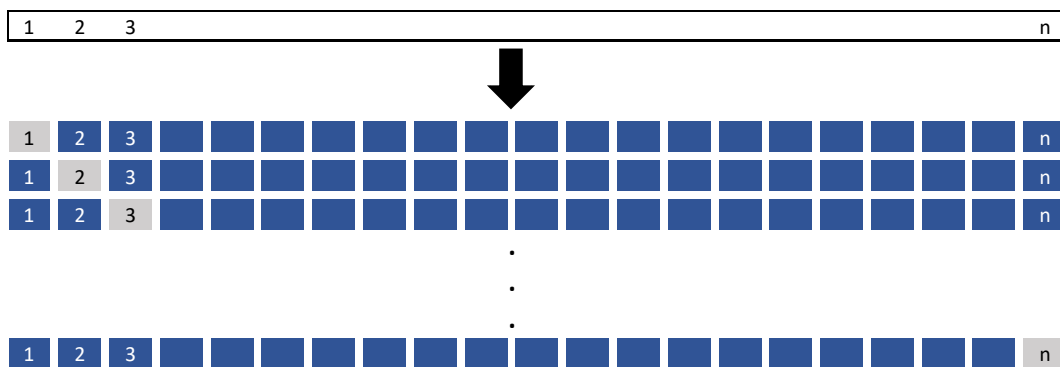
A validação cruzada é um dos métodos mais utilizados para avaliar a generalização de algoritmos em classificação e regressão (HASTIE *et al.*, 2009). Os padrões mais amplamente utilizados são:

- ✓ *Holdout Cross Validation* (HCV) – as observações são selecionadas aleatoriamente para cada uma das partições (treinamento e validação).



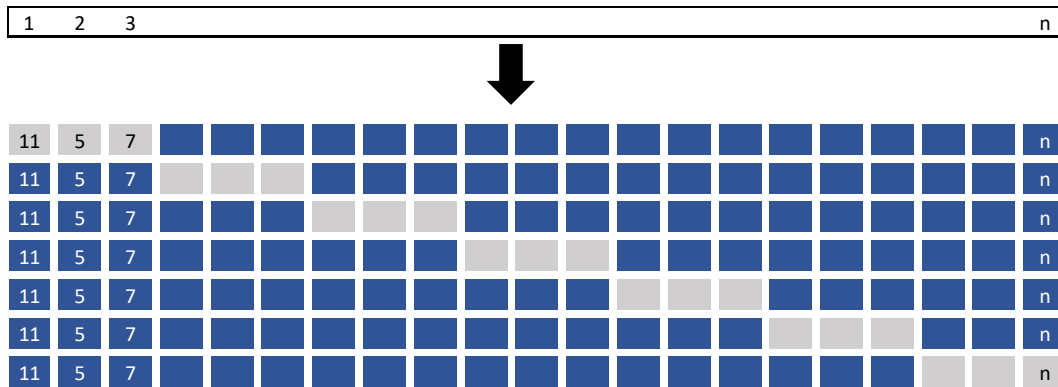
**Figura 4.9** - Exibição esquemática do *Holdout Cross Validation*. Um conjunto de  $n$  observações são divididas aleatoriamente em um conjunto de treinamento (em azul, contendo as observações 3, 5 e 9, entre outras) e um conjunto de validação (em cinza e contendo a observação 21, entre outras). O método de aprendizado estatístico é adequado ao conjunto de treinamento e seu desempenho é avaliado no conjunto de validação (Adaptado de JAMES *et al.*, 2013).

- ✓ *Leave-One-Out Cross Validation (LOOCV)* – dado um conjunto de dados contendo  $n$  observações, o conjunto de treinamento emprega  $n - 1$  observações e o conjunto de validação emprega uma observação.



**Figura 4.10** - Exibição esquemática do LOOCV. Um conjunto de  $n$  pontos de dados é repetidamente dividido em um conjunto de treinamento (em azul) contendo todas, exceto uma observação, e um conjunto de validação que contém apenas essa observação (em cinza). O teste do erro é então estimado pela média dos  $n$  MSE resultantes. O primeiro conjunto de treinamento contém todas as observações menos a primeira, o segundo conjunto de treinamento contém todas as observações menos a segunda e assim por diante (Adaptado de JAMES *et al.*, 2013).

- ✓ *K-Fold Cross Validation (K-fold)* – método que consiste em dividir as  $n$  observações em  $k$  subconjuntos mutuamente exclusivos do mesmo tamanho.



**Figura 4.11** - Exibição esquemática do *k-fold*. Um conjunto de  $n$  observações é dividido aleatoriamente em  $k$  grupos mutuamente exclusivos. Cada um desses  $1/k$  atua como um conjunto de validação (em cinza) e o restante como um conjunto de treinamento (em azul). O erro de teste é estimado pela média do MSE das  $k$  estimativas resultantes (Adaptado de JAMES *et al.*, 2013).

Contudo, quando se trata de previsão em séries temporais, devido à correlação serial inerente e a potencial não estacionariedade da série fazem com que o uso da validação cruzada pareça problemático, uma vez que não leva em conta esses problemas (BERGMEIR e BENÍTEZ, 2012). Ainda, segundo estes autores, o problema das dependências dentro do treinamento e do conjunto de testes pode ser resolvido usando blocos de dados em vez de escolher os dados de forma aleatória. Na previsão tradicional o bloco final da série, reservado para teste, é denominado de avaliação *out-of-sample* (fora da amostra) (TASHMAN, 2000).

Hyndman (2014), recomenda o uso da validação cruzada em séries temporais sempre que possível. Nesta abordagem, o autor propõe usar vários conjuntos de treinamento diferentes, cada um contendo uma observação a mais que a anterior. A Figura 4.12 mostra-se uma série com conjunto de treino (em azul) e outro de teste (em cinza) e as medidas de precisão da previsão são calculadas em cada conjunto de teste.



**Figura 4.12** – Validação cruzada de séries temporais, onde cada conjunto de treino (azul) contém uma observação a mais que o conjunto anterior, e conseqüentemente cada conjunto de teste (cinza) tem menos uma observação que o anterior (Adaptado de HYNDMAN, 2014).

Uma variação dessa abordagem, e que será empregada no presente estudo, se concentra em um único horizonte de previsão para cada conjunto de teste. De acordo com Tashman (2000), esta abordagem é denominada de *rolling origin recalibration* (recalibração de origem rolante). Fildes *et al.* (1998) fornecem evidências de que a recalibração de pesos, ao ajustar um método de suavização exponencial, melhora a precisão fora da amostra do método. Ainda, segundo Tashman (2000), a recalibração de origem rolante é o procedimento preferível. Nesse procedimento, as previsões são executadas movendo sequencialmente os valores de testes para o conjunto de treinamento e alterando a origem da previsão. Para cada previsão o modelo será recalibrado usando todos os dados disponíveis no conjunto de treinamento, significando um treinamento completo do modelo (BERGMEIR e BENÍTEZ, 2012). Makridakis *et al.* (2018), em suas conclusões, também sugerem o uso do enfoque de janela rolante para se obter o máximo de informações possíveis sobre os valores futuros e a incerteza resultante e aprender com mais eficiência.

Na Figura 4.13 ilustra-se uma série com conjuntos de teste que contêm apenas um passo empregando a recalibração de origem rolante.



**Figura 4.13** - Validação cruzada de séries temporais com base em previsões de um passo. Os pontos azuis representam os conjuntos de treino, o cinza de teste e os pontos em branco são ignorados (Adaptado de HYNDMAN, 2014).

Em qualquer uma dessas abordagens de validação cruzada, precisa-se de um tamanho mínimo para o conjunto de treinamento porque muitas vezes não é possível fazer qualquer previsão significativa se não houver dados suficientes para estimar o modelo escolhido. O tamanho mínimo do conjunto de treinamento depende da complexidade do modelo que deseja usar (HYNDMAN, 2014).

Suponha que  $k$  observações são necessárias para produzir uma previsão confiável  $h$ -passos à frente. O procedimento é o seguinte:

1. Para  $i = 1, 2, \dots, T - k - h + 1$ , em que  $T$  é o número total de observações, seleccionar a observação no tempo  $k + h + i - 1$  para o conjunto de teste, e usar as observações do tempo  $1, 2, \dots, k + i - 1$  para estimar o modelo de previsão. Calcular o erro  $h$ -passos da previsão para o tempo  $k + h + i - 1$ ; e
2. Calcular as medidas de precisão das previsões com base nos erros obtidos.

Quando  $h = 1$ , isto dá o mesmo procedimento tal como descrito acima.

Outra técnica de validação cruzada, que pode ser empregada em séries temporais, é a janela rolante. Assim, como na recalibração de origem rolante, essa técnica é treinada novamente para cada conjunto de teste. Entretanto, o tamanho do conjunto de treinamento é fixo de modo que há uma poda nos valores mais antigos à medida que novos dados se tornam disponíveis. O emprego desta técnica é especialmente útil quando os dados não são estacionários (BERGMEIR, 2013). Na Figura 4.14 ilustra-se uma série com conjuntos de teste que contêm apenas um passo empregando a janela rolante.



**Figura 4.14** – Validação cruzada usando a janela rolante (Adaptado de BERGMEIR, 2013)

### 4.3.7 Dilema viés-variância<sup>8</sup>

O problema de aprendizagem é construir uma função  $f(x)$  que se baseia no conjunto de dados  $\{(x_1, y_1) \dots (x_n, y_n)\}$ , ou seja, uma função  $f(x)$  que aproxima a resposta desejada de  $y$ . Geralmente  $f$  é escolhido para minimizar alguma função de custo.

$$\sum_{i=1}^N [y_i - f(x_i)]^2 \quad (4.3.7.1)$$

$$\left[ (Y - \hat{f}(X))^2 \mid X = x_i \right] = E \left[ (Y - f(X) + f(X) - \hat{f}(X))^2 \right] \quad (4.3.7.2)$$

$$\begin{aligned} \left[ (Y - \hat{f}(X))^2 \mid X = x_i \right] \\ = E \left[ (Y - f(X))^2 \right] + E \left[ (f(X) - \hat{f}(X))^2 \right] \\ + 2E \left[ (Y - f(X)) (f(X) - \hat{f}(X)) \right] \end{aligned} \quad (4.3.7.3)$$

$$E \left[ (Y - \hat{f}(X))^2 \mid X = x_i \right] = E[(\epsilon)^2] + E \left[ (f(X) - \hat{f}(X))^2 \right] \quad (4.3.7.4)$$

$$E \left[ (Y - \hat{f}(X))^2 \mid X = x_i \right] = Var(\epsilon) + E \left[ (f(X) - \hat{f}(X))^2 \right] \quad (4.3.7.5)$$

A parcela  $E \left[ (f(X) - \hat{f}(X))^2 \right]$  é o erro redutível. Surge porque  $\hat{f}(X)$  não será uma estimativa perfeita para  $f(X)$ . Pode ser reduzido pois sempre há possibilidade de se melhorar a acurácia de  $\hat{f}(X)$  empregando o método de aprendizagem de máquina mais apropriado. Contudo, a parcela do erro irreduzível  $Var(\epsilon)$  pode conter variáveis que não foram medidas e que seriam úteis na predição de  $Y$ , além de contar com variações não mensuráveis nos dados, uma vez que, para um dado  $X = x$  pode existir uma infinidade de possíveis valores de  $y$ .

$$E \left[ (f(X) - \hat{f}(X))^2 \right] = E \left[ (f(X) - E[\hat{f}(X)] + E[\hat{f}(X)] - \hat{f}(X))^2 \right] \quad (4.3.7.6)$$

---

<sup>8</sup> Para maiores detalhes ver GEMAN, S.; BIENENSTOCK, E.; DOURSAT, R. Neural network and the Bias/Variance Dilemma. **Neural Computation**, v. 4, p. 1-58, 1992.

$$\begin{aligned}
E \left[ \left( f(X) - \hat{f}(X) \right)^2 \right] \\
&= E \left[ \left( f(X) - E[\hat{f}(X)] \right)^2 \right] + E \left[ \left( E[\hat{f}(X)] - \hat{f}(X) \right)^2 \right] \\
&\quad + 2E \left[ \left( f(X) - E[\hat{f}(X)] \right) \left( E[\hat{f}(X)] - \hat{f}(X) \right) \right]
\end{aligned} \tag{4.3.7.7}$$

$$\left[ \left( f(X) - \hat{f}(X) \right)^2 \right] = E \left[ \left( f(X) - E[\hat{f}(X)] \right)^2 \right] + E \left[ \left( E[\hat{f}(X)] - \hat{f}(X) \right)^2 \right] \tag{4.3.7.8}$$

O valor esperado do erro  $MSE_{Teste}$ , para uma dada observação  $x_0$ , pode ser decomposto na soma de três erros fundamentais:

$$\begin{aligned}
E \left[ \left( Y - \hat{f}(X) \right)^2 \right] \\
&= E \left[ \left( f(X) - E[\hat{f}(X)] \right)^2 \right] + E \left[ \left( E[\hat{f}(X)] - \hat{f}(X) \right)^2 \right] + Var(\epsilon)
\end{aligned} \tag{4.3.7.9}$$

Segundo a decomposição viés-variância, para a amostra de teste, o erro quadrático esperado em um ponto qualquer  $X$   $\left( E \left[ \left( Y - \hat{f}(X) \right)^2 \right] \right)$  sempre pode ser decomposto em (i)  $Var(\epsilon)$  é a variância do erro do modelo ( $\sigma_\epsilon^2$ ); (ii)  $E \left[ \left( f(X) - E[\hat{f}(X)] \right)^2 \right]$  é o quadrado do viés do modelo no ponto ( $viés^2(\hat{f})$ ); e (iii)  $E \left[ \left( E[\hat{f}(X)] - \hat{f}(X) \right)^2 \right]$  é a variância do modelo  $var(\hat{f})$ .

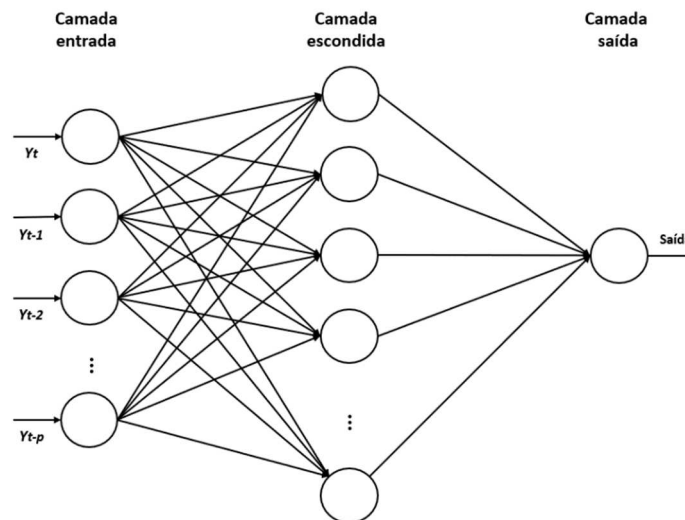
O viés está relacionado à habilidade do modelo em se ajustar aos dados, ou seja, se o seu problema é um subajuste, o seu modelo tem um viés alto. Já a variância está relacionada à habilidade do modelo se ajustar a novos dados, ou seja, se o seu problema é um superajuste, o seu modelo tem uma alta variância.

#### 4.3.8 Rede neural autorregressiva não-linear (NNAR)

Segundo Ruiz *et al.* (2016), na maioria dos casos as aplicações de séries temporais são caracterizadas por altas variações e períodos transitórios, dificultando a modelagem de séries temporais utilizando um modelo linear. Portanto, sugere a aplicação de uma rede neural autorregressiva não-linear (NNAR), assim como (IBRAHIM *et al.*, 2016 e TEALAB, *et al.*,

2017) para a previsão de séries temporais. A NNAR é uma rede *feedforward*, e é uma aproximação não-linear  $h(\cdot)$ .

A topologia de uma rede NNAR está descrita na Figura 4.15 a seguir.



**Figura 4.15** – Rede neural autorregressiva não-linear (NNAR) (Elaborado pelo autor)

A rede NNAR é usada para prever o valor de uma série de dados  $y$  no tempo  $t$ ,  $y_t$ . As variáveis defasadas  $y_{t-1}, y_{t-2}, \dots, y_{t-p}$  são denominadas de termos autorregressivos. A função  $h(\cdot)$  é previamente desconhecida, e o treinamento da rede neural visa aproximar a função por meio da otimização dos pesos sinápticos da rede e do viés. Finalmente, o termo  $\varepsilon_t$  representa o erro da aproximação da série  $y$  no tempo  $t$ .

O número de camadas ocultas e de neurônios por camada é completamente flexível, sendo otimizado por meio de um procedimento de tentativa e erro para obter a topologia de rede que pode fornecer o melhor desempenho, isto é, o menor erro quadrático médio (MSE). Contudo, como apresentado na seção (4.3.4), é importante ter em mente que o aumento do número de neurônios pode gerar superajuste, enquanto um número baixo de neurônios pode gerar o problema de subajuste. Ambas as situações estão ligadas à capacidade de generalização da rede.

Neste estudo será empregada a função “*nnetar*” presente no pacote<sup>9</sup> *Forecast*. Esta função ajusta um modelo de rede neural *NNAR* ( $p, P, k$ ) $m$  a uma série temporal com valores defasados das séries temporais como entradas. Portanto, é um modelo autorregressivo não-linear em que  $p$  indica o número de variáveis defasadas (camada de entrada da rede),  $P$  e  $m$  são as ordens de sazonalidade e  $k$  são o número de nós da camada oculta. Se os valores de  $p$  e  $P$  não forem especificados, serão selecionados automaticamente. No caso de a série temporal apresentar comportamento sazonal, os valores propostos serão  $P = 1$  e  $p$  será escolhido a partir do modelo linear ótimo ajustado aos dados dessazonalizados. Se  $k$  não for especificado, será definido como  $k = (p + P + 1)/2$ . Esta função também emprega uma transformação Box-Cox<sup>10</sup> (1964) para estabilizar ou reduzir a variabilidade existente e normalizar os resíduos. Empregou-se um  $\lambda = 0,5$  como forma de garantir que os resíduos sejam aproximadamente homocedásticos. O argumento *npaths* no *forecast.nnetar* controla quantas simulações serão realizadas (*default* são 1000). Por padrão, os erros são extraídos de uma distribuição normal e o argumento de *bootstrap* permite que os erros sejam reamostrados (isso é gerados aleatoriamente a partir dos erros históricos).

Exemplificando, um modelo *NNAR* (3,1,2)[12] tem entradas  $y_{t-1}, y_{t-2}, y_{t-3}$  e  $y_{t-12}$ , e dois neurônios na camada oculta. Mas geralmente, um modelo de *NNAR* ( $p, P, k$ ) $m$  tem entradas  $(y_{t-1}, y_{t-2}, \dots, y_{t-p}, y_{t-m}, y_{t-2m}, y_{t-pm})$  e  $k$  neurônios na camada oculta. O modelo *NNAR* ( $p, P, 0$ ) $m$  é equivalente a um modelo *ARIMA* ( $p, 0, 0$ ) $\times$ ( $P, 0, 0$ ) $s$ , mas sem as restrições dos parâmetros para garantir a estacionariedade da série.

## 4.4 Extreme learning machine (ELM)

### 4.4.1 Conceitos gerais

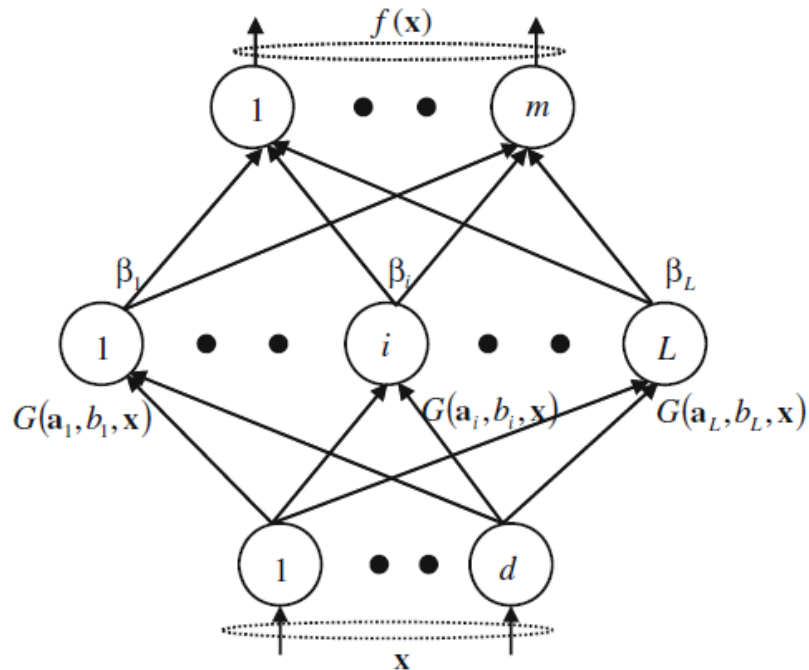
O *Extreme Learning Machine* (ELM), proposto por Huang *et al.* (2006), é uma arquitetura de rede *feedforward* semelhante à MLP com somente uma camada escondida, denominada *single-hidden layer feedforward network* (SLFN). Entretanto, ela se diferencia pelo processo de

---

<sup>9</sup> Pacote: consiste num conjunto de funções que quando chamadas executam rotinas de programação. (R Development Core Team, 2017)

<sup>10</sup> Transformações Box Cox:  $y^{(\lambda)} = \begin{cases} \frac{y^\lambda - 1}{\lambda}, & \lambda \neq 0 \\ \log y, & \lambda = 0 \end{cases}$

treinamento, uma vez que os pesos de entrada e as polarizações dos vieses da camada oculta são escolhidas de maneira aleatória e independente. A Figura 4.16 apresenta uma rede SLFN.



**Figura 4.16** – Single-hidden layer feedforward network (SLFN) (HUANG *et al.*, 2011)

A aleatoriedade da camada interna traz implícita a idéia de desorganização, visto que parte da rede permanece sem ajuste, remetendo ao trabalho de Turing (1968). Essa arquitetura pertence ao grupo de redes neurais conhecida como máquinas desorganizadas (BOCCATO *et al.*, 2011).

Embora as RNA tenham vantagens, como capacidade de aproximação e estruturas de rede simples, no entanto, sofrem de vários problemas, como presença de mínimos locais, taxa de aprendizado impreciso, seleção do número de neurônios ocultos e sobreposição. Além disso, os algoritmos de aprendizado baseados no gradiente descendente, como a retropropagação, possuem uma convergência lenta durante a aprendizagem das redes (CHEN *et al.*, 2012; CAO e LIN, 2015; ALBARD e TIUN, 2017), podendo levar horas ou dias, dependendo do tamanho e frequência do conjunto de treinamento. Segundo Ding *et al.* (2015), quando se compara o ELM com as técnicas tradicionais de inteligência computacional, como por exemplo a retropropagação, a velocidade de treinamento melhora em mais de dez vezes.

Diferentemente dos algoritmos de aprendizagem tradicionais, o ELM não só tende a atingir o menor erro de treinamento, mas também a menor norma de pesos da saída (HUANG *et al.*, 2011).

Segundo Huang *et al.* (2015), o desenho original da ELM, do ponto de vista da eficiência de aprendizagem, possui três objetivos: alta precisão de aprendizagem, menos intervenção humana e rápida velocidade de aprendizado.

O algoritmo ELM se destaca devido ao seu custo computacional reduzido e sua essência está no uso de pesos sinápticos aleatórios e independentes na camada escondida (HUANG *et al.*, 2004; 2006). Portanto, o processo de ajuste não adapta os pesos desta camada, mas somente da camada de saída. Os valores ótimos dos pesos são tipicamente calculados de forma analítica, já que o treinamento envolve a solução de um problema de regressão linear (HUANG *et al.*, 2006). Dessa forma, não há necessidade de cálculos de derivadas, retropropagação de sinais dos erros ou o emprego de algoritmos iterativos (HAYKIN, 2001), reduzindo o custo computacional.

A capacidade de aprendizagem de uma ELM foi estudada segundo dois aspectos: capacidade de interpolação (HUANG *et al.*, 2006) e capacidade de aproximação universal (HUANG *et al.*, 2006; HUANG e CHEN, 2008).

Segundo Huang *et al.* (2011), a ELM possui a capacidade de aproximador universal, uma vez que o erro de aproximação pode ser sempre reduzido com a inclusão de um novo neurônio na camada oculta através da determinação rigorosa do novo peso da camada de saída. Dessa maneira, se o número de amostras for igual ao número de neurônios, o erro de saída pode ser zero, provando, assim, que as ELM são aproximadores universais tais quais a MLP (HORNIK *et al.*, 1989; HORNIK, 1991) e a RBNN (PARK e SANDBERG, 1991).

Devido a sua eficiência, simplicidade e desempenho de generalização, o ELM tem sido aplicado em diversas áreas do conhecimento como engenharia biomédica, reconhecimento facial, processamento de imagem, modelagem e previsão de sistemas, classificação e compreensão de texto, detecção e diagnóstico de falhas, sensoriamento remoto, controle e robótica e realização de previsões (HUANG *et al.*, 2015; DING *et al.*, 2015; ALBARD e TIUN, 2017). Nesse contexto, Singh e Balasundaram (2007) estudaram a aplicação do algoritmo ELM para problemas de previsão de séries temporais caóticas. Ainda, segundo estes autores, os resultados

obtidos estão em estreita concordância com a solução exata dos problemas considerados que mostram claramente que o ELM é um método muito promissor para a previsão de séries temporais.

A formulação ELM leva a resolução de um sistema de equações lineares em termos dos pesos desconhecidos que conectam a camada oculta à camada de saída. A solução deste sistema geral de equações lineares é obtida empregando o operador generalizado de Moore-Penrose<sup>11</sup> (RAO e MITRA, 1971), que possui a capacidade de minimizar o erro quadrático médio entre o sinal que chega na camada de saída e o desejado. Além disto, existem evidências teóricas que apontam um aumento da capacidade de generalização quando o vetor de pesos possui norma mínima (BARTLETT, 1998). A conclusão que Bartlett chega é que o controle da norma dos pesos sinápticos é mais relevante em termos de capacidade de generalização do que o controle do número de neurônios da camada intermediária.

#### 4.4.2 Algoritmo *extreme learning machine*

Considere uma SLFN com um número  $L$  de neurônios ocultos. Seja  $G(\cdot, \cdot, \cdot)$  uma função de valor real, de modo que  $G(a_i, b_i, x)$  seja a saída do  $i$ -ésimo neurônio oculto com termo de viés  $b_i \in \mathbb{R}$  correspondente ao vetor de entrada  $x \in \mathbb{R}^d$  e ao vetor de peso  $a_i = (a_{i1}, \dots, a_{im})'$ , na qual  $a_i$  é o vetor de pesos que liga o  $i$ -ésimo neurônio oculto à entrada da rede. É conhecido que, para as redes neurais do tipo *feedforward*, a função de saída  $f(\cdot)$  é representada por

$$f_L(x) = \sum_{i=1}^L \beta_i g_i(x) = \sum_{i=1}^L \beta_i G(a_i, b_i, x), \quad (4.4.2.1)$$

$$x \in \mathbb{R}^d, \beta_i \in \mathbb{R}^m$$

em que  $\beta_i = (\beta_{i1}, \dots, \beta_{im})' \in \mathbb{R}^m$  é o vetor de pesos que conecta o  $i$ -ésimo neurônio escondido com os neurônios da camada de saída e  $g_i$  denota a função  $G(a_i, b_i, x)$  do  $i$ -ésimo nó oculto. Para o nó oculto aditivo com a função de ativação  $g$ ,  $g_i$  terá a seguinte forma

---

<sup>11</sup> Maiores detalhes no Anexo I

$$g_i = G(a_i, b_i, x) = g(a_i' \cdot x + b_i) \quad (4.4.2.2)$$

em que  $g: \mathbb{R} \rightarrow \mathbb{R}$  será a função de ativação que é infinitamente diferenciável em qualquer intervalo.

Suponha que tenhamos um conjunto de dados de treinamento  $\{(x_i, t_i) | x_i \in \mathbb{R}^d, t_i \in \mathbb{R}^m, i = 1, \dots, N\}$  onde  $x_i = (x_{i1}, \dots, x_{in})'$  denota o vetor de entrada e  $t_i = (t_{i1}, \dots, t_{im})'$  é o vetor de saída correspondente e  $N$  é o total de padrões dos dados de entrada. Suponha ainda que os valores dos vetores de peso  $a_i \in \mathbb{R}^d$  e os termos de viés  $b_i \in \mathbb{R}$  são atribuídos aleatoriamente. Então, o SLFN padrão com número  $L$  de neurônios ocultos aproxima as amostras de entrada com erro zero se e somente se houver  $\beta_i \in \mathbb{R}^m$  de modo que:

$$t_j = \sum_{i=1}^L \beta_i G(a_i, b_i, x_j) \quad \forall j = 1, 2, \dots, N \quad (4.4.2.3)$$

O conjunto de equações acima pode ser reescrito na seguinte forma de matriz como:

$$H\beta = T \quad (4.4.2.4)$$

em que:

$$H(a_1, \dots, a_L, b_1, \dots, b_L, x_1, \dots, x_N) = \begin{bmatrix} g(a_1 \cdot x_1 + b_1) & \cdots & g(a_L \cdot x_1 + b_L) \\ \vdots & \cdots & \vdots \\ g(a_1 \cdot x_N + b_1) & \cdots & g(a_L \cdot x_N + b_L) \end{bmatrix}_{N \times L},$$

$$\beta = \begin{bmatrix} \beta_1' \\ \vdots \\ \beta_L' \end{bmatrix}_{L \times m} \quad \text{e} \quad T = \begin{bmatrix} t_1' \\ \vdots \\ t_N' \end{bmatrix}_{N \times m}$$

Nota-se que a  $i$ -ésima coluna de  $H$  será a saída do  $i$ -ésimo neurônio para as entradas  $x_1, \dots, x_N$ . Além disso, observa-se que a matriz  $H$  não precisa ser uma matriz quadrada.

Sob o pressuposto de que a função de ativação  $g(\cdot)$  é infinitamente diferenciável, Huang *et al.* (2006) demonstram que para vetores de peso de entrada fixos  $a_i$  e viés  $b_i$ , a solução de mínimos quadrados  $\hat{\beta}$  para a equação da matriz ( $H\beta = T$ ) com a norma mínima dos pesos de saída  $\beta$

podem ser obtidos e que o menor erro de treinamento pode ser alcançado pela solução  $\hat{\beta}$ . Além disso, a solução  $\hat{\beta}$  da equação da matriz ( $H\beta = T$ ) será dada por:

$$\hat{\beta} = H^{\dagger}T \quad (4.4.2.5)$$

em que  $H^{\dagger}$  é a pseudo inversa generalizada de Moore-Penrose da matriz  $H$ .

O treinamento de uma SLFN é equivalente à obtenção de uma solução de mínimos quadrados mínima da equação matricial  $H\beta = T$ . No decorrer da aprendizagem, uma vez que os pesos de entrada e os vieses da camada oculta são escolhidos aleatoriamente, não serão ajustados. Pelo **Teorema 1**, apresentado no Anexo A.I.1, a menor solução de mínimos quadrados da norma da máquina de aprendizado acima é obtida quando  $\hat{\beta} = H^{\dagger}T$

O algoritmo ELM para SLFN pode ser indicado (HUANG *et al.*, 2006) da seguinte forma:

**Entrada:** Dado um conjunto de treinamento  $\{(x_i, t_i) | x_i \in \mathbb{R}^d, t_i \in \mathbb{R}^m, i = 1, \dots, N\}$  no qual  $L$  é o número de neurônios ocultos e a função de ativação é dada por  $g(\cdot)$ .

1. Atribui aleatoriamente o peso do vetor de entrada  $a_i$  e o viés  $b_i, i = 1, \dots, L$
2. Determine a matriz  $H$
3. Calcule  $H^{\dagger}$
4. Calcule a matriz de pesos de saída  $\hat{\beta} = H^{\dagger}T$

**Saída:** A SLFN com os vetores de peso de saída determinados  $\hat{\beta}_i \in \mathbb{R}^m$  para os vetores de peso escolhidos aleatoriamente  $a_i \in \mathbb{R}^m$  e vieses  $b_i \in \mathbb{R}$  para  $i = 1, 2, \dots, L$ .

Para qualquer amostra de entrada  $x \in \mathbb{R}^d$  o valor de saída  $t$  pode ser calculado usando a seguinte equação (4.4.2.6):

$$t = \sum_{i=1}^L \hat{\beta}_i g(a_i' \cdot x + b_i) \quad (4.4.2.6)$$

em que  $a_i$ ,  $b_i$  e a função de ativação  $g(\cdot)$  são entradas e os vetores de peso  $\widehat{\beta}_t \in \mathbb{R}^m$  são a saída do algoritmo ELM.

Segundo Rao e Mitra (1971), existem diversos métodos para calcular a pseudo inversa Moore-Penrose generalizada, dentre os quais tem-se o método de projeção ortogonal. Segundo Huang *et al.* (2012), este método pode ser eficientemente usado em ELM:  $H^\dagger = (H'H)^{-1}H'$  se  $H'H$  é não singular ou  $H^\dagger = H'(HH')^{-1}$  se  $HH'$  é não singular.

Se  $(n + 1) \leq N$  e  $H'H$  é não singular, então  $H^\dagger = (H'H)^{-1}H'$

Se  $(n + 1) \geq N$  e  $HH'$  é não singular, então  $H^\dagger = H'(HH')^{-1}$

Inúmeras pesquisas, no que tange à regularização dos pesos de saída, foram e continuam sendo realizadas (DENG *et al.*, 2009; MARTÍNEZ-MARTÍNEZ *et al.*, 2011; KULAIIF, 2009; MAO *et al.*, 2015). A determinação do número de neurônios, que também pode ser considerado como um meio de regularização, foi estudado por Miche *et al.* (2010), Miche *et al.* (2011), Kulaif, (2009).

No presente estudo as ELM serão ajustadas usando o pacote *nnfor* do software R (KOURENTZES, 2017), pacote este que tem como base o estudo de Crone e Kourentzes (2010) e Kourentzes *et al.* (2014).

#### 4.5 Seleção de modelos

Para que a generalização do modelo não sofra nem de superajuste nem de subajuste é necessário que haja um equilíbrio entre o viés e a variância. Existem algumas abordagens propostas na literatura como a validação cruzada, descrita no tópico (4.3.5), e a regularização. A regularização consiste em penalizar o erro dentro da amostra por meio de um termo de penalidade. Uma abordagem proposta para calcular o termo de penalidade (*overfit penalty*) é o critério de informação de Akaike (AIC) (AKAIKE, 1969).

A regularização ou parâmetro de suavização  $\lambda$  desempenha papel similar ao número de pesos de uma rede neural FFNN. Pequenos valores de  $\lambda$  produzem estimadores de baixa variância e alto viés. Os dados são essencialmente ignorados em favor da restrição (“*oversmoothing*”).

Grandes valores de  $\lambda$  produzem interpolações do tipo *splines* podendo ter alta variância. Parâmetros de suavização geralmente fornecem um grau de regularidade (viés), controlando assim a variância. Um método automático de suavização e o mais utilizado é a validação cruzada. Métodos de regularização relacionados com uma função de penalidade que é adicionada à soma dos erros quadrados observados, promovem o alisamento, ou parcimônia ou outro mapeamento regular (GEMAN *et al.*,1992).

#### **4.6 Seleção de variáveis e/ou atributos (filtros, invólucros e embarcado)**

Os métodos de seleção automática de variáveis/atributos são importantes, pois selecionam o conjunto de parâmetros mais significativos para a resolução de um dado problema, sendo uma importante ferramenta na redução do custo computacional (PIRAMUTHU, 2004), além de melhorar a eficácia do processo ocasionado pela mitigação da conhecida maldição da dimensionalidade (GUYON e ELISSEEFF, 2003).

De acordo com Saeys *et al.* (2007), os principais objetivos da seleção de atributos são: (i) evitar superajuste e melhorar o desempenho do modelo, (ii) fornecer modelos mais rápidos e econômicos e (iii) obter uma visão mais profunda dos processos subjacentes que geraram os dados. Portanto, o objetivo será escolher estatisticamente um subconjunto mínimo de atributos a partir do conjunto original das variáveis de entrada do processo (KOHAVI e JOHN, 1997).

Para Siqueira (2013), a seleção busca identificar um conjunto de entradas que venham auxiliar a previsão, o reconhecimento de padrões ou mesmo a regressão de dados, objetivando a melhora dos resultados do modelo. Este problema de seleção envolve encontrar um conjunto de atributos adequado sob alguma função objetivo. Segundo John (1994) as funções objetivos mais comuns são precisão de previsão, tamanho da estrutura e uso mínimo de recursos de entrada.

As técnicas de seleção de atributos podem ser organizadas em três categorias, dependendo de como elas combinam a pesquisa na seleção dos atributos com a construção do modelo: métodos de filtro, métodos de *wrapper* (invólucros) e métodos *embedded* (embarcado). Estes métodos têm seus respectivos tipos de busca, critério e avaliação dos subconjuntos de variáveis. Na Tabela 4.4 ilustram-se as principais características de cada um destes métodos.

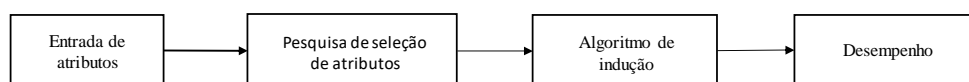
As vantagens das técnicas de seleção de atributos têm um determinado preço, já que a pesquisa por subconjuntos de atributos relevantes introduz uma camada adicional de complexidade na tarefa de modelagem.

**Tabela 4.4** – Tipos de busca, critério e avaliação do filtro, invólucro e embarcado dos subconjuntos de variáveis.

Tipos	Busca	Critério	Avaliação
Filtro	Ordena variáveis e subconjuntos	Mérito variável e subconjunto	Testes estatísticos
Invólucro	Espaço de subconjuntos	Avalia subconjuntos	Validação cruzada
Embarcado	Guiada pelo aprendizado	Avalia subconjuntos	Validação cruzada

**Fonte:** Freitas (2007)

No modelo de filtro a seleção de subconjuntos é feita como uma etapa de pré-processamento dos dados, independente da escolha do algoritmo de indução (Figura 4.17). O pré-processamento de dados pode ter um efeito significativo no desempenho do modelo (MAIER e DANDY, 2000). Por não dependerem diretamente do algoritmo, são mais rápidos. Este filtro faz o ordenamento das variáveis individualmente ou em subconjuntos, independentemente do preditor, e geralmente é robusto a superajuste, mas falha em encontrar o subconjunto de variáveis mais promissor (FREITAS, 2007). A principal desvantagem desta abordagem é que ela ignora totalmente os efeitos dos subconjuntos selecionados ao desempenho do algoritmo de indução (JOHN *et al.*, 1994; KOHAVI e JOHN, 1997).



**Figura 4.17** - Abordagem filtro no qual a seleção de atributos é independente do algoritmo de indução. (Elaborado pelo autor)

Alguns tipos de filtros são empregados na literatura tais como: (i) regressão *stepwise* utilizada para identificar os atrasos significativos do componente autorregressivo da variável dependente como entradas de uma MLP (DAHL e HYLLEBERG, 2004; ALVES *et al.*, 2013); (ii) análise espectral empregada para avaliar os padrões cíclicos dos dados, decompondo a série de tempo

em funções subjacentes de seno e cosseno dos comprimentos das ondas (KAY e MARPLE, 1981); (iii) transformada *wavelet* decompõe a série original em componentes de alta e baixa frequência (ADAMOWSKI e CHAN, 2011; ADAMOWSKI *et al.*, 2012; CAMPISI-PINTO *et al.*, 2012; TIWARI e ADAMOWSKI, 2013; NOURANI *et al.*, 2014; ADAMOWSKI *et al.*, 2014; HUANG *et al.*, 2014; TIWARI e ADAMOWSKI, 2015; TIWARI *et al.*, 2016, ALTUNKAYNAK e NIGUSSIE, 2017; CHEN *et al.*, 2017, DU *et al.*, 2017); (iv) transformada de Fourier para decompor o sinal em suas componentes elementares seno e cosseno (ZHOU *et al.*, 2000; BOUGADIS *et al.*, 2005; PARTAL e KISI, 2007; BRENTAN *et al.*, 2016); (v) filtro Kalman<sup>12</sup> (POLI e JONES, 1994; NASSERI *et al.*, 2011; ARANDIA *et al.*, 2016; SANTHUSITHA *et al.*, 2018; KARUNASINGHA e LIONG, 2018); (vi) análise de espectro singular (HASSANI, 2007; DU *et al.*, 2017); (vii) decomposição STL (BERGMEIER *et al.*, 2016); e (viii) decomposição em modo empírico<sup>13</sup> (EMD) é uma abordagem de resolução de frequência de tempo pela qual o comportamento estacionário e não-linear da série temporal pode ser decomposta (DI *et al.*, 2014; SHABRI e SAMSUDIN, 2015).

A metodologia invólucro, popularizada por John *et al.* (1994) e Kohavi e John (1997), oferece uma maneira simples e poderosa de abordar o problema da seleção de variáveis e/ou atributos, independentemente da máquina de aprendizagem empregada. Em sua formulação mais geral, este método avalia a seleção dos subconjuntos de atributos usando o próprio algoritmo de indução como parte da função de avaliação. Segundo John *et al.* (1994), para se determinar um subconjunto útil de atributos, o algoritmo de seleção de subconjunto deverá levar em consideração os vieses do algoritmo de indução, a fim de selecionar um subconjunto que acabará gerando em uma estrutura induzida com alta precisão preditiva em dados de teste. John *et al.* (1994) empregaram a validação cruzada *n*-fold (BREIMAN *et al.*, 1984), na qual os dados de treinamento foram divididos em *n* partições de tamanho aproximadamente igual. Posteriormente, o algoritmo de indução foi executado *n* vezes usando *n* – 1 partições como conjunto de treino e a outra parte como conjunto de teste. Os resultados de acurácia de cada uma das *n* iterações são então calculadas para produzir a acurácia estimada.

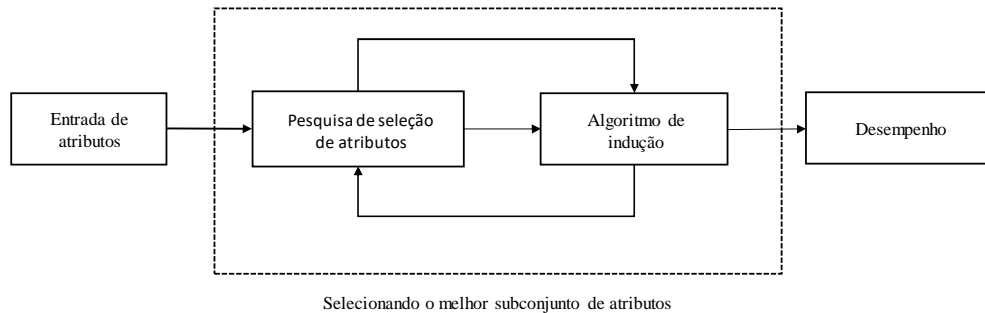
---

<sup>12</sup> Segundo Santhusitha *et al.* (2018), o filtro proposto por Kalman (1960) é a ferramenta mais usada para estimar os estados do sistema a partir de medições ruidosas. A extensão deste filtro para sistemas não lineares é denominado Filtro Kalman Estendido e a sua derivação pode ser encontrada em Welch e Bishop (2004).

<sup>13</sup> A decomposição em modo empírico (EMD) é uma parte integral da transformada de Hilbert-Huang (HHT) desenvolvida por Huang *et al.* (1998) e é amplamente utilizada para detectar mudanças nas escalas de variabilidade e para lidar com dados não-lineares e não-estacionários.

As avaliações de desempenho dos invólucros geralmente são realizadas empregando um conjunto de validação ou por meio de validação cruzada (GUYON e ELISSEEFF, 2003).

A Figura 4.18 ilustra o modelo invólucro, para qual o algoritmo de indução é usado como um “caixa preta” pelo algoritmo de seleção de atributos.



**Figura 4.18** - Abordagem do invólucro para seleção de atributos. (Elaborado pelo autor)

As principais vantagens deste método são: levar em conta o viés do algoritmo de indução e considerar as variáveis dentro do contexto. Os principais tipos de busca são: exponencial, randômica (estocástica) e sequencial. A ordenação exponencial é a técnica pela qual os indivíduos são ordenados em função das suas aptidões, sendo atribuído a cada indivíduo um valor que corresponde à sua posição ordenada. Esta busca é do tipo exaustiva, garantindo uma solução ótima, mas é mais complexa computacionalmente. Segundo Freitas (2007), a princípio a busca é exponencial, mas pode-se implementar buscas estocásticas (ex. algoritmos genéticos<sup>14</sup> e recozimento simulado<sup>15</sup>) ou sequenciais (ex. busca direta, eliminação para trás<sup>16</sup>, bi-direcional, seleção flutuante e a primeira melhor busca). Segundo Uncu e Türksen (2007), os algoritmos sequenciais de busca direta e de eliminação para trás são largamente utilizados. Contudo, apesar dos métodos sequenciais serem amplamente empregados, devido a facilidade

<sup>14</sup> O algoritmo genético é uma meta-heurística inspirada no processo de seleção natural, motivado por uma analogia à evolução biológica. Estes algoritmos em vez de pesquisar hipóteses gerais e específicas, ou de simples a complexas, geram hipóteses sucessoras, mutando e recombinao repetidamente partes das melhores hipóteses atualmente conhecidas (MITCHELL, 1997).

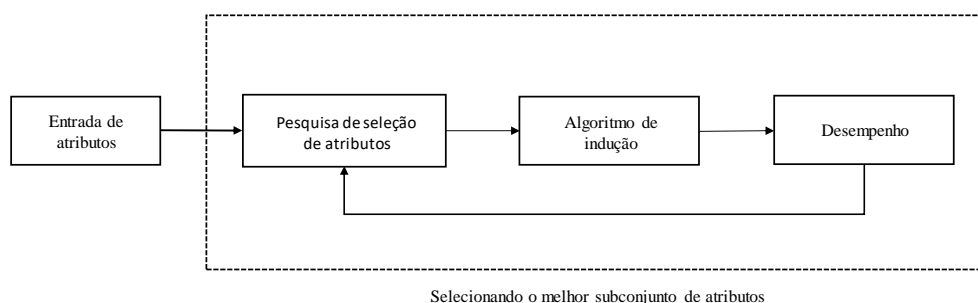
<sup>15</sup> O recozimento simulado (*simulated annealing*) é uma meta-heurística para otimização que consiste em uma técnica de busca local probabilística e surgiu no contexto da mecânica estatística, desenvolvida por Kirkpatrick *et al.* (1983) e independentemente por Cerny (1985). Esta técnica é utilizada em problemas de otimização combinatória.

<sup>16</sup> A performance da eliminação para trás é ligeiramente superior à busca direta por considerar as variáveis no contexto (FREITAS, 2007).

de implementação e de sua atratividade computacional, o algoritmo pode ficar preso em mínimos locais. Uma solução para este problema é o uso de métodos randômicos (BLUM e LANGLEY, 1997).

Segundo Guyon e Elisseeff (2003), os invólucros costumam ser criticados porque parecem ser um método de "força bruta" requerendo um custo computacional elevado, contudo não é necessariamente assim. Segundo estes autores, estratégias eficientes de busca podem ser planejadas sem sacrificar, necessariamente, o desempenho da previsão. Diversas estratégias de seleção estão sendo desenvolvidas para minimizar a carga computacional ocasionada pela busca exaustiva, como o trabalho de Sorjamaa *et al.* (2007) e o Hsu *et al.* (2011).

Por fim, será abordado o modelo embarcado/embutido, para o qual a tarefa de seleção de subconjuntos de atributos é realizada internamente pelo próprio algoritmo de aprendizado de máquina, de forma dinâmica, enquanto procuram por uma hipótese, conforme ilustrada na Figura 4.19, utilizando mecanismos semelhantes à metodologia invólucro. A diferença entre o invólucro e o embarcado está na busca que é guiada pelo processo de aprendizado, tornando menos propenso a superajuste, mais atraente computacionalmente, mas é mais complexo (FREITAS, 2007). Nestes algoritmos o processo de seleção não é claramente diferenciável do treinamento do modelo e os resultados obtidos são “calibrados” em relação a um dado classificador ou regressor específico. Por esse motivo, na maioria dos casos, o atributo selecionado por um método embarcado pode não ser adequado para outros. Para Zanchettin (2008) uma das vantagens desta abordagem é fazer o melhor uso dos dados que estão disponíveis, não sendo necessário a divisão dos dados de treinamento, e ser mais rápida por não realizarem múltiplos treinamentos.



**Figura 4.19** – Abordagem embutida ou embarcada para seleção de atributos. (Elaborado pelo autor)

Os métodos embarcados combinam as vantagens dos filtros e dos invólucros fazendo uso total do algoritmo de seleção, portanto, sua desvantagem é determinar o critério de seleção e o algoritmo de pesquisa com antecedência (MALDONADO e LOPEZ, 2015).

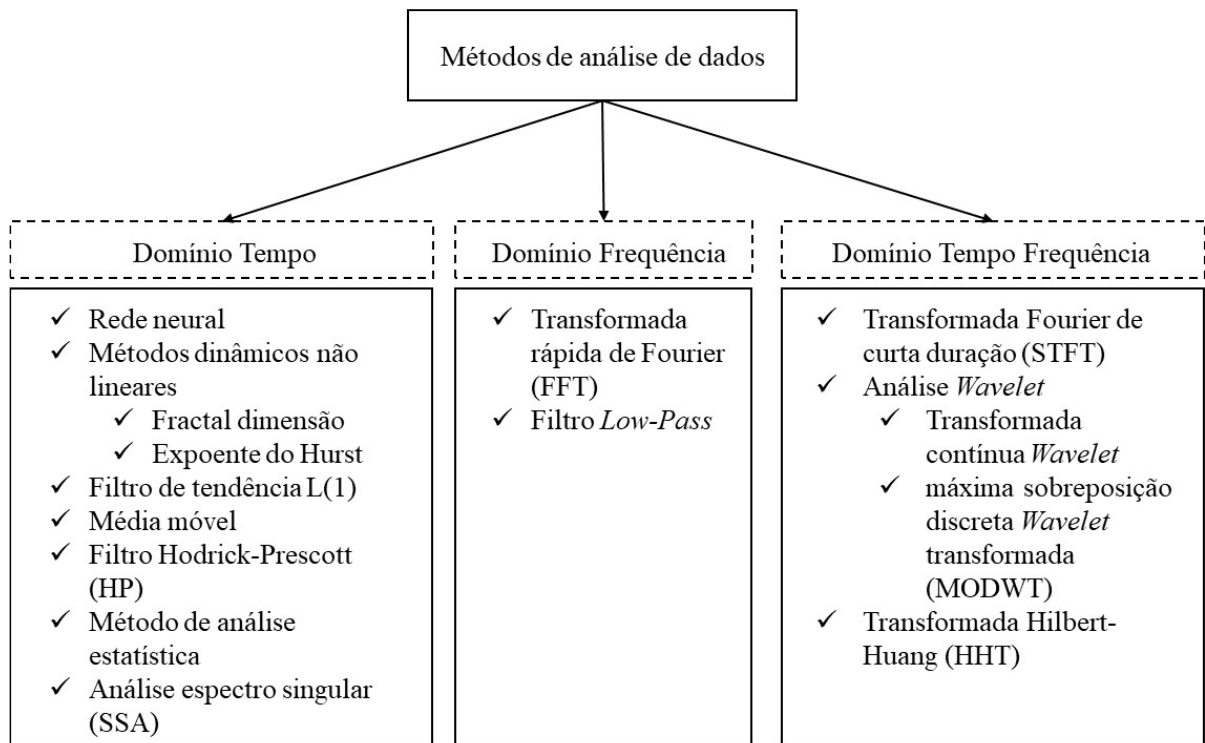
Na Tabela 4.5 apresenta-se uma síntese das principais vantagens e desvantagens dos três tipos de métodos de seleção de variáveis e/ou atributos.

**Tabela 4.5** – Comparação de métodos de seleção de características.

<b>Tipos</b>	<b>Vantagens</b>	<b>Desvantagens</b>
Filtro	Execução rápida e robustez à superajuste Computacionalmente mais barato que os invólucros e os embarcados	Falha em encontrar o subconjunto de variáveis mais promissor, podendo selecionar variáveis redundantes
Invólucro	Leva em consideração o viés do algoritmo de indução Considera as variáveis dentro do contexto Alta precisão preditiva em dados de teste	Computacionalmente caro em relação aos demais métodos (execução lenta) Maior risco de superajuste quando comparados com os filtros e os embarcados
Embarcado	Combinação das vantagens dos filtros e dos invólucros Computacionalmente menos intensivo que o invólucro	Computacionalmente mais caro que os filtros Não é adequado para dados de alta dimensão Conhecimento preliminar para procedimento de seleção

**Fonte:** Elaborado pelo autor

Os métodos orientados a dados empregados na extração de atributos podem ser classificados em métodos baseados no domínio tempo, domínio frequência ou domínio tempo-frequência, como apresentado na Figura 4.20 (PYAYT *et al.*, 2014).



**Figura 4.20** - Classificação de métodos baseados em dados nos domínios de tempo, frequência e tempo-frequência (PYAYT *et al.*, 2014)

As abordagens filtro e invólucros produzem diferentes ênfases na melhoria da precisão ou na redução de dimensionalidade. Recentemente, as abordagens híbridas têm sido propostas para combinar as vantagens dos filtros e dos invólucros (STAŃCZYK, 2015). Segundo Zhang e Qi (2005), os métodos de aprendizagem de máquinas, sem o devido pré-processamento adequado, podem se tornar instáveis e produzir resultados abaixo do ideal.

Um exemplo deste modelo híbrido é o algoritmo automático proposto por Crone e Kourentzes (2010). Este algoritmo, puramente orientado a dados, combina as vantagens dos filtros e dos invólucros e será empregado nesta tese para realizar a previsão de demanda horária de água urbana usando o ELM.

Segundo os autores, o princípio dessa metodologia é utilizar a eficiência dos filtros para as propriedades conhecidas e bem especificadas e os invólucros para opções de modelagem adicionais com espaços de pesquisa limitado, combinando, assim, as melhores práticas de estatística com as de inteligência artificial. O filtro proposto, denominado de filtro neural iterativo (INF), baseia-se na combinação da estimativa de Distância Euclidiana (DE) e da MLP. O uso da MLP para estimar o INF ocorre devido a sua capacidade de aproximação universal.

Contudo, essa topologia difere-se daquela usada para previsão. As entradas da MLP não consistem nas defasagens da variável dependente, mas das variáveis explicativas contemporâneas que agrupam padrões deterministas de séries temporais. Visando codificar a sazonalidade, duas variáveis explicativas de entradas são criadas. Estas variáveis codificam uma sazonalidade determinística como sendo pares de entrada seno-cosseno para cada comprimento sazonal da variável, diminuindo substancialmente o tamanho do vetor de entrada e das múltiplas sazonalidades.

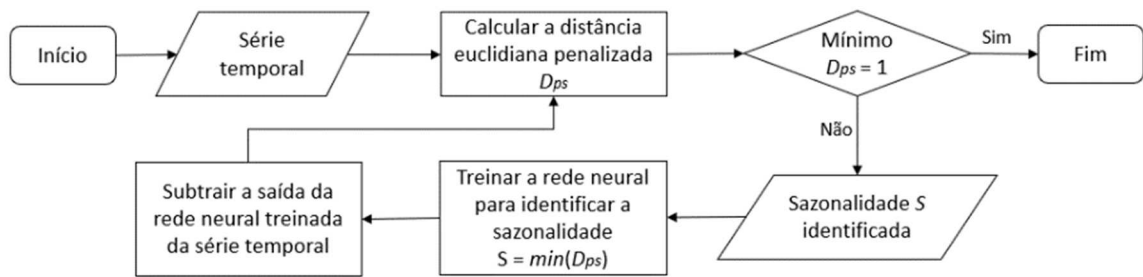
Ainda, segundo os autores, esse filtro é capaz de remover qualquer tipo de sazonalidade não-linear na presença de outros efeitos sazonais, tendências e irregularidades na série temporal. Embora o INF seja capaz de avaliar e identificar com precisão todas as características relevantes da série temporal, não consegue distinguir entre sazonalidade estocástica ou determinística, necessária para construção de características adequadas das variáveis *dummy*. Do mesmo modo, este algoritmo não fornece informações sobre o pré-processamento mais adequado da série temporal por meio da transformação, como a diferenciação única para séries temporais não estacionárias ou a diferenciação sazonal para séries sazonais.

A topologia utilizada neste filtro é composta de  $16^{17}$  nós ocultos dispostos em uma única camada com funções de ativação tangente hiperbólica e um único nó de saída. Os pesos da MLP são inicializados aleatoriamente para cada iteração de estimativa INF e a rede é treinada empregando o algoritmo retropropagação. Posteriormente, a saída da rede é subtraída da série de tempo original, criando efetivamente uma série de tempo filtrada a partir da qual o padrão dominante do sinal periódico será removido. Após isso, este processo é repetido para identificar gradualmente um delimitador de periodicidades remanescentes até que não haja mais sazonalidade nas séries temporais como ilustrado na Figura 4.21.

A Figura 4.21, que segue abaixo, apresenta o fluxograma do INF.

---

<sup>17</sup> Segundo Crone e Kourentzes (2010) experiências preliminares em ensaios de sazonalidade única e múltipla, diferentes níveis de ruídos indicaram a necessidade de um número relativamente grande de nós ocultos para capturar os sinais periódicos complexos.



**Figura 4.21** – Fluxo do filtro neural iterativo (CRONE e KOURENTZES, 2010)

Após realizar a filtragem da série empregando o algoritmo INF, os componentes de séries temporais identificados fornecerão o primeiro conjunto de candidatos a vetores de entrada. Posteriormente, um invólucro simples em torno desse filtro é usado para criar candidatos adicionais. Após a identificação dos possíveis candidatos por meio da seleção de recursos, um conjunto de arquiteturas MLP é especificado por meio de invólucros para avaliar os nós ocultos, as camadas e as funções de ativação para cada vetor de entrada e treinados para prever os dados não vistos. Como resultado tem-se um grande conjunto de possíveis MLP candidatos com diferentes vetores de entrada utilizando várias informações temporais sobre tendência, sazonalidade determinística e estocástica. De todos os modelos candidatos combinados, um *ensemble* (comitê) com os 10 melhores modelos é usado para a previsão final.

## 4.7 Ensemble

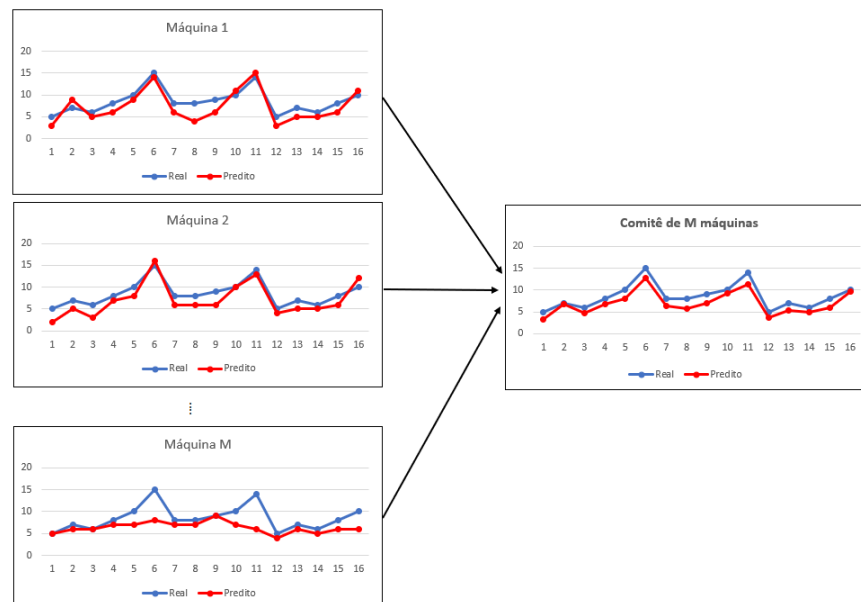
### 4.7.1 Conceitos gerais

O aprendizado empregando *ensemble* (comitê) refere-se a métodos que geram vários modelos combinados para prever um novo caso, seja em problemas de classificação ou de regressão. Segundo Ren *et al.* (2016), a principal teoria por trás do comitê é a decomposição de viés-variância-covariância, oferecendo a justificativa teórica para melhorar o desempenho de um conjunto sobre seus preditores de base.

A ideia destes métodos de comitê pode ser descrita como sendo a construção de um modelo preditivo através da integração de múltiplos modelos (DIETTERICH, 2000). Uma das vantagens é a melhora do desempenho da generalização (HANSEN e SALAMON, 1990;

SHARKEY, 1996; ZHANG, 2003; MELIN *et al.*, 2012). No entanto, tal melhoria se apoia na qualidade de seus componentes e na diversidade do erro apresentada por eles (GRANGER, 1989; PERRONE e COOPER, 1993; KROGH e VEDELSBY, 1995; SOLLICH e KROGH, 1996; GRANITTO *et al.*, 2005, GASHLER *et al.*, 2008; AL-ZAHRANI e ABO-MONASAR, 2015; REN *et al.*, 2016; WANG *et al.*, 2018), ou seja, cada um dos componentes em um comitê deve apresentar um bom desempenho quando aplicado isoladamente ao problema e, ao mesmo tempo, deve generalizar de forma diferente. Não faz muito sentido combinar modelos que adotam os mesmos procedimentos e hipóteses para a solução de um problema (PERRONE e COOPER, 1993). Se forem combinados componentes que apresentam um mesmo padrão de erro, não haverá incremento de desempenho, aumentando apenas o custo computacional, sem resultados práticos de desempenho.

Na Figura 4.22 ilustra-se o ganho de desempenho usando um comitê de  $M$  máquinas em predição de séries temporais. Neste exemplo há  $M$  máquinas com desempenhos diferentes para o conjunto de predição (teste). As máquinas 1 e 2 apresentam melhores ajustes aos picos da série, e a máquina  $M$  obteve melhor desempenho nos valores baixos. Ao combinar essas predições utilizando uma média simples, obtém-se o resultado apresentado no lado direito da Figura 22. Conforme observado, o resultado obtido empregando um comitê apresenta um desempenho superior a qualquer uma das  $M$  máquinas tomadas de forma isolada.



**Figura 4.22** – Exemplo ilustrativo de ganho de desempenho usando um comitê de “ $M$ ” máquinas em predição de séries temporais. (Elaborado pelo autor)

As motivações estatísticas para usar um comitê no contexto de RNA também estão presentes em modelos estatísticos. O trabalho seminal a defender a combinação de modelos de previsão de séries temporais foi o trabalho proposto por Bates e Granger em 1969 (CLEMEN, 1989; WALLIS, 2011). Segundo Bates e Granger (1969), a possibilidade de aumentar a precisão de prognósticos beneficia-se da complementariedade da informação contida em cada previsão individual. Esse resultado decorre da proposição de que a variância esperada dos erros da previsão combinada é inferior a menor das variâncias das previsões individuais. Este resultado requer que as previsões sejam não-viesadas e não apresentem elevada correlação. A interação entre a variância e a correlação dos erros das previsões individuais é determinante para a performance do comitê.

O desempenho relativo de várias técnicas de previsão foi examinado em uma série de estudos empíricos: Newbold e Granger<sup>18</sup> (1974), Makridakis *et al.* (1982), Makridakis e Winkler (1983), Granger e Ramanathan (1984), Granger (1989), Diebold e Pauly (1990), Baxt (1992), Shamseldin *et al.* (1997), Makridakis e Hibon (2000), Armstrong (2001), Hendry e Clements (2004), Zhang (2007), Jose e Winkler (2008), Caiado (2010), Khashei e Bijari (2010) e Wallis (2011).

Os métodos de comitês mais amplamente conhecidos são o *Bagging (Bootstrap Aggregating)* introduzida por Breiman (1996), o *Boosting* (FREUND e SCHAPIRE, 1996) e a Floresta Aleatória (BREIMAN, 2001).

Segundo Mendes-Moreira *et al.* (2012), a geração de comitês homogêneos é a área de aprendizado com melhor cobertura pela literatura. Apresentam-se, como exemplos, os trabalhos de Dietterich (1997), Brown *et al.* (2005), Adhikari e Agrawal (2011) e de Chen *et al.* (2017). Em comitês homogêneos, os modelos são gerados empregando o mesmo algoritmo. Já os comitês heterogêneos são gerados quando mais de um algoritmo de aprendizado de máquina é usado na sua construção (Wang *et al.*, 2018). Espera-se que na abordagem heterogênea se obtenham modelos com uma maior diversidade devido à natureza diferente dos aprendizes de base (WEBB e ZHENG, 2004). Alguns autores afirmam que os comitês heterogêneos têm desempenhos melhores quando comparados com comitês homogêneos (GRANGER, 1989; WICHARD *et al.*, 1990; KROGH e VEDELSBY, 1995). Os comitês heterogêneos podem

---

<sup>18</sup> Os resultados obtidos demonstraram que, na prática, é preciso ignorar as correlações na estimativa de pesos combinados. Médias ponderadas que dependiam de correlações estimadas tiveram desempenho insatisfatório.

servir-se de modelos de comitês homogêneos como aprendizes de base. Uma outra abordagem, usualmente empregada, é a que combina o uso de diferentes algoritmos de indução mesclados com diferentes conjuntos de parâmetros (ROONEY *et al.*, 2004). Em trabalho posterior, Gashler *et al.* (2008), apresenta uma técnica que combina algoritmos heterogêneos de árvores e os contrasta com algoritmos de florestas homogêneas. Os resultados obtidos indicam que as florestas aleatórias se saem mal quando confrontadas com atributos irrelevantes, enquanto a técnica heterogênea proposta as manipula de forma robusta.

De acordo com Lin *et al.* (2017), existem dois problemas na predição de séries temporais que precisam ser resolvidos. O primeiro problema é como alcançar a precisão, a estabilidade e a eficiência juntas, e o outro é como lidar com séries temporais com múltiplos regimes ou estados diferentes. A precisão e a estabilidade são importantes, ou seja, os resultados previstos para diferentes ensaios devem ser semelhantes e, assim, garantir que a previsão seja confiável. Ainda, no que tange à estabilidade, a abordagem deve ser robusta contra ruídos das séries temporais. No tocante à eficiência a resposta deve ser rápida, portanto, a abordagem de previsão não deve consumir muito recurso computacional. Para resolver estes problemas, modelos de comitê baseados em florestas aleatórias foram propostos (HEINERMANN e KRAMER, 2016; LIN *et al.*, 2017).

Segundo Heinermann e Kramer (2016), os métodos de comitê requerem menos ajuste e conhecimento de domínio especializado. Contudo, a fim de encontrar um preditor ideal do comitê, um *trade-off* entre múltiplos objetivos ocorre, ou seja, uma redução no erro de previsão muitas vezes é alcançada empregando mais tempo de computação.

#### 4.7.2 Fundamento teórico para o uso de um comitê<sup>19</sup>

Considere que  $y_m(x)$  representa a saída do componente  $m$ , para o padrão de treinamento  $x \in X$ , onde  $X$  é o espaço de entrada e  $m = 1, 2, \dots, M$  sendo  $M$  o número de componentes treinados. Suponha que  $h(\cdot): X \rightarrow \mathbb{R}$  seja a verdadeira função de regressão que estamos tentando prever, de modo que cada um dos modelos seja dado como um valor verdadeiro  $h(x)$  mais um erro  $e_m(x)$ . Pode-se escrever a função de mapeamento que cada componente realizará como sendo:

---

<sup>19</sup> Esse tópico baseia-se em Bishop (2006)

$$y_m(x) = h(x) + e_m(x), x \in X \quad (4.7.2.1)$$

O erro quadrático médio para o componente  $m$  ( $MSE_m$ ) assume a seguinte forma:

$$MSE_m = E[\{y_m(x) - h(x)\}^2] = E[e_m(x)^2] \quad (4.7.2.2)$$

em que  $E[\cdot]$  denota a esperança matemática com relação à distribuição do vetor de entrada  $x$ .

A partir da equação (4.7.2.1), o erro quadrático médio obtido pelos  $M$  componentes que atuam individualmente é, portanto:

$$MSE_{m\acute{e}dio} = \frac{1}{M} \sum_{m=1}^M MSE_m = \frac{1}{M} \sum_{m=1}^M E[e_m(x)^2] \quad (4.7.2.3)$$

Considere agora a abordagem em que o comitê utiliza a média simples como forma de combinação. Logo, a saída do comitê  $y_{comit\hat{e}}(x)$  é:

$$y_{comit\hat{e}}(x) = \frac{1}{M} \sum_{m=1}^M y_m(x) \quad (4.7.2.4)$$

Similarmente, o erro cometido pelo comitê,  $MSE_{comit\hat{e}}$ , será dado como:

$$MSE_{comit\hat{e}} = E \left[ \left\{ \frac{1}{M} \sum_{m=1}^M y_m(x) - h(x) \right\}^2 \right] = E \left[ \left\{ \frac{1}{M} \sum_{m=1}^M e_m(x) \right\}^2 \right] \quad (4.7.2.5)$$

Ao assumir que os erros  $e_m(x)$  possuem média zero e não são correlacionados, tem-se então:

$$E[e_m(x)] = 0, \quad E[e_m(x)e_l(x)] = 0 \quad \text{se } m \neq l \quad (4.7.2.6)$$

Da Equação (4.7.2.3) e da Equação (4.7.2.5), e levando-se em conta os resultados da Equação (4.7.2.6), obtém-se:

$$MSE_{comit\hat{e}} = \frac{1}{M^2} \sum_{m=1}^M E[e_m(x)^2] = \frac{1}{M} MSE_{m\acute{e}dio} \quad (4.7.2.7)$$

Este resultado é bastante interessante, pois sugere que a somatória do erro quadrático pode ser reduzida por um fator  $M$  simplesmente tirando a média das saídas dos  $M$  modelos. No entanto, isso depende da suposição de que os modelos individuais sejam não correlacionados, o que na prática não ocorre. Geralmente os modelos diferentes são correlacionados, logo, a suposição representada na equação (4.7.2.6) não é verdadeira. Portanto, é necessário gerar modelos com  $MSE$  baixo e que sejam poucos correlacionados com outros componentes.

Usando as desigualdades de Cauchy-Schwarz<sup>20</sup>, Perrone (1993) demonstra que, mesmo para preditores correlacionados e tendenciosos, o erro de predição ao quadrado da máquina do comitê (obtido pela média) é igual ou menor ao erro médio de predição quadrática dos membros do comitê, ou seja, o  $MSE_{comitê} \leq MSE_{médio}$ . O teorema de desigualdade de Cauchy-Schwarz e sua prova matemática se encontram descritos no Anexo I.

### 4.7.3 Bagging

A ideia principal do método *bagging* (BREIMAN, 1996) é criar preditores independentes usando amostras de conjunto de treinamento e calcular a média (para regressão) e a votação (para classificação) da saída desses preditores. O nome *bagging* vem do **Bootstrap Aggregating**. Como o nome insinua, este método utiliza a técnica de *bootstrap* (EFRON, 1979), com reposição, para gerar conjuntos distintos de treinamento a partir de um único conjunto de dados, que serão usados para aprender hipóteses diferentes. Breiman (1996) demonstra que os ensacamentos de árvores de decisão e árvores de regressão funcionam muito bem quando comparadas com árvores isoladas. Ainda, segundo Breiman (1996), a variância do preditor via *bagging* é menor ou igual ao preditor sem *bagging*, enquanto o erro ao quadrado do preditor permanece próximo, podendo ser superior, ao preditor sem *bagging*.

Este método baseia-se na instabilidade dos aprendizes de base, podendo ser explorados objetivando melhorar o desempenho preditivo de tais aprendizes. Conforme relatado na literatura, um fator crítico para que este método funcione melhor é a instabilidade do algoritmo

---

<sup>20</sup> Para maiores detalhes ver BECKENBACK, E.; BELLMAN, R. **Inequalities**. Springer-Verlag, Berlin, 1961.

de treinamento (BREIMAN, 1996; FREUND e SCHAPIRE, 1996; BUJA e STUETZLE, 2000; QUINLAN, 2006). Breiman (1998) define informalmente instabilidade. Segundo este autor um preditor é dito instável quando pequenas variações no conjunto de treinamento causam grandes variações nos preditores treinados com ele. Alguns exemplos de algoritmos instáveis são as redes neurais, as árvores de decisão, os algoritmos de aprendizado de regras e as MARS (BREIMAN, 1994; DIETTERICH, 1997). Contudo, métodos tais como *K*-NN, *Ridge Regression*, *Naive Bayes* e regressão linear são estáveis, não podendo ser melhorados via *bagging*.

Em um ajuste típico de regressão, dado um determinado conjunto de treinamento  $D$  com  $m$  instâncias, tal como  $D = \{(X_1, y_1), \dots, (X_m, y_m)\}$ , onde  $X_i$  é uma instância e  $y_i$  é um alvo, a tarefa é aprender uma função aproximada  $f: X \rightarrow \mathbb{R}$  da verdadeira função  $f^0$  de  $D$ . Dado  $f_j, j = 1, \dots, k$  seja um conjunto de aprendizes de regressão de base que produzem previsões  $f_j(X_i)$ . A saída de um único conjunto de regressão  $\hat{f}(X_i)$ , por exemplo  $X_i$ , pode ser expressa como:

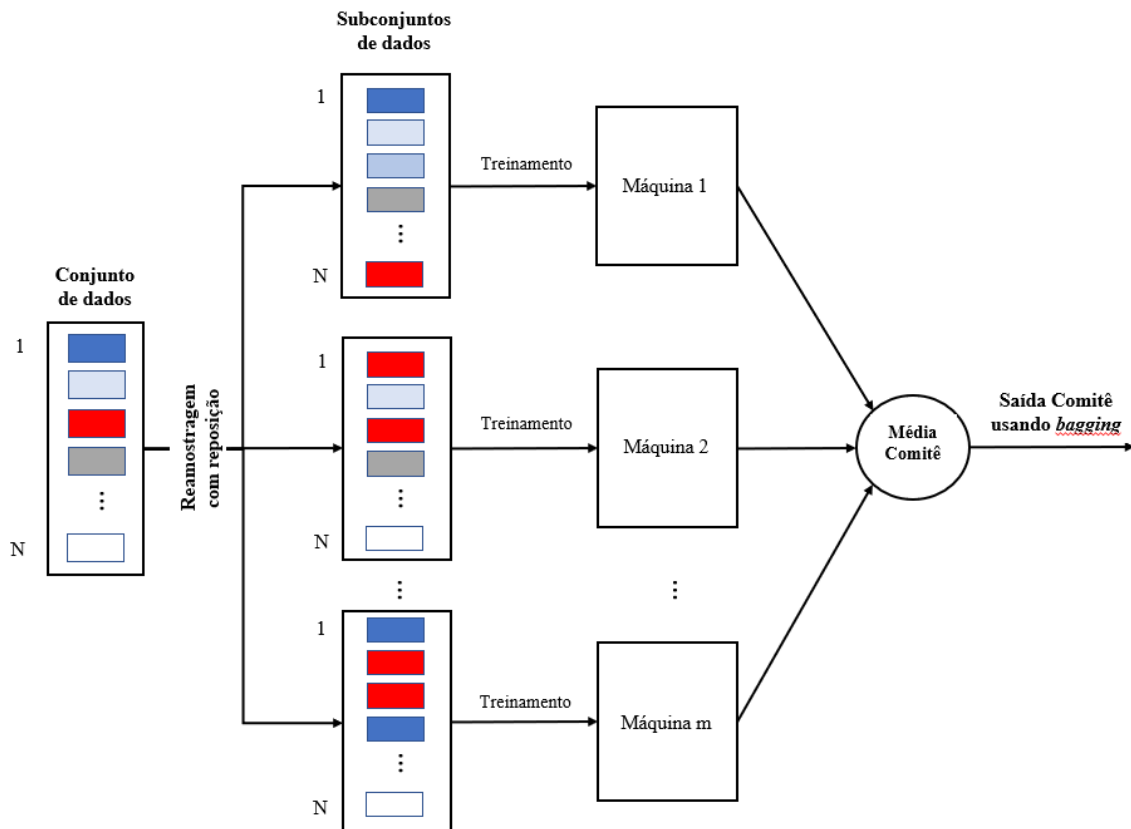
$$\hat{f}(X_i) = \sum_{j=1}^k w_j f_j(X_i) \quad (4.7.3.1)$$

em que  $f_j$  é a predição do  $i$ -ésimo membro das entradas do comitê  $x$  e  $w_i$  é o peso que muitas vezes são necessários para ser positivo e soma para um. A saída final do ensacamento é baseada na média simples das  $k$  amostras *bootstrap* (BREIMAN, 1996).

Por exemplo, em uma configuração de regressão usando a Equação (4.7.3.1), o peso  $w_i$  aprendiz  $f_j$  é  $1/k$ . Então a resposta da estimação será dada por:

$$\hat{f}(X_i) = \frac{1}{k} \sum_{j=1}^k w_j f_j(X_i) \quad (4.7.3.2)$$

Na Figura 4.23 apresenta-se esquematicamente a geração de conjuntos utilizando a técnica de ensacamento.

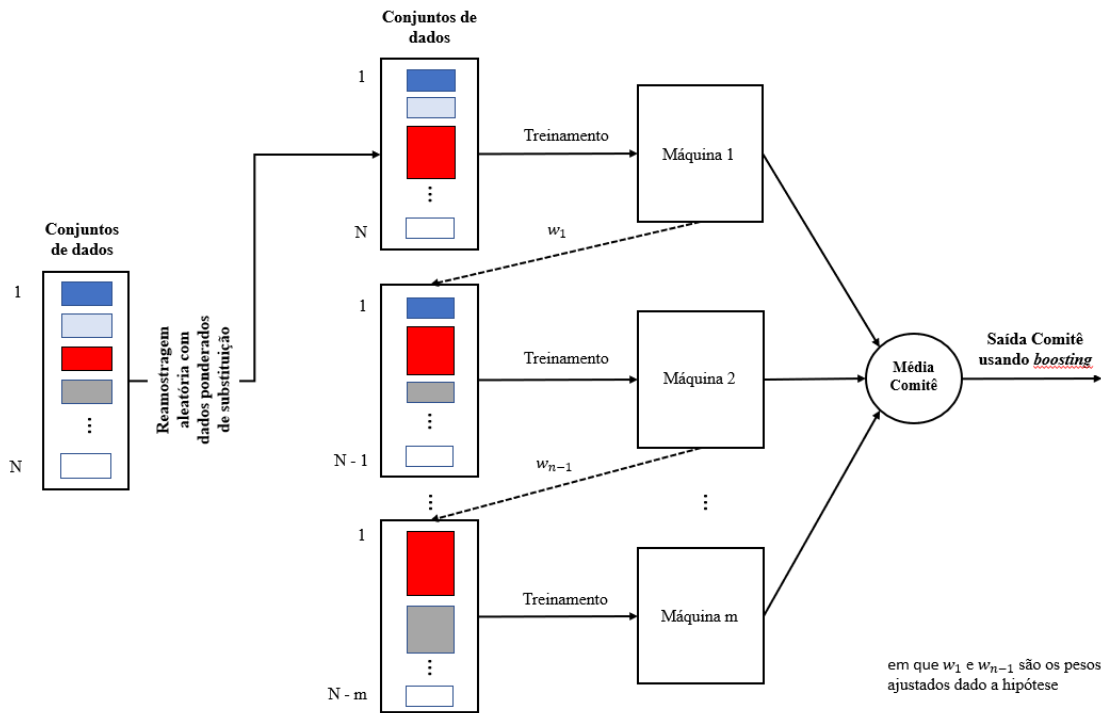


**Figura 4.23** – Geração de conjuntos de treinamento utilizando *bagging*. (Elaborado pelo autor)

#### 4.7.4 *Boosting*

Nos algoritmos de *boosting* (impulsioneamento), cada regressor é treinado em dados, levando em consideração o sucesso dos regressores anteriores. Após cada etapa de treinamento, os pesos são redistribuídos a fim de obter uma média ponderada de suas estimativas. A principal diferença entre os dois métodos ocorre no estágio de treinamento. No *bagging* cada modelo é construído paralelamente (ou seja, cada modelo ocorre de forma independente), já no *boosting* ocorre de forma sequencial.

Em *bagging*, o resultado é obtido pela média das respostas dos  $N$  aprendizes (ou pelo voto da maioria). Contudo, no *boosting* atribui-se um segundo conjunto de pesos a fim de obter uma média ponderada de suas estimativas. A Figura 4.24 apresenta esquematicamente a geração de conjuntos utilizando a técnica de *boosting*.



**Figura 4.24** – Geração de conjuntos de treinamento utilizando *boosting*. (Elaborado pelo autor)

De acordo com Barrow e Crone (2016), a primeira aplicação de reforço à regressão empregando dados de séries temporais é creditada a Avnimelech e Intrator (1999), que utilizaram uma modificação do algoritmo AdaBoost original.

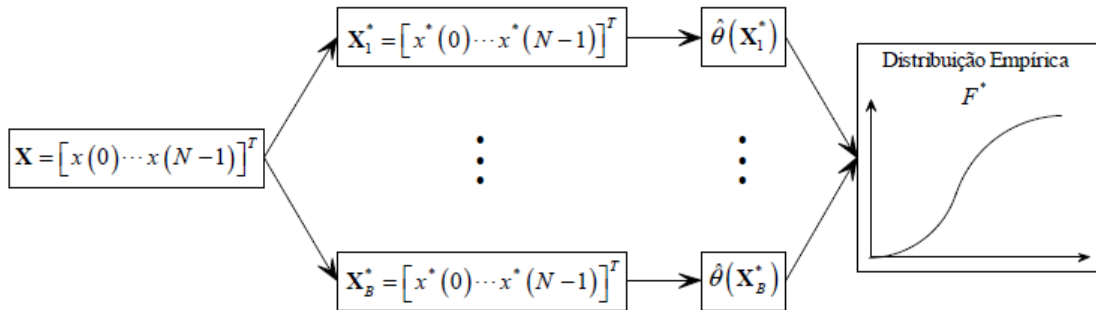
#### 4.7.5 Bootstrap

Segundo James *et al.* (2013), a técnica estatística de amostragem *bootstrap* é amplamente aplicável e extremamente poderosa podendo ser usada para quantificar a incerteza associada a um determinado estimador ou método de aprendizado estatístico. Tal procedimento foi introduzido por Efron (1979) e tem como base a amostragem de mesmo tamanho e com reposição de dados da amostra original.

Sejam  $N$  observações da variável aleatória  $X$  dispostas em forma de vetor  $X = [x(0) \cdots x(N-1)]^T$ , no qual são escolhidas aleatoriamente, e com igual probabilidade,  $N$  observações  $x(n)$  de  $X$ , criando um vetor amostrado de  $X^* = [x^*(0) \cdots x^*(N-1)]^T$  de  $X$ . Após a geração da amostragem calcula-se a estimativa  $\hat{\theta}(X^*)$  para o parâmetro  $\theta(X)$  por meio

do estimador  $\hat{\theta}$ . Posteriormente, repita esses passos  $B$  vezes e obtenha a distribuição empírica  $F^*$  de  $\hat{\theta}$ , que tende a se aproximar da verdadeira distribuição de  $F$  quando o  $B$  aumenta.

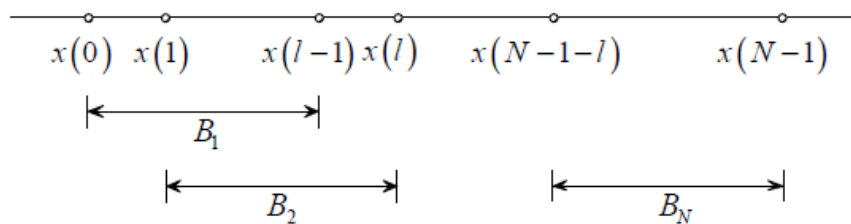
Na Figura 4.25 apresenta-se a rotina de *bootstrap* descrita acima.



**Figura 4.25** – Diagrama do procedimento de reamostragem *bootstrap* (EVARISTO, 2010)

Segundo Evaristo (2010), para se empregar a técnica de reamostragem *bootstrap* em séries temporais, é necessário primeiramente remover a correlação entre as observações.

Um método de reamostragem de séries temporais correlacionadas que é realizado no domínio do tempo, conhecido como *moving block bootstrap* (MBB) foi originalmente proposto por Künsch (1989) e Liu e Singh (1992). Esta técnica é realizada em blocos  $B_i$  de comprimento  $l$  do sinal, e está apresentada na Figura 4.26 em que são escolhidos os trechos de comprimento  $l$  para serem reamostrados.



**Figura 4.26** - Diagrama do MBB (EVARISTO, 2010)

No MBB os blocos de dados de igual tamanho são determinados até o comprimento da série desejada ser alcançado. Para uma série de comprimento  $n$ , com um tamanho de bloco  $l$ , existem  $n - l + 1$  blocos possíveis. Bergmeir *et al.* (2016) empregaram tamanhos de blocos  $l = 8$  para

dados anuais e trimestrais e  $l = 24$  para dados mensais, ou seja, pelo menos dois anos completos, visando garantir que qualquer sazonalidade seja capturada.

#### 4.8 Séries temporais<sup>21</sup>

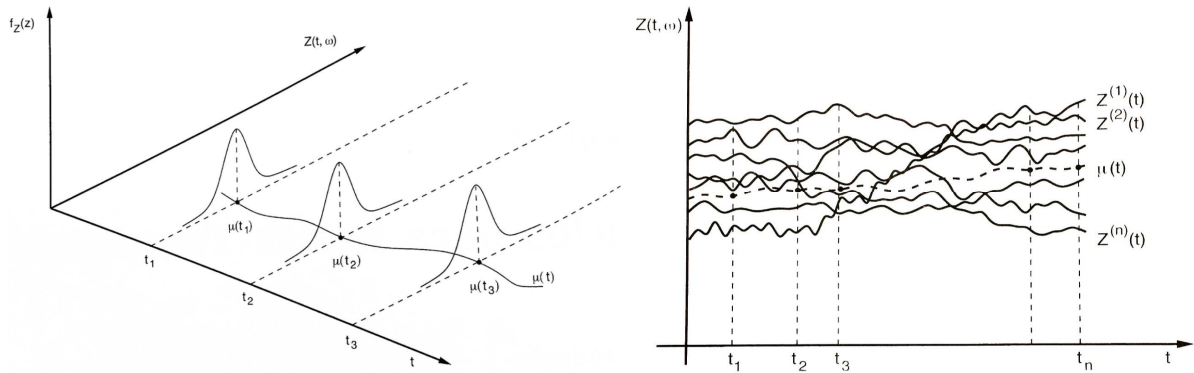
Segundo Hyndman *et al.* (2008), uma série temporal surge sempre que uma variável é observada ao longo do tempo, podendo esta ser contínua ou discreta. Muitas vezes, uma série temporal discreta  $\{t = 1, 2, \dots, T\}$  é obtida por amostragem contínua em intervalos de tempos iguais  $\Delta t$ . Portanto, a série contínua observada no intervalo  $[0, T]$  pode ser convertida em uma série discreta com  $N$  pontos, em que  $N = T/\Delta t$  (MORETTIN e TOLOI, 2004). No presente estudo as séries temporais serão sempre discretas e com espaçamento de tempo igual entre as observações.

Formalmente, pode-se definir uma série temporal  $\{Z_t, t = 1, 2, \dots, T\}$  como a realização de um processo estocástico, ou seja, uma família de variáveis aleatórias,  $Z = \{Z(t), t \in T\}$  na qual cada  $t \in T$ ,  $Z(t)$  é uma variável aleatória (v.a.) definida num mesmo espaço de probabilidades  $(\Omega, A, P)$ . O conjunto  $T$  é normalmente tomado como um conjunto de números inteiros  $Z = \{0, \pm 1, \pm 2, \dots\}$  ou um conjunto dos números reais. Também, para cada  $t \in T$ ,  $Z(t)$  será uma v.a. real. O conjunto  $\{Z(t), t \in T\}$  é denominado “Espaço dos Estados” e os valores de  $Z(t)$  são chamados de “Estados”. Como, para cada  $t \in T$ ,  $Z(t)$  é uma v.a. definida sobre  $\Omega$ , na realidade  $Z(t)$  é uma função de dois argumentos  $Z(t, w), t \in T, w \in \Omega$ .

Na Figura 4.27 (a), tem-se uma v.a.  $Z(t, w)$  com uma distribuição de probabilidade na qual é possível que a função de densidade de probabilidade (fdp) no instante de tempo  $t_1$  seja diferente da fdp nos instantes  $t_2$  e  $t_3$ , mas a situação usual é aquela em que a fdp de  $Z(t, w)$  é a mesma para todo  $t \in T$ .

---

<sup>21</sup> Este tópico terá como base o Capítulo 2 de Morettin e Tolo (2004).



Um processo estocástico interpretado como uma família de variáveis aleatórias

Um processo estocástico interpretado como uma família de trajetórias

(a)

(b)

**Figura 4.27** - Representação de um processo estocástico e seu conjunto de trajetórias (Adaptado de MORETTIN e TOLOI, 2004)

Em um processo estocástico  $\{Z_t: t = 0, \pm 1, \pm 2, \dots\}$  a função média é definida por

$$\mu_t = E(Z_t) \tag{4.8.1}$$

a função de autocovariância (FACV) por

$$\gamma(s, t) = cov(Z_s, Z_t) = E[(Z_s - \mu_s), (Z_t - \mu_t)] \tag{4.8.2}$$

e a função de autocorrelação (FAC) representada por

$$\rho(s, t) = corr(Z_s, Z_t) = \frac{\gamma(s, t)}{\sqrt{\gamma(s, s) \gamma(t, t)}} \tag{4.8.3}$$

A estacionariedade é uma suposição importante para um processo estocástico. A ideia básica de estacionariedade é que as leis de probabilidade que atuam no processo não mudam com o tempo, isso é, o processo mantém o equilíbrio ao redor de uma média constante, refletindo um equilíbrio estável. Se todas as v.a. tiverem a mesma distribuição tem-se, então, uma sequência de v.a. independentes e identicamente distribuídas (*i.i.d.*).

Diz-se que um processo estocástico  $Z = \{Z(t), t \in T\}$  é fracamente estacionário ou estacionário de segunda ordem apenas se:

$E[Z(t)] = \mu(t) = \mu$ , constante, para todo  $t \in T$ ;

$E[Z^2(t)] = E[(Z(t) - \mu)^2] = \sigma^2 < \infty$ , para todo  $t \in T$ ;

$\gamma(s, t) = Cov[Z(s), Z(t)]$  é uma função de  $|s - t|$

Um processo estocástico  $Z = \{Z(t), t \in T\}$  é dito gaussiano (ou normal) se para qualquer conjunto  $\{t_1, t_2, \dots, t_n\}$  de  $T$ , as v.a.  $Z(t_1), \dots, Z(t_n)$  têm distribuição normal n-variada. Se um processo for gaussiano ele será determinado pelas médias e covariâncias; em particular, se ele for estacionário de segunda ordem, ele será estritamente estacionário.

Devido à FACV  $\gamma(s, t)$  de uma série temporal estacionária ( $Z(t)$ ) depender somente de suas diferenças  $|s - t|$ , pode-se simplificar a notação com  $|s = t + h|$ , na qual  $h$  representa uma defasagem (ou alterações no tempo). Logo,  $\gamma(t + h, t) = cov(z_{t+h}, z_t) = cov(z_h, z_0) = \gamma(h, 0)$ , porque a diferença entre  $t + h$  e  $t$  é a mesma que a diferença do tempo  $h$  e 0. Portanto, a FACV de uma série temporal estacionária pode ser reescrita para:

$$\gamma(h) = cov(z_{t+h}, z_t) = E[(z_{t+h} - \mu), (z_t - \mu)] \quad (4.8.4)$$

e a FAC pode ser para:

$$\rho(h) = \frac{\gamma(t + h, t)}{\sqrt{\gamma(t + h, t + h) \gamma(t, t)}} = \frac{\gamma(h)}{\gamma(0)}, \quad h \in Z \quad (4.8.5)$$

As propriedades gerais da FACV para um processo estacionário são:

$$\gamma(0) = var(Z_t), \quad \rho_0 = 1$$

$$\gamma(h) = \gamma(-h), \quad \rho(h) = \rho(-h)$$

$$|\gamma(h)| \leq \gamma(0), \quad v|\rho(h)| \leq 1$$

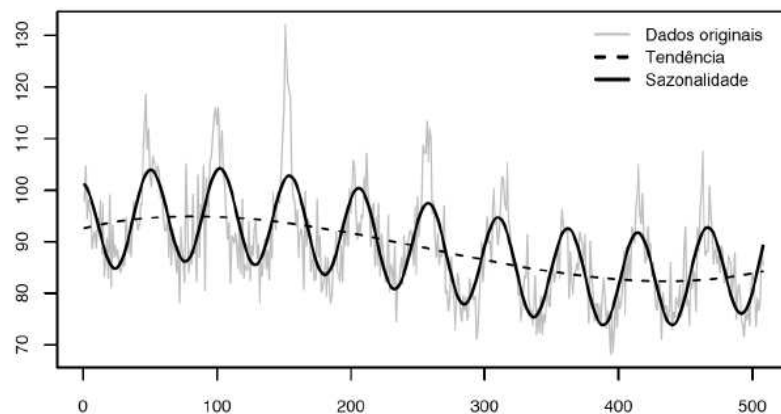
Um processo  $\{\epsilon_t, t \in Z\}$  é dito ruído branco discreto quando a sucessão de v.a. com a mesma distribuição, média constante  $E\{\epsilon_t\} = \mu$  (usualmente  $\mu = 0$ ), variância constante  $Var\{\epsilon_t\} = \sigma^2$  e com funções covariância e correlação iguais a:

$$\gamma_{\tau} = \begin{cases} \sigma^2 & \tau = 0 \\ 0 & \tau \neq 0 \end{cases} \quad \rho_{\tau} = \begin{cases} 1 & \tau = 0 \\ 0 & \tau \neq 0 \end{cases} \quad (4.8.6)$$

#### 4.9 Decomposição de séries temporais

Segundo Morettin e Toloí (2004), uma série temporal é qualquer conjunto de observações ordenadas no tempo podendo ser decomposta em quatro elementos: tendência, ciclo, sazonalidade e ruído aleatório ou erro que compreende a variabilidade intrínseca aos dados, não podendo ser modelado. Desse modo, cada uma das observações  $t_i$  que compõem a série temporal  $Z$  pode ser influenciada por uma ou mais dessas componentes. Contudo, não é possível identificar diretamente a atuação dessas componentes na série temporal, de modo que somente podem ser extraídas e compreendidas por meio de aplicação de métodos de decomposição (BROCKWELL e DAVIS, 2002).

Comumente as componentes sazonais e de tendência possuem uma forte relação, de modo que a influência da tendência sobre a componente sazonal pode afetar sobremaneira os métodos de análise. Nesse sentido, por meio da decomposição de uma série, pode-se isolar uma componente da outra (BROCKWELL e DAVIS, 2002; MORETTIN e TOLOI, 2004). A Figura 4.28 é uma representação dos componentes tendência e sazonalidade.



**Figura 4.28** - Representação dos componentes tendência e sazonalidade (FERRERO, 2009)

Os métodos de decomposição identificam esses padrões individualmente, possibilitando sua extração da série para uma melhor compreensão ou apenas para melhorar as previsões.

O termo ciclo ( $E_t$ ) é empregado quando a série temporal exhibe flutuações que não estão num período fixo. A tendência ( $T_t$ ) é verificada quando se observa uma alteração a longo prazo nos valores observados e o componente sazonal ( $S_t$ ) é descrito quando a série é influenciada pelos fatores sazonais, como por exemplo, o trimestre do ano, o mês, ou dia da semana. A sazonalidade é sempre de um período fixo e conhecido (HYNDMAN e ATHANASOPOULOS, 2014). A decomposição final poderá assumir dois modelos, aditivo ou multiplicativo, representados pelas equações a seguir:

$$Z_t = E_t + T_t + S_t \quad (4.9.1)$$

$$Z_t = E_t \cdot T_t \cdot S_t \quad (4.9.2)$$

Quando a magnitude das flutuações sazonais ou a variação em torno do ( $E_t$ ) não varia com o nível da série temporal, o emprego do modelo aditivo é mais adequado. Se a variação no padrão ( $S_t$ ), ou a variação em torno do ( $E_t$ ), parece ser proporcional ao nível das séries temporais, um modelo multiplicativo é mais apropriado. Sempre é possível obter um modelo aditivo a partir de um multiplicativo pela aplicação da função logaritmo a esse modelo, o que resulta na equação que segue:

$$\log(Z_t) = \log(E_t) + \log(T_t) + \log(S_t) \quad (4.9.3)$$

Segundo Hyndman e Athanasopoulos (2014), existem diversos métodos de decomposição dos componentes de uma série. O método clássico, que surgiu na década de 1920, é um procedimento relativamente simples e serve de base para a maioria dos outros métodos de decomposição. Outro método muito comum e robusto é o *Seasonal and Trend decomposition using Loess* (STL), desenvolvido por Cleveland *et al.* (1990).

#### **4.9.1 A Seasonal Trend decomposition procedure based on Loess (STL)**

O algoritmo STL, apresentado por Cleveland *et al.* (1990), é um filtro que tem como objetivo decompor a série temporal sazonal em três componentes: tendência, sazonalidade e resíduo. Suponha que os dados, o componente tendência, o componente sazonal e o componente resíduo sejam denotados por  $Y_v$ ,  $T_v$ ,  $S_v$  e  $R_v$ , respectivamente, para um determinado instante da série temporal  $v = 1$  até  $N$ . Tem-se então

$$Y_v = T_v + S_v + R_v \quad (4.9.1.1)$$

Este método consiste em uma sequência de operações de suavização, cada uma das quais, empregando a regressão ponderada localmente ou LOESS, desenvolvida por Cleveland (1979). A regressão local é um método não-paramétrico que estima as curvas e superfícies através da suavização. A ideia básica deste método pode ser observada ao considerar o método de regressão simples

$$y = g(x_i) + \epsilon_i \quad (4.9.1.2)$$

em que  $g$  é uma função de regressão desconhecida e  $\epsilon_i$  denota o erro independente e identicamente distribuído (*i.i.d.*) com distribuição normal, média zero e variância constante.

Diferentemente dos métodos paramétricos que estimam a função globalmente, a regressão local estima a  $g(\cdot)$  na vizinhança de cada ponto de interesse  $x = x_0$ . Duas escolhas devem ser feitas para realizar esta estimativa, sendo a primeira a definição do tamanho da vizinhança ( $h$ ) do ponto  $x = x_0$  e a segunda a definição da função ( $K$ ) que pondera o conjunto de pontos vizinhos a  $x_0$ . A função  $h$  é denominada de banda ou parâmetro de suavização (*bandwidth*) e a função  $K$  é denominada de núcleo (*Kernel*).

Portanto, a Equação 4.9.1.3 para a média local ponderada pelo núcleo é dada por

$$\hat{g}(x_0) = \frac{\sum_{i=1}^n K_h(x_i - x_0)y_i}{\sum_{i=1}^n K_h(x_i - x_0)} \quad (4.9.1.3)$$

Este estimador de núcleo foi proposto inicialmente por Nadaraya (1964) e Watson (1964). Contudo existem limitações na estimação de uma constante localmente. Com o objetivo de contornar este problema Stone (1977) e Cleveland (1979) propuseram a regressão local linear ponderada, na qual os pontos nessa vizinhança são ponderados de acordo com sua distância do respectivo ponto de dados. Se ao invés de estimar uma constante estimar uma linha reta localmente, o problema do viés de primeira ordem será eliminado e, portanto, a regressão local linear resolverá o problema de mínimos quadrados ponderados a cada ponto de interesse  $x_0$ , conforme a Equação 4.9.1.4:

$$\min_{\alpha\beta} \sum_{i=1}^n K_h(x_i - x_0) [y_i - \alpha - \beta(x_i - x_0)]^2 \quad (4.9.1.4)$$

A regressão local linear será igual ao estimador de Nadaraya-Watson expresso pela equação acima se o termo  $\beta(x_i - x_0)$  for removido e, assim, uma constante será estimada localmente. Segundo Bowman e Azzalini (1997), muitos autores utilizam a regressão local linear como técnica padrão, entretanto não há razões para não utilizar polinômios de ordem mais alta, mesmo porque o LOESS pode apresentar viés quando a função a ser estimada possui uma curvatura elevada. Para estes casos estima-se uma polinomial de grau  $d$  empregando-se a Equação 4.9.1.5 que segue abaixo

$$\min_{\alpha\beta_j, j=1, \dots, d} \sum_{i=1}^n K_h(x_i - x_0) \left[ y_i - \alpha - \sum_{j=1}^d \beta_j (x_i - x_0)^j \right]^2 \quad (4.9.1.5)$$

Para que se possa modelar um determinado processo por LOESS deve-se realizar três escolhas: (i) a função do núcleo (*Kernel*) responsável por ponderar as observações na vizinhança no entorno de  $x_0$ , (ii) o parâmetro de suavização (*bandwidth*) que determina o tamanho desta vizinhança e (iii) o grau polinomial que afeta a relação entre viés e variância.

Segundo Cleveland e Loader (1996), para ajustar um modelo LOESS, deve-se escolher o parâmetro de suavização  $h$  e o grau polinomial  $d$ . Normalmente,  $d$  tem um valor de 1 a 3, onde  $d = 1$  corresponde a ajustes lineares locais,  $d = 2$  a polinômios quadráticos locais e  $d = 3$  a polinômios tri-cúbicos locais. Quanto maior o grau do polinômio menor será o viés e maior a variância e vice-versa. O valor do parâmetro  $h$  indica qual a proporção dos dados será utilizada na regressão local e, portanto, deve ser um valor positivo entre  $[0,1]$ . Este parâmetro é denominado de parâmetro de suavização porque controla a suavidade de um ajuste da regressão. Quando  $h$  se aproxima de zero, a estimativa tende a interpolar as observações. Exemplificando, se  $h = 0,3$ , somente 30% dos dados serão empregados para a regressão local e, com efeito, o modelo resultante reflete variações drásticas e é menos suave. Por outro lado, se  $h = 0,7$ , 70% dos dados são usados para um ajuste local, gerando um modelo muito mais forte. Ou seja, um  $h$  elevado torna o modelo menos sensível a variações de dados.

A ideia do método STL é aplicar amortecimentos baseados na regressão localmente ponderada (LOESS) uma quantidade considerável de vezes gerando estimativas para os componentes de tendência e sazonalidade.

O algoritmo deste método conta com dois processos recursivos estando o *loop* interno inserido no *loop* externo. Portanto, os processos serão designados por processo interno e externo, respectivamente. O processo interno conta  $n_{(i)}$  iterações sendo feita uma atualização dos componentes tendência e sazonalidade. Cada iteração é seguida pelo cálculo dos pesos de robustez, e esses pesos são empregados na próxima iteração reduzindo a influência do comportamento transitório e anômalo na tendência e sazonalidade. A iteração inicial do *loop* externo é executada com peso de robustez igual a 1 e, em seguida, as passagens  $n_{(0)}$  do *loop* externo são executadas.

O pseudo-algoritmo STL conta com dois processos recursivos estando um inserido no outro.

**Processo interno** – Suponha  $S_v^k$  e  $T_v^k$  para  $v = 1$  até  $N$  são os componentes sazonalidade e tendência até a  $k^{\text{th}}$  iteração. Os passos para calcular a  $S_v^{(k+1)}$  e a  $T_v^{(k+1)}$  são:

**Primeiro passo:** Remoção da tendência. A dessazonalização da série  $Y_v - T_v^{(k)}$  é computada.

**Segundo passo:** Suavização das subséries. Cada subsérie é suavizada pelo método LOESS usando  $q = n_{(s)}$  e  $d = 1$ . Os valores suavizados são computados para todos pontos de cada subsérie. O conjunto destes valores, para todas as subséries, forma a série sazonal  $C_v^{(k+1)}$ , consiste em  $N + 2n_{(p)}$  valores de que variam de  $v = -n_{(p)} + 1$  para  $N + n_{(p)}$ .

**Terceiro passo:** Aplicação do filtro *low-pass*. Este filtro consiste no cálculo de  $L_v^{(k+1)}$ , para  $v = 1$  até  $N$ , aplicado à série sazonal  $C_v^{(k+1)}$  uma média móvel de tamanho  $n_{(p)}$ , seguida por outra média móvel de tamanho  $n_{(p)}$ , seguida por uma média móvel de comprimento 3, seguida ainda por uma suavização empregando LOESS com  $d = 1$  e  $q = n_{(l)}$ .

**Quarto passo:** Remoção da suavização das subséries. O componente sazonal para  $(k + 1)$  iterações é dada por  $S_v^{(k+1)} = C_v^{(k+1)} - L_v^{(k+1)}$  para  $v = 1$  até  $N$ .

**Quinto passo:** Dessazonalização. A série dessazonalizada  $Y_v - S_v^{(k+1)}$  é computada.

**Sexto passo:** Suavização da tendência. A série dessazonalizada é suavizada empregando o método LOESS com  $d = 1$  e  $q = n_{(t)}$ . O cálculo de  $T_v^{(k+1)}$ , é dado para os valores das  $(k + 1)$  iterações.

**Processo externo** – Após a execução inicial do *loop* interno, que tem como objetivo de obter a estimativa de  $T_v$  (componente tendência) e  $S_v$  (componente sazonal), o resíduo será dado por

$$R_v = Y_v - T_v - S_v \quad (4.9.1.6)$$

(observe que o resíduo  $R_v$ , diferentemente do  $T_v$  e  $S_v$ , não será definido onde  $Y_v$  tenha valor ausente). Após definir um peso para cada ponto onde  $Y_v$  é observado, a robustez destes pesos refletirá o quanto o  $R_v$  é extremo. Um *outlier* nos dados que resulta em um  $R_v$  muito grande terá um peso pequeno ou zero.

Um esquema de ponderação alternativo consiste em pesar os resíduos usando uma função *bisquare* definida como

$$B(x) = \begin{cases} (1 - |x|^2)^2 & \text{para } |x| < 1 \\ 0 & \text{para } |x| \geq 1 \end{cases} \quad (4.9.1.7)$$

a robustez  $\rho_v$  do peso será dada como

$$\rho_v = B\left(\frac{|R_v|}{h}\right) \quad (4.9.1.8)$$

em que  $h$  é seis vezes a mediana (Md) do valor absoluto dos resíduos  $h = 6 \text{ Md}(|R_v|)$ , isto é, o peso será ajustado para 0 se o valor absoluto do resíduo for maior que 6 Md.

Esse método fornece uma alternativa eficaz para excluir pontos específicos. Os *outliers* extremos são excluídos, mas os valores extremos moderados são menos ponderados do que completamente excluídos.

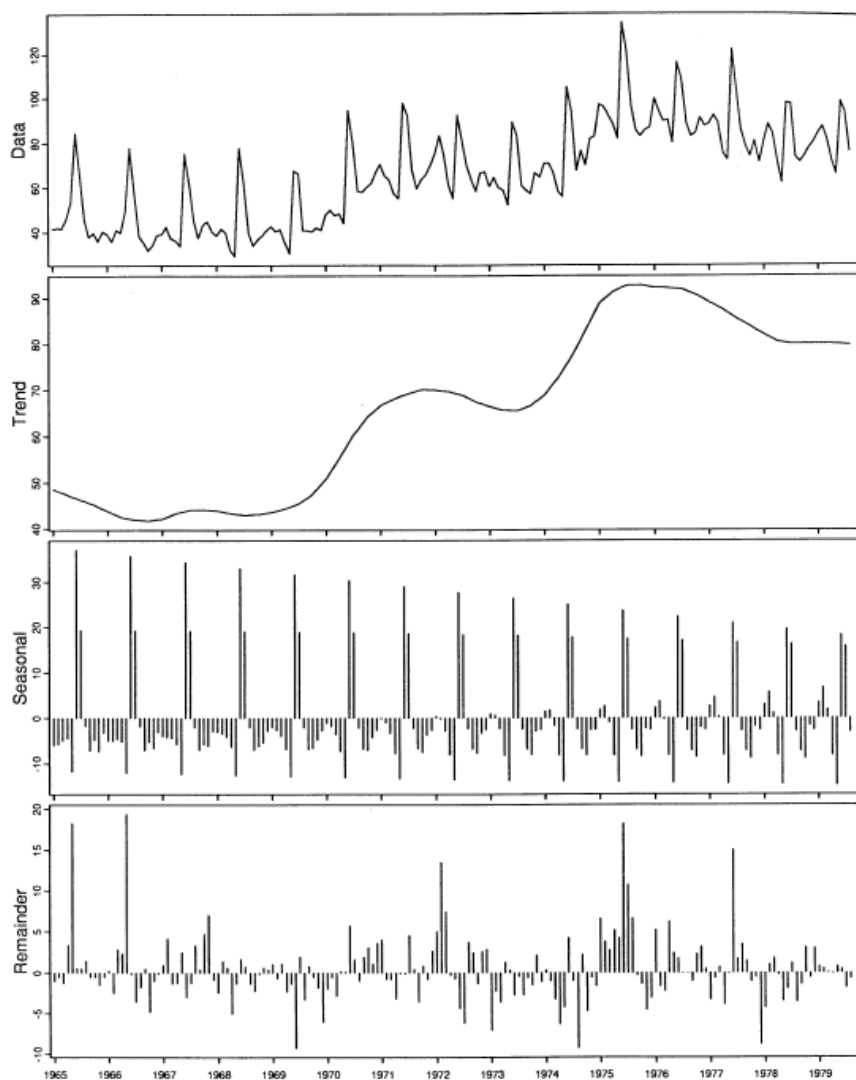
O STL possui seis parâmetros a serem definidos que estão sintetizados na Tabela 4.6.

**Tabela 4.6** – Escolha dos parâmetros do STL, descrição e estimação

Parâmetro	Descrição	Estimação
$n_{(p)}$	Número de observações em cada ciclo da componente sazonal	Para dados anuais com periodicidade mensal $n_{(p)} = 12$ , a primeira subsérie é o valor de janeiro, a segunda é fevereiro, e assim por diante. Cada uma dessas $n_{(p)}$ subséries é definida como subsérie de ciclos.
$n_{(i)}$ e $n_{(o)}$	Número de iterações para cada passagem pelo <i>loop</i> interno e número de iterações de robustez do <i>loop</i> externo, respectivamente	A estimação robusta é necessária quando o comportamento não gaussiano da série possui valores extremos (existência de <i>outliers</i> ). Se isto acontecer, tornar $n_{(o)} = 0$ e $n_{(i)} = 2$ ; caso contrário, tornar $n_{(i)} = 2$ e $n_{(o)} = 5$ ou $n_{(o)} = 10$ . O valor $n_{(o)}$ pode também ser determinado segundo um critério de convergência que termina quando o número de iterações satisfaz o critério.
$n_{(l)}$	Parâmetro de suavização para o filtro <i>low-pass</i> (transmitindo todas as frequências abaixo de um determinado valor).	Menor inteiro ímpar tal que $n_{(l)} \geq n_{(p)}$
$n_{(t)}$	Parâmetro de suavização da componente tendência	Menor inteiro ímpar tal que $n_{(t)} \geq \frac{1,5 n_{(p)}}{1 - 1,5 n_{(s)}^{-1}}$
$n_{(s)}$	Parâmetro de suavização da componente sazonalidade	Um número inteiro ímpar ou igual a 7 é escolhido com base no conhecimento da série temporal e na base dos métodos de diagnóstico de forma a minimizar a distância entre a reta resultante da aplicação da suavização e os valores de $S_k$

**Fonte:** Elaborado pelo autor

A Figura 4.29 apresenta a decomposição de uma série tempo empregando o método STL desenvolvido por Cleveland *et al.* (1990). O painel superior mostra os dados originais, o segundo mostra a tendência, o terceiro mostra a tendência sazonal e o último mostra a variação do resíduo ou "ruído".



**Figura 4.29** – Decomposição de uma série temporal empregando o STL (CLEVELAND *et al.*, 1990)

## **4.10 Modelos no contexto Espaço-Tempo**

### **4.10.1 Alisamento exponencial (ETS)**

Segundo Bergmeir *et al.* (2016), o alisamento exponencial é um dos métodos de previsão mais populares. Em estudo recente, Makridakis *et al.* (2018) empregaram 1045 subconjuntos de

dados mensais da competição M-3<sup>22</sup> para avaliar diversos métodos estatísticos e de aprendizagem de máquina. O método empregado que obteve a melhor acurácia foi o ETS.

Os primeiros trabalhos que serviram como base para esta classe de modelos, conforme salienta Gardner (1985), foram os trabalhos de Holt (1957), Brown (1959) e Winters (1960). Posteriormente, Pegels (1969) incorporou alternativa para o amortecimento da tendência; Ord *et al.* (1997) e Hyndman *et al.* (2002) formularam os modelos no contexto de espaços de estados; Taylor (2003) com o aprimoramento do amortecimento das tendências e Billah *et al.* (2006) na seleção do modelo de amortecimento exponencial.

#### 4.10.2 Formulação dos modelos de alisamento exponencial

A maioria dos métodos de previsão baseia-se na ideia de que observações passadas contêm informações sobre o padrão do comportamento da série temporal (MORETTIN e TOLOI, 2004). O propósito deste método é identificar este padrão e usá-lo para realizar previsões. Uma classe desses métodos é o do alisamento exponencial, cuja popularidade deve-se à sua simplicidade, à eficiência computacional e à razoável precisão. Estes modelos empregam pesos exponenciais decrescentes no sentido das observações mais recentes em direção às observações mais antigas, em que existem um ou mais parâmetros de suavização a serem determinados e a escolha desses parâmetros definirá os pesos a serem dados a cada observação.

Se os dados não possuem padrões de tendência ou sazonalidade, o alisamento exponencial simples é adequado. Se os dados indicarem tendência linear, o método *Holt* é apropriado. Porém, se os dados são sazonais é necessário aplicar o método *Holt-Winters* (HW) para captar a sazonalidade com base em três equações de suavização que compreendem o nível, que é estacionário e não exibe tendência ou sazonalidade, além dos componentes de tendência e de sazonalidade, e podem ser modelados de forma multiplicativa ou aditiva.

---

<sup>22</sup> A primeira Makridakis Competition aconteceu em 1982 e ficou conhecida na literatura de previsão como M-Competition, utilizado-se 1001 séries temporais e 15 métodos de previsão (com outras nove variações desses métodos). A M-2 utilizou somente 29 séries. Contudo, na M-3 empregaram 3003 séries, e teve o seu resultado apresentado em artigo de Makidrakis e Hibon (2000).

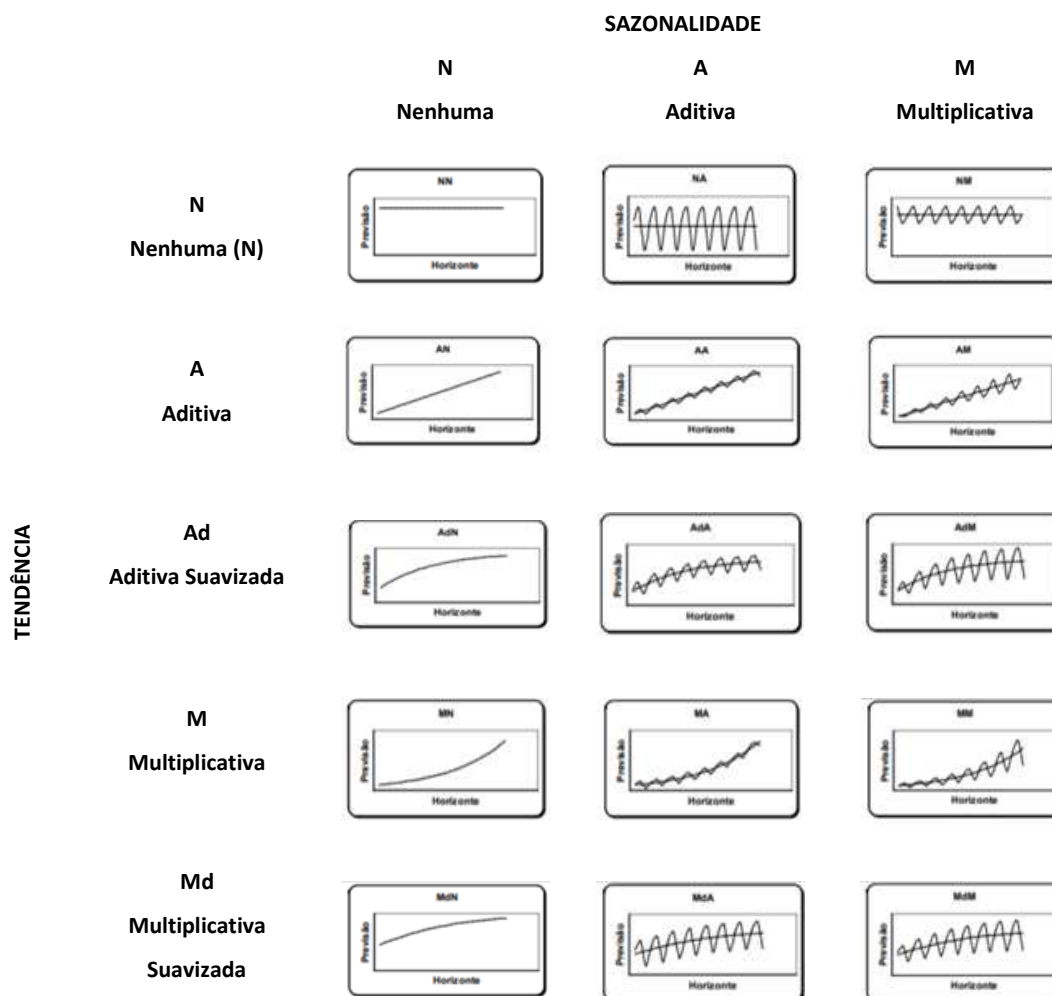
Hyndman *et al.* (2002) propuseram a taxonomia para os modelos de alisamento exponencial, de acordo com Tabela 4.7.

**Tabela 4.7** - Família de modelos de alisamento exponencial com diferentes tipos de sazonalidade e tendência.

Componente Tendência	Componente Sazonal		
	N (nenhum)	A (aditiva)	M (multiplicativa)
N (Nenhum)	N, N	N, A	N, M
A (Aditiva)	A, N	A, A	A, M
A <sub>d</sub> (Amortecimento aditiva)	A <sub>d</sub> , N	A <sub>d</sub> , A	A <sub>d</sub> , M
M (Multiplicativa)	M, N	M, A	M, M
M <sub>d</sub> (Amortecimento multiplicativa)	M <sub>d</sub> , N	M <sub>d</sub> , A	M <sub>d</sub> , M

**Fonte:** Hyndman *et al.* (2002).

Combinando as cinco formas de tendência com os três componentes sazonais, apresentados na Tabela 7, têm-se os 15 modelos de alisamento exponencial. O perfil destes 15 modelos é graficamente ilustrado na Figura 30.



**Figura 4.30** – Perfil de previsão (Adaptado de PEGELS (1969) e GARDNER (2006))

A nomenclatura NN descreve a suavização exponencial simples, AN se refere ao modelo *Holt*, AA caracteriza o modelo aditivo de *Holt-Winters*, enquanto o *Holt-Winters* multiplicativo é denotado por AM. As demais nomenclaturas representam modelos menos utilizados.

O conjunto de equações que descrevem cada um destes modelos apresentados segue abaixo:

NN (Sem tendência e sem sazonalidade)

$$I_t = \alpha y_t + (1 - \alpha)I_{t-1} \quad (4.10.2.1)$$

$$\hat{y}_{t+h|t} = l_t \quad (4.10.2.2)$$

AN (Tendência aditiva e sem sazonalidade)

$$l_t = \alpha y_t + (1 - \alpha)(l_{t-1} + b_{t-1}) \quad (4.10.2.3)$$

$$b_t = \beta^*(l_t - l_{t-1}) + (1 - \beta^*)b_{t-1} \quad (4.10.2.4)$$

$$\hat{y}_{t+h|t} = l_t + hb_t \quad (4.10.2.5)$$

AdN (Tendência aditiva amortecida e sem sazonalidade)

$$l_t = \alpha y_t + (1 - \alpha)(l_{t-1} + \phi b_{t-1}) \quad (4.10.2.6)$$

$$b_t = \beta^*(l_t - l_{t-1}) + (1 - \beta^*)\phi b_{t-1} \quad (4.10.2.7)$$

$$\hat{y}_{t+h|t} = l_t + \phi_h b_t \quad (4.10.2.8)$$

MN (Tendência multiplicativa e sem sazonalidade)

$$l_t = \alpha y_t + (1 - \alpha)l_{t-1}b_{t-1} \quad (4.10.2.9)$$

$$b_t = \beta^*(l_t/l_{t-1}) + (1 - \beta^*)b_{t-1} \quad (4.10.2.10)$$

$$\hat{y}_{t+h|t} = l_t b_t^h \quad (4.10.2.11)$$

MdN (Tendência multiplicativa amortecida e sem sazonalidade)

$$l_t = \alpha y_t + (1 - \alpha)l_{t-1}b_{t-1}^\phi \quad (4.10.2.12)$$

$$b_t = \beta^*(l_t/l_{t-1}) + (1 - \beta^*)b_{t-1}^\phi \quad (4.10.2.13)$$

$$\hat{y}_{t+h|t} = l_t b_t^{\phi h} \quad (4.10.2.14)$$

NA (Sem tendência e sazonalidade aditiva)

$$l_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)l_{t-1} \quad (4.10.2.15)$$

$$s_t = \gamma(y_t - l_{t-1}) + (1 - \gamma)s_{t-m} \quad (4.10.2.16)$$

$$\hat{y}_{t+h|t} = l_t + s_{t-m+h_m^+} \quad (4.10.2.17)$$

AA (Tendência aditiva e sazonalidade aditiva)

$$l_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(l_{t-1} + b_{t-1}) \quad (4.10.2.18)$$

$$b_t = \beta^*(l_t - l_{t-1}) + (1 - \beta^*)b_{t-1} \quad (4.10.2.19)$$

$$s_t = \gamma(y_t - l_{t-1} - b_{t-1}) + (1 - \gamma)s_{t-m} \quad (4.10.2.20)$$

$$\hat{y}_{t+h|t} = l_t + hb_t + s_{t-m+h_m^+} \quad (4.10.2.21)$$

AdA (Tendência aditiva amortecida e sazonalidade aditiva)

$$l_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(l_{t-1} + \phi b_{t-1}) \quad (4.10.2.22)$$

$$b_t = \beta^*(l_t - l_{t-1}) + (1 - \beta^*)\phi b_{t-1} \quad (4.10.2.23)$$

$$s_t = \gamma(y_t - l_{t-1} - \phi b_{t-1}) + (1 - \gamma)s_{t-m} \quad (4.10.2.24)$$

$$\hat{y}_{t+h|t} = l_t + \phi_h b_t + s_{t-m+h_m^+} \quad (4.10.2.25)$$

MA (Tendência multiplicativa e sazonalidade aditiva)

$$l_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)l_{t-1}b_{t-1} \quad (4.10.2.26)$$

$$b_t = \beta^*(l_t/l_{t-1}) + (1 - \beta^*)b_{t-1} \quad (4.10.2.27)$$

$$s_t = \gamma(y_t - l_{t-1}b_{t-1}) + (1 - \gamma)s_{t-m} \quad (4.10.2.28)$$

$$\hat{y}_{t+h|t} = l_t b_t^h + s_{t-m+h_m^+} \quad (4.10.2.29)$$

Mda (Tendência multiplicativa amortecida e sazonalidade aditiva)

$$l_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)l_{t-1}b_{t-1}^\phi \quad (4.10.2.30)$$

$$b_t = \beta^*(l_t/l_{t-1}) + (1 - \beta^*)b_{t-1}^\phi \quad (4.10.2.31)$$

$$s_t = \gamma(y_t - l_{t-1}b_{t-1}^\phi) + (1 - \gamma)s_{t-m} \quad (4.10.2.32)$$

$$\hat{y}_{t+h|t} = l_t b_t^{\phi h} + s_{t-m+h_m^+} \quad (4.10.2.33)$$

NM (Sem tendência e sazonalidade multiplicativa)

$$l_t = \alpha(y_t/s_{t-m}) + (1 - \alpha)l_{t-1} \quad (4.10.2.34)$$

$$s_t = \gamma(y_t/l_{t-1}) + (1 - \gamma)s_{t-m} \quad (4.10.2.35)$$

$$\hat{y}_{t+h|t} = l_t s_{t-m+h_m^+} \quad (4.10.2.36)$$

AM (Tendência aditiva e sazonalidade multiplicativa)

$$l_t = \alpha(y_t/s_{t-m}) + (1 - \alpha)(l_{t-1} + b_{t-1}) \quad (4.10.2.37)$$

$$b_t = \beta^*(l_t - l_{t-1}) + (1 - \beta^*)b_{t-1} \quad (4.10.2.38)$$

$$s_t = \gamma(y_t/(l_{t-1} - b_{t-1})) + (1 - \gamma)s_{t-m} \quad (4.10.2.39)$$

$$\hat{y}_{t+h|t} = (l_t + hb_t)s_{t-m+h_m^+} \quad (4.10.2.40)$$

AdM (Tendência aditiva amortecida e sazonalidade multiplicativa)

$$l_t = \alpha(y_t/s_{t-m}) + (1 - \alpha)(l_{t-1} + \phi b_{t-1}) \quad (4.10.2.41)$$

$$b_t = \beta^*(l_t - l_{t-1}) + (1 - \beta^*)\phi b_{t-1} \quad (4.10.2.42)$$

$$s_t = \gamma(y_t/(l_{t-1} - \phi b_{t-1})) + (1 - \gamma)s_{t-m} \quad (4.10.2.43)$$

$$\hat{y}_{t+h|t} = (l_t + \phi_h b_t)s_{t-m+h_m^+} \quad (4.10.2.44)$$

MM (Tendência multiplicativa e sazonalidade multiplicativa)

$$l_t = \alpha(y_t/s_{t-m}) + (1 - \alpha)l_{t-1}b_{t-1} \quad (4.10.2.45)$$

$$b_t = \beta^*(l_t/l_{t-1}) + (1 - \beta^*)b_{t-1} \quad (4.10.2.46)$$

$$s_t = \gamma(y_t/(l_{t-1}b_{t-1})) + (1 - \gamma)s_{t-m} \quad (4.10.2.47)$$

$$\hat{y}_{t+h|t} = l_t b_t^h s_{t-m+h_m^+} \quad (4.10.2.48)$$

MdM (Tendência multiplicativa amortecida e sazonalidade multiplicativa)

$$l_t = \alpha(y_t/s_{t-m}) + (1 - \alpha)l_{t-1}b_{t-1}^\phi \quad (4.10.2.49)$$

$$b_t = \beta^*(l_t/l_{t-1}) + (1 - \beta^*)b_{t-1}^\phi \quad (4.10.2.50)$$

$$s_t = \gamma(y_t/(l_{t-1}b_{t-1}^\phi)) + (1 - \gamma)s_{t-m} \quad (4.10.2.51)$$

$$\hat{y}_{t+h|t} = l_t b_t^{\phi h} s_{t-m+h_m^+} \quad (4.10.2.52)$$

em que:  $l_t$  denota o nível da série no tempo  $t$ ,  $b_t$  denota a inclinação no tempo  $t$ ,  $s_t$  denota o componente sazonal da série no tempo  $t$ , e  $m$  denota o número de sazonalidades no ano;  $\alpha, \beta^*, \gamma$  e  $\phi$  são as constantes de suavização da componente nível, tendência, sazonalidade e tendência amortecida, respectivamente, e  $\phi_h = \phi + \phi^2 + \dots + \phi^h$  e  $h_m^+ = [(h - 1) \bmod m] + 1$ .

Para cada um dos 15 métodos indicados acima, ainda existem duas possibilidades de modelagem no espaço de estados que correspondem ao modelo com erros aditivos e outro com erros multiplicativos. Dessa forma, existem 30 possibilidades de modelos de amortecimento exponencial para descrever uma série temporal (HYNDMAN *et al.*, 2002; GARDNER Jr., 2006).

## **4.11 Metodologia de Box e Jenkins**

### **4.11.1 Conceitos gerais**

O trabalho seminal de George Box e Gwilym Jenkins denominado de *Time Series Analysis: Forecasting and control* possibilitou a integração de todo conhecimento existente e promoveu um avanço nessa área de conhecimento. Adicionalmente, a evolução computacional ampliou o uso destes modelos a diversos campos da ciência (CHRISTODOULOS *et al.*, 2010).

Os modelos de Box e Jenkins, conhecidos genericamente como modelos autorregressivos integrados de Médias Móveis (ARIMA), são modelos matemáticos que visam captar o comportamento de autocorrelação entre os valores da série temporal.

Segundo Martínez e Zamprogno (2003), a modelagem de Box e Jenkins é classificada como uma metodologia clássica de previsão, a qual se baseia na construção de funções fundamentadas em um ciclo iterativo que utiliza os próprios dados da série com o intuito de encontrar uma estrutura que permita fazer previsões. Isto é, permite que valores futuros de uma série (neste caso, previsão de demanda urbana de água) sejam previstos utilizando apenas seus valores presente e passados, através da correlação temporal existente entre os valores exibidos pela série.

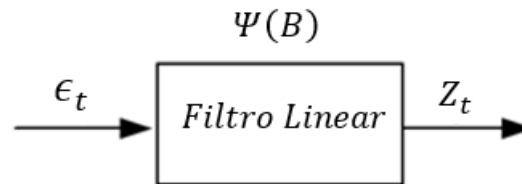
Esses modelos em teoria são a classe mais geral de modelos com estrutura estocástica para previsão de séries temporais estacionárias ou não estacionárias homogêneas, que por meio de transformações simples, como aplicar diferenciação ou o uso de logaritmos, tornam-se estacionárias.

### **4.11.2 Modelos Estacionários**

Os modelos estacionários são apropriados para descrever séries estacionárias, ou seja, séries que se desenvolvem no tempo aleatoriamente ao redor de uma média constante, refletindo de

alguma forma a estabilidade. Esses modelos são casos particulares de um modelo de filtro linear, cuja entrada é um ruído branco<sup>23</sup>.

Dessa forma, os modelos supõem que a série temporal seja gerada através de um filtro linear ou um sistema linear (MORETTIN e TOLOI, 2004), definido na Figura 4.31.



**Figura 4.31** - Filtro linear, com entrada  $\epsilon_t$ , função de transferência  $\Psi(B)$  e saída  $Z_t$   
(Adaptado MORETTIN e TOLOI, 2004)

Formalmente, temos que

$$Z_t = \mu + \epsilon_t + \psi_1 \epsilon_{t-1} + \psi_2 \epsilon_{t-2} + \dots = \mu + \psi(B) \epsilon_t \quad (4.11.2.1)$$

em que:  $\psi(B) = 1 + \psi_1 B + \psi_2 B^2 + \dots$  é denominada função de transferência do filtro e  $\mu$  é um parâmetro determinando o nível da série.

#### 4.11.3 Modelos autorregressivo AR( $p$ )

Suponha que  $\{\epsilon_t\}$  seja um processo puramente aleatório com média zero e variância  $\sigma^2$ . Um processo  $\{Z_t\}$  é denominado de processo autorregressivo de ordem  $p$ , ou AR( $p$ ), se  $Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \dots + \phi_p Z_{t-p} + \epsilon_t$ . Existe uma similaridade com um modelo de regressão múltipla, no qual os valores passados de  $Z_t$  fazem o papel das variáveis regressoras. Portanto, os processos AR podem ser usados como modelos se for razoável assumir que o valor atual de uma série temporal depende do seu passado imediato mais um erro aleatório. Exemplificando, um processo de primeira ordem AR(1), é dado por  $Z_t = \phi_1 Z_{t-1} + \epsilon_t$ .

---

<sup>23</sup> Uma sequência  $\{\epsilon_t\}_{t \in T}$  de variáveis aleatórias é denominada ruído branco quando estas variáveis são não correlacionadas com média zero e variância finita constante.

O Modelo denotado por  $AR(p)$  é definido como

$$Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \dots + \phi_p Z_{t-p} + \epsilon_t \quad (4.11.3.1)$$

$$Z_t - \phi_1 Z_{t-1} - \phi_2 Z_{t-2} - \dots - \phi_p Z_{t-p} = \epsilon_t \quad (4.11.3.2)$$

Empregando-se o operador de defasagens  $B$ , na Equação acima tem-se:

$$Z_t - \phi_1 B Z_{t-1} - \phi_2 B Z_{t-2} - \dots - \phi_p B Z_{t-p} = \epsilon_t \quad (4.11.3.3)$$

$$(1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p) Z_t = \epsilon_t \quad (4.11.3.4)$$

$$\phi_p(B) Z_t = \epsilon_t \quad (4.11.3.5)$$

Em que  $\epsilon_t \sim WN(0, \sigma^2)$  é ruído branco,  $\phi_p(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p$ , é o polinômio  $AR(p)$ , associado ao processo  $Z_t$ . O processo  $AR(p)$  é sempre invertível e é estacionário quando as raízes do polinômio autorregressivo associado estão fora do círculo unitário. A FAC deste tipo de processo apresenta decaimento exponencial e/ou sinusoidal para zero e a FACV é zero a partir de  $k = p + 1$ .

#### 4.11.4 Modelos de médias móveis $MA(p)$

O modelo  $MA(q)$ , conhecido como processo de média móvel de ordem  $q$ , consiste em expressar os valores de determinada série como uma combinação linear do valor corrente e valores passados de erros aleatórios. Semelhantemente aos modelos  $AR(p)$ , os modelos  $MA(q)$  também são uma soma ponderada de valores. Contudo, neste modelo os valores considerados são os erros aleatórios e seus  $q$  valores passados (FAVA, 2000). Segue abaixo a representação do modelo genérico:

$$Z_t = \epsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - \dots - \theta_q \epsilon_{t-q} \quad (4.11.4.1)$$

$$(1 - \theta_1 B + \theta_2 B^2 + \dots + \theta_q B^q) \epsilon_t = Z_t \quad (4.11.4.2)$$

$$\theta_q(B)\epsilon_t = Z_t \quad (4.11.4.3)$$

#### 4.11.5 Modelos autorregressivo de médias móveis ARMA ( $p, q$ )

Combinando-se modelos AR( $p$ ) e MA( $q$ ), obtêm-se os processos autorregressivos de médias móveis que formam uma classe de modelos muito úteis e parcimoniosos para descrever as séries temporais. O modelo ARMA( $p, q$ ) é dado por

$$Z_t = \phi_1 Z_{t-1} + \dots + \phi_p Z_{t-p} + \epsilon_t - \theta_1 \epsilon_{t-1} - \dots - \theta_q \epsilon_{t-q} \quad (4.11.5.1)$$

Em que  $\{\epsilon_t\}$  é um processo puramente aleatório com média zero e variância  $\sigma_\epsilon^2$ . Nota-se que, modelos AR ou MA podem ser obtidos como casos especiais quando  $p = 0$  ou  $q = 0$ . Usando o operador diferença, o modelo pode ser reescrito como:

$$(1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p)Z_t = (1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q)\epsilon_t \quad (4.11.5.2)$$

$$\phi(B)Z_t = \theta(B)\epsilon_t \quad (4.11.5.3)$$

Fava (2000) ressalta que para atender as condições de estacionariedade e invertibilidade de um modelo ARMA( $p, q$ ) os valores de  $\phi_1, \dots, \phi_p$ , que tornam o processo estacionário, devem ser tais que as raízes de  $\phi(B) = 0$  estejam fora do círculo unitário. Equivalentemente, os valores de  $\theta_1, \dots, \theta_p$ , que tornam o processo invertível<sup>24</sup>, sejam tais que as raízes de  $\theta(B) = 0$  também estejam fora do círculo unitário.

Em geral, para um processo ARMA( $p, q$ ) estacionário a FAC tem um decaimento exponencial ou oscilatório após a defasagem  $q$  enquanto que a função de autocorrelação parcial (FACP) tem o mesmo comportamento após a defasagem  $p$  (BOX e JENKINS, 1976). A princípio este resultado pode ser empregado para auxiliar na determinação da ordem ( $p, q$ ) do processo, mas na prática pode ser bastante difícil distinguir entre decaimentos exponenciais e oscilatórios por meio das estimativas destas funções.

---

<sup>24</sup> Para maiores detalhes ver Morettin e Toloi (2004) p. 112.

#### 4.11.6 Modelos ARMA integrados ARIMA( $p, d, q$ )

No mundo real, muitas das séries temporais encontradas apresentam alguma forma de não estacionariedade, ou seja, as características do processo estocástico mudam ao longo do tempo (GUJARATI, 2000; MORETTIN e TOLOI, 2004).

A classe de modelos ARIMA<sup>25</sup> é resultante da combinação de três componentes denominados de filtros, quais sejam, o componente autorregressivo (AR), o componente de médias móveis (MA) e o filtro de integração<sup>26</sup> (I). Segundo Fava (2000), uma série de tempo poderá ser modelada empregando um ou mais destes filtros.

A inclusão do filtro de integração, no modelo ARMA( $p, q$ ), permite que sejam utilizados os modelos ARIMA( $p, d, q$ ), dada pela equação 4.11.6.1:

$$W_t = \phi_1 W_{t-1} + \dots + \phi_p W_{t-p} + \epsilon_t - \theta_1 \epsilon_{t-1} - \dots - \theta_q \epsilon_{t-q} \quad (4.11.6.1)$$

em que:

$$W_t = \nabla^d Z_t = (1 - B)^d Z_t$$

Segundo Morettin e Tolo (2004), essa classe de modelos será capaz de descrever de maneira satisfatória séries estacionárias e séries que não possuem um comportamento explosivo. Esse tipo de não-estacionariedade é chamado de homogêneo. Entende-se por séries não estacionárias homogêneas, aquelas que não são estacionárias na média ou no nível, sendo as mesmas transformadas em estacionárias empregando o filtro de integração. A ordem de integração é o número de diferenças necessárias para tornar uma série não estacionária em estacionária.

Um modelo ARMA( $p, q$ ) no qual  $Z_t$  é substituído pela sua  $d$ -ésima diferença  $\nabla^d Z_t$  é capaz de descrever alguns tipos de séries não estacionárias.

---

<sup>25</sup> Este modelo é o mais popular para séries temporais não estacionárias, sendo considerado um dos principais modelos de previsão de séries temporais (Box *et al.*, 2008)

<sup>26</sup> O termo "integração" refere-se ao processo de diferenciação da série original para torná-la estacionária.

$$(1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p)W_t = (1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q)\epsilon_t \quad (4.11.6.2)$$

$$(1 - B)^d \phi(B)Z_t = \theta(B)\epsilon_t \quad (4.11.6.3)$$

Os modelos ARIMA também são capazes de modelar uma ampla gama de dados sazonais. Um modelo SARIMA é formado pela inclusão de termos sazonais adicionais aos modelos ARIMA. Tais modelos contêm uma parte não sazonal, com os parâmetros  $(p, d, q)$ , e uma sazonal, com parâmetros  $(P, D, Q)_s$ , podendo ser escrito como  $ARIMA(p, d, q) \times (P, D, Q)_s$ , em que  $p$  é o número de parâmetros autorregressivos não sazonal;  $d$  é o número de diferenças não sazonal necessário para tornar a série sazonal;  $q$  é o número de parâmetros médias móveis não sazonal;  $P$  é o número de parâmetros autorregressivos sazonal;  $D$  é o o número de diferenças sazonal necessário para tornar a série sazonal;  $Q$  é o número de parâmetros médias móveis sazonal; e  $s$  é o número de períodos por sazonalidade.

Seja  $Z_t$  a série de interesse observada em  $s$  períodos por ano e sejam

$$\phi(B^s) = 1 - \phi_1 B^s - \dots - \phi_P B^{sP} \quad (4.11.6.4)$$

o operador autorregressivo sazonal de ordem  $P$ , estacionário, e

$$\theta(B^s) = 1 - \theta_1 B^s - \dots - \theta_Q B^{sQ} \quad (4.11.6.5)$$

o operador de médias móveis sazonal de ordem  $Q$ , invertível,  $\nabla_s^D = (1 - B^s)^D$ , em que  $D$  indica o número de diferenças sazonais. A classe de modelos sazonais multiplicativos  $(p, d, q) \times (P, D, Q)$  é então dada por

$$(1 - \phi_1 B^s - \dots - \phi_P B^{sP})(1 - \theta_1 B^s - \dots - \theta_Q B^{sQ})[(1 - B)^d (1 - B^s)^D]Z_t \quad (4.11.6.6)$$

#### 4.11.7 Etapas da metodologia Box e Jenkins

As três etapas da metodologia Box e Jenkins são: (1) identificação, (2) estimação e teste, e (3) aplicação.

A fase de identificação consiste em preparar os dados e selecionar qual modelo, dentre as diversas versões da metodologia Box e Jenkins, descreve melhor o comportamento da série temporal. A identificação do modelo a ser estimado ocorre pelo comportamento das FAC e FACP. A estimação consiste em estimar os parâmetros  $(\phi, \Phi)$  do comportamento AR, os parâmetros  $(\theta, \Theta)$  do componente de MA e a variância de  $\epsilon_t$ , e selecionar o melhor modelo usando os critérios<sup>27</sup> adequados discutidos mais à frente. Posteriormente, checam-se as FAC e FACP dos resíduos. Se porventura o modelo não for adequado, o ciclo será novamente repetido, voltando-se à fase de identificação. Quando se obtém um modelo ajustado, passa-se para a terceira e última etapa da metodologia Box e Jenkins. A última etapa consiste da aplicação do modelo escolhido para realizar previsões.

Devido à diversidade dos modelos ARIMA, muitas vezes o esforço computacional empregado para selecionar um único modelo pode ser muito grande. Portanto, diversos procedimentos têm sido propostos a fim de selecionar os melhores modelos dentre as diversas possibilidades. A metodologia proposta por Hyndman e Khandakar (2008), cujo objetivo é selecionar automaticamente um modelo para previsão de séries temporais é denominada de Auto ARIMA. Este procedimento é baseado em testes de raízes unitárias e algoritmos chamados de *stepwise*, para reduzir o esforço computacional sem prejudicar a seleção do melhor modelo para realizar a previsão.

O principal objetivo em prever automaticamente modelos ARIMA é selecionar um modelo de ordem apropriada  $(p, d, q) \times (P, D, Q)$ . Hyndman e Khandakar (2008) optaram por iniciar o processo de maneira a descobrir o número de raízes unitárias que existem no processo. Existem diversos testes apresentados na literatura. O primeiro teste desenvolvido para testar a presença de raiz unitária foi proposto por Fuller (1976), posteriormente complementado por Dickey e Fuller (1979 e 1981). Outro teste foi proposto por Perron (1989, 1990) sob a hipótese de não estacionariedade na presença de quebra estrutural<sup>28</sup> nas séries. Ainda há o teste de Phillips e Perron (1988), Perron (1997), o teste HEGY<sup>29</sup> de Hylleberg *et al.* (1990), Zivot e Andrews

---

<sup>27</sup> Como exemplo temos o *Bayesian Information Criterion* (BIC) (Schwarz, 1978), o *Akaike Information Criterion* (AIC) (Akaike, 1973) e o *Akaike Information Criterion with correction* (AICc) (Hurvich e Tsai 1989)

<sup>28</sup> Ocorre quando as séries temporais são caracterizadas por uma mudança estrutural na média da amostra. Perron (1990) demonstra que a existência de tais mudanças em séries temporais estacionárias envia os testes usuais de raiz unitária, na direção da não rejeição.

<sup>29</sup> O objetivo deste procedimento é determinar se existem ou não raízes de unidade sazonal em uma série univariada.

(1992) e o teste KPSS proposto por Kwiatkowski et al. (1992). No entanto, por simplificação Hyndman e Khandakar (2008) consideram o teste que assume como hipótese nula a não existência de raízes unitárias (teste KPSS).

O teste de raiz unitária KPSS foi criado com o intuito de contrapor aos testes tradicionais. Este teste considera:

$H_0: \rho < 1$  (a série é estacionária)

$H_a: \rho = 1$  (a série é estacionária)

Sendo  $S_t = \sum_{i=1}^t \varepsilon_i$  e  $\hat{\sigma}^2$  um estimador para a variância dos erros, a estatística do teste KPSS é dada por:

$$LM = \sum_{i=1}^t \frac{S_t^2}{n^2 \hat{\sigma}^2} \quad (4.11.7.1)$$

O critério de rejeição da  $H_0$  do teste é: rejeitar  $H_0$  se  $LM >$  valores críticos. Tais valores críticos foram indicados por Kwiatkowski *et al.* (1992) de acordo com os valores assintóticos determinados.

#### 4.11.8 Critérios de comparação e seleção de modelos

Na análise de uma série temporal podem existir diversos modelos que descrevem, de forma satisfatória, o comportamento da série temporal em estudo. Em certos casos a escolha do “melhor modelo” acaba sendo relativamente fácil, por existir um modelo díspar dos demais. Contudo, existem casos em que não é tão fácil assim, e nesses casos é necessário o uso de um ou mais critérios de seleção de modelos. Estes critérios são usualmente baseados em estatísticas construídas a partir dos resíduos de um modelo ajustado ou baseados nos erros de previsão produzidos por esse mesmo modelo. Os critérios mais utilizados se baseiam no máximo da função de verossimilhança e, serão apresentados abaixo.

### **Critério de informação de Akaike (AIC).**

O AIC, proposto por Akaike (1973), é uma medida de qualidade do ajuste de um modelo estatístico dada por:

$$AIC(k) = 2\log\hat{L} + 2k \quad (4.11.8.1)$$

em que:  $\hat{L}$  é a função de verossimilhança maximizada e  $k$  é o número de parâmetros estimáveis (graus de liberdade).

Hurvich e Tsai (1989) refinaram ainda mais esta estimativa propondo um modelo de correção de viés denominado de  $AIC(k)_c$ . A correção é de uso particular quando o tamanho da amostra é pequeno, ou quando o número de parâmetros ajustados é uma fração moderada a grande do tamanho da amostra. Segundo Hurvich e Tsai (1991), o viés de  $AIC(k)_c$  é tipicamente menor, muitas vezes dramaticamente menor do que o da  $AIC(k)$ .

A formulação do critério corrigido é:

$$AIC(k)_c = AIC(k) + \frac{2k(k+1)}{n-k-1} \quad (4.11.8.2)$$

em que:  $n$  é o tamanho da amostra. Se  $n$  for grande em relação a  $k$ , essa correção se torna insignificante e  $AIC(k)$  é suficiente.

Segundo o critério de informação de Akaike, a ordem ótima de um modelo corresponde ao valor de  $k$  que minimiza o  $AIC(k)$  ou  $AIC(k)_c$ , ou seja, o “melhor modelo” é o que apresenta o menor valor para o  $AIC(k)$  ou  $AIC(k)_c$ .

### **Critério de informação bayesiana (BIC)**

O critério de informação bayesiana, muitas vezes referido como critério de informação Schwarz (SIC), foi proposto por Schwarz (1978). Este critério de seleção de modelos faz parte de uma classe de modelos paramétricos com diferentes números de parâmetros. Quando os parâmetros

de um determinado modelo são estimados empregando a estimação de máxima verossimilhança, é possível aumentar a verossimilhança empregando um número maior de parâmetros, podendo resultar em superestimação. O BIC resolve este problema ao introduzir uma penalidade para o número de parâmetros do modelo.

Define-se a grandeza de um modelo com  $k$  parâmetros e  $n$  observações como:

$$\text{BIC}(k) = n \ln(\hat{\sigma}_\varepsilon^2) - (n - k) \ln\left(1 - \frac{k}{n}\right) + k \ln(n) + k \ln\left[\frac{1}{k}\left(\frac{\hat{\sigma}_x^2}{\hat{\sigma}_\varepsilon^2} - 1\right)\right] \quad (4.11.8.3)$$

em que:  $\sigma_\varepsilon^2$  é a variância do erro,  $\hat{\sigma}_\varepsilon^2$  é a estimativa de máxima verossimilhança de  $\sigma_\varepsilon^2$  e  $\hat{\sigma}_x^2$  a variância residual do modelo.

No caso em que o número de parâmetros  $k$  for pequeno relativamente ao número de observações  $n$ , considera-se a aproximação

$$\left[-(n - k) \ln\left(1 - \frac{k}{n}\right)\right] \approx k \quad (4.11.8.4)$$

e assim obtém a expressão

$$\text{BIC}(k) = n \ln(\hat{\sigma}_\varepsilon^2) + k(1 + \ln n) + k \ln\left[\frac{1}{k}\left(\frac{\hat{\sigma}_x^2}{\hat{\sigma}_\varepsilon^2} - 1\right)\right] \quad (4.11.8.5)$$

Assim como no  $\text{AIC}(k)$  ou  $\text{AIC}(k)_c$ , quanto menor for o valor do  $\text{BIC}(k)$  melhor será o modelo em questão.

#### **4.12 Trigonometric Box-Cox transformation ARMA erros, Trend and Seasonal components (TBATS)**

O TBATS, proposto por De Livera *et al.* (2011), é um método de previsão e decomposição de séries temporais para lidar com dados que apresentam padrões sazonais complexos. Método este que é semelhante ao de amortecimento exponencial no contexto de espaço de estados (Hyndman *et al.*, 2008), mas que incorpora uma transformação do tipo Box-Cox (1964) para lidar com a não-linearidade e a heterocedasticidade (Equação 4.12.1), um filtro do tipo ARMA

para os resíduos (Equação 4.12.6) e funções trigonométricas como séries de Fourier para tratar os padrões de sazonalidade (Equações 4.12.7 a 4.12.9).

Segundo os autores, as principais vantagens do método sobre outras formulações são: os parâmetros podem ser calculados por métodos de máxima verossimilhança e a capacidade em lidar com padrões complexos de sazonalidade onde existem múltiplos ciclos.

A formulação do método está representada pela formulação que segue:

$$y_t^{(w)} = \begin{cases} (y_t^{(w)} - 1)/w & w \neq 0 \\ \log y_t & w = 0 \end{cases} \quad (4.12.1)$$

$$y_t^{(w)} = l_{t-1} + \phi b_{t-1} + \sum_{i=1}^T s_{t-m_i}^{(i)} + d_t \quad (4.12.2)$$

$$l_t = l_{t-1} + \phi b_{t-1} + \alpha d_t \quad (4.12.3)$$

$$b_t = (1 - \phi)b + \phi b_{t-1} + \beta d_t \quad (4.12.4)$$

$$s_t^{(i)} = s_{t-m_i}^{(i)} + \gamma_i d_t \quad (4.12.5)$$

$$d_t = \sum_{i=1}^p \varphi_i d_{t-i} + \sum_{i=1}^q \theta_i \epsilon_{t-i} + \epsilon_t \quad (4.12.6)$$

$$s_t^{(i)} = \sum_{j=1}^{k_i} s_{j,t}^{(i)} \quad (4.12.7)$$

$$s_{j,t}^{(i)} = s_{j,t-1}^{(i)} \cos \lambda_j^{(i)} + s_{j,t-1}^{*(i)} \sin \lambda_j^{(i)} + \gamma_1^{(i)} d_t \quad (4.12.8)$$

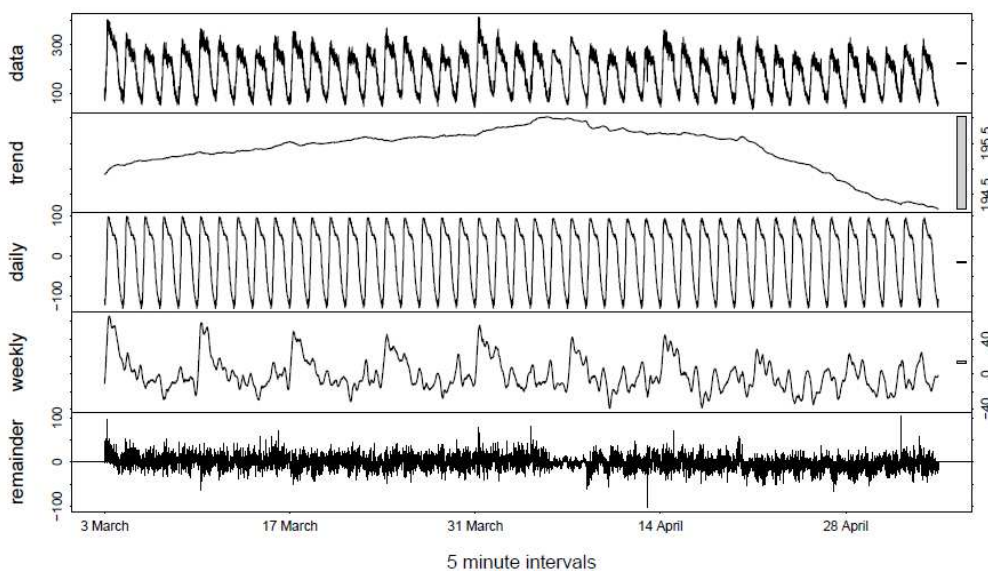
$$s_{j,t}^{*(i)} = -s_{j,t-1}^{(i)} \sin \lambda_j^{(i)} + s_{j,t-1}^{*(i)} \cos \lambda_j^{(i)} + \gamma_2^{(i)} d_t \quad (4.12.9)$$

em que:  $y_t^{(w)}$  denota a transformação Box-Cox com parâmetro  $w$ ;  $m_1, \dots, m_T$  denotam os períodos sazonais;  $l_t$  é o nível local no período  $t$ ;  $b$  é a tendência de longo prazo;  $b_t$  é a tendência de curto prazo no período  $t$ ;  $s_t^{(i)}$  representa o  $i$ -ésimo componente sazonal no período  $t$ ;  $d_t$  é

um processo  $ARMA(p, q)$ ;  $\varphi$  e  $\theta$  são os parâmetros da parte autorregressiva e de médias móveis do modelo ARMA, respectivamente;  $\epsilon_t$  é um ruído branco gaussiano com média zero e variância constante ( $\sigma^2$ );  $\alpha, \beta, \gamma_i$  são os parâmetros de suavização para  $i = 1, \dots, T$ ;  $T$  é a quantidade de padrões sazonais;  $k_i$  é o número de harmônicas necessárias para a  $i$ -ésima componente sazonal  $s_t^{(i)}$ ;  $\lambda_j^{(i)} = 2\pi j/m_i$ ; e  $s_{j,t}^{*(i)}$  é uma variável auxiliar para o cálculo das componentes sazonais.

Dessa forma, a ordem do modelo TBATS pode ser então representada pela notação  $TBATS(\omega, \emptyset, p, q, \{m_1, k_1\}, \{m_2, k_2\}, \dots, \{m_T, k_T\})$ . Exemplificando, um modelo com sazonalidade mensal pode ser descrito como  $TBATS(1, 1, 0, 0, 12)$ . Este modelo possui dois ciclos, uma transformação Box-Cox e um filtro autorregressivo de ordem 1 para os resíduos e pode ser descrito como  $TBATS(\omega, 1, 1, 0, m_1, m_2)$ . Os detalhes do procedimento de estimação dos parâmetros podem ser encontrados em De Livera *et al.* (2011).

A Figura 4.32 apresenta um exemplo de decomposição de uma série temporal de ligações em um *call center* com frequência de aquisição de 5 minutos, onde existe um ciclo diário e um ciclo semanal. A decomposição obtida empregando o TBATS mostra claramente forte sazonalidade diária e semanal. A sazonalidade diária é relativamente estável enquanto o padrão sazonal semanal evolui ao longo do tempo.



**Figura 4.32** - Decomposição empregando o TBATS para uma série temporal de ligações de um *call center* com frequência a cada 5 minutos (De LIVERA *et al.*, 2011)

### 4.13 Naive Bayes

O método *Naive* (do inglês ingênuo), apresentado a seguir, será aplicado no estudo em dois momentos. O primeiro é para gerar os dados faltantes nas séries temporais ocasionados por problemas de coleta de dados. O segundo diz respeito ao cálculo do custo computacional (CC). Este custo será utilizado para avaliar a complexidade dos modelos e da efetiva possibilidade de implementação no mundo real.

O método conhecido como *Naive Bayes* é usado para estimar o nível de modelos não sazonais sem tendência e pode ser escrito como:

$$Z_t = \mu + \varepsilon_t \quad (4.13.1)$$

Para esses modelos a previsão dos valores futuros da série é a própria estimativa do nível, chamado de  $a$ :

$$\hat{Z}_{t+k} = \hat{\mu}_t = \hat{a}_t \quad (4.13.2)$$

Há várias maneiras de obter a estimativa de  $a_t$ . No método mais simples, que agrega o mínimo de informação possível à previsão, e que será utilizado para imputar os dados faltantes na série de dados, a estimativa para a previsão no instante  $t + 1$  é dada no instante  $t$  pelo último valor observado  $Z_t$ :

$$\hat{Z}_{t+1|t} = Z_t \quad (4.13.3)$$

Este método pode ser utilizado para a validação de outros modelos e para calcular o custo computacional (CC). Contudo, para este fim será empregado o previsor ingênuo chamado de *Naive-24*. Isto é, a demanda em uma hora do dia será igual à da mesma hora do dia anterior, podendo ser escrita como:

$$\hat{Z}_t = Z_{t-24} \quad (4.13.4)$$

#### **4.14 Métodos de similaridade baseados em distância**

Segundo Aguiar (2016), o problema de agrupamento é um dos mais importantes no campo de aprendizagem da máquina, especificamente na aprendizagem não supervisionada. Devido a esta importância existem diversos métodos de medida de similaridade baseados no cálculo de distância entre objetos após serem devidamente modelados como séries temporais.

No trabalho de Cha (2007), são apresentadas as equações que definem uma série de medidas que podem ser aplicadas para resolverem muitos problemas de reconhecimento de padrões, classificação e agrupamento. Inúmeras medidas de distância/similaridade foram apresentadas sendo aplicáveis para se comparar funções de densidade de probabilidade. De acordo com Giusti e Batista (2013), essas distâncias poderão ser adaptadas para comparação de séries temporais. Ainda segundo esses autores, existem na literatura inúmeros métodos que dependem de uma medida de dissimilaridade como a principal maneira de comparar objetos. Um exemplo notável é o algoritmo de classificação  $k$ -NN. Frequentemente esses métodos superam os métodos mais complexos em tarefas como classificação, agrupamento, previsão e detecção de anomalias (GIUSTI e BATISTA, 2013). Todos esses métodos usam a função de distância ou dissimilaridade, sendo a Distância Euclidiana (DE) e o *Dynamic Time Warping*<sup>30</sup> (DTW), as duas medidas de dissimilaridade mais empregadas na literatura (GIUSTI e BATISTA, 2013). Os resultados obtidos por estes autores mostram que o *Complex Invariant Distance* DTW (CID-DTW), proposto por Batista *et al.* (2011), supera o DTW, e que somente o CID-DTW, DTW e CID superam a DE. Os resultados obtidos evidenciam ainda mais a qualidade do DTW como medida de distância adequada para a classificação de séries temporais por similaridade, visto que nenhuma das medidas superou o DTW com diferença estatisticamente significativa.

##### **4.14.1 Distância Euclidiana (DE) e *Dynamic Time Warping* (DTW)**

A DE, também denominada métrica pitagórica, é a mais básica, e talvez a mais importante medida de distância existente, uma vez que outros métodos mais sofisticados são uma derivação da DE. Este método consiste em comparar todos os pontos da série com o ponto de mesma

---

<sup>30</sup> Para maiores detalhes ver: SAKOE, H.; CHIBA, S. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*. v. 26, n.1, p.43-49, 1978.

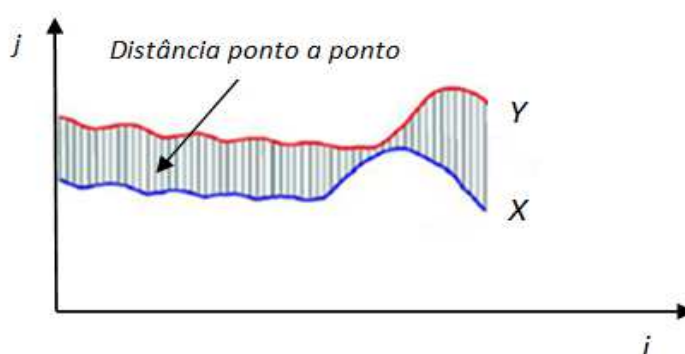
ordem na série temporal. Posteriormente somam-se as diferenças entre as séries e obtém-se a dissemelhança da distância euclidiana.

Dada duas séries de tempo  $X = (x_1, x_2, \dots, x_n)$  e  $Y = (y_1, y_2, \dots, y_n)$ , pode-se calcular a distância euclidiana entre os pontos X e Y como o somatório de todos os pontos que compõem X e Y.

$$DE(X, Y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (4.14.1.1)$$

Este método é o mais rápido e apresenta um crescimento linear, ao contrário do método de deformação dinâmica do tempo (*Dynamic Time Warping*) que apresenta um crescimento quadrático. Outra característica do método diz respeito a sua sensibilidade a distorções, compressões e defasagens, impondo que as duas séries tenham o mesmo comprimento (KEOGH e PAZZANI, 1999, 2000).

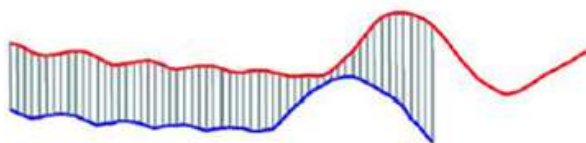
Na Figura 4.33 mostra-se que cada ponto da série X tem uma única distância com o ponto da série Y e vice-versa, mantendo uma relação de um para um.



**Figura 4.33** - Dada as séries temporais X e Y, para cada ponto de X, o cálculo da distância é realizado apenas uma única vez para apenas um ponto em Y.

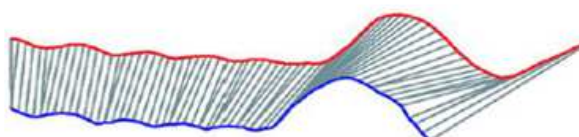
Para séries temporais que possuem o mesmo tamanho a distância euclidiana produzirá um bom resultado final. Contudo, em casos onde as séries apresentam tamanhos diferentes, este método não será a melhor alternativa de medida de similaridade, pois a DE ignora os pontos da série

temporal de maior tamanho que excedem os pontos da série de menor tamanho, conforme ilustrado na Figura 4.34.



**Figura 4.34** - Alinhamento de séries com diferentes tamanhos empregando a DE.

Visando resolver este problema de dimensionamento entre duas séries temporais de tamanhos diferentes, o DTW apresenta uma abordagem interessante (Figura 4.35). Este algoritmo minimiza os possíveis efeitos de distorção no tempo e permitindo uma transformação elástica da série temporal, isto é, este algoritmo possibilita a detecção de similaridades entre padrões que podem estar defasadas temporalmente ou com uma compressão diferente.



**Figura 4.35** - Alinhamento de séries com diferentes tamanhos empregando a DTW.

## 4.15 *Dynamic Time Scan Forecasting (DTSF)*

*“Nothing is more practical than a good theory”.*

Vladimir Vapnik

### 4.15.1 *Background histórico*

Os modelos de previsão bem-sucedidos baseiam-se em abordagens de *soft computing*, como redes neurais, sistemas difusos, máquinas de vetor de suporte e sistemas híbridos. Este trabalho propõe o emprego do novo modelo que verifica a série temporal procurando por janelas compatíveis, ou seja, partes da série passada que são semelhantes aos últimos valores observados dos quais a previsão é necessária. A partir das janelas compatíveis selecionadas, as seguintes observações são usadas como valores de previsão usando funções de similaridade. O

método chamado de *Dynamic Time Scan Forecasting* é notavelmente intuitivo e pode ser usado tanto como uma ferramenta exploratória para identificar padrões semelhantes nas séries temporais e como um modelo de previsão melhorado. Em comparação com abordagens de *soft computing*, o método proposto é extremamente rápido.

O método denominado como *Dynamic Time Scan Forecasting*, aqui chamado nomeado como *DTSF* (COSTA *et al.*, 2019), foi inspirado por estatísticas de varredura (GLAZ *et al.*, 2009). As estatísticas de digitalização compreendem uma classe de métodos estatísticos que digitalizam fluxos de dados para encontrar um comportamento anômalo. Foi introduzido por Joseph Naus em 1965 (NAUS, 1965) e estendido à vigilância epidemiológica usando dados espaciais (KULLDORFF, 1997), temporais e espaciais-temporais (KULLDORFF *et al.*, 1998; KULLDORFF, 2001). Resumidamente, uma janela de varredura com forma fixa, como uma circular de uma janela cilíndrica varre os dados, e uma estatística de teste é calculada para cada posição da janela. A posição com o maior valor da estatística de teste é um candidato potencial para um comportamento anômalo. A inferência estatística é obtida usando simulações de Monte Carlo (MOONEY, 1997) sob a hipótese nula de que o fluxo de dados não apresenta dados anômalos. Mais informações sobre estatísticas de varredura são encontradas em Glaz *et al.* (2009).

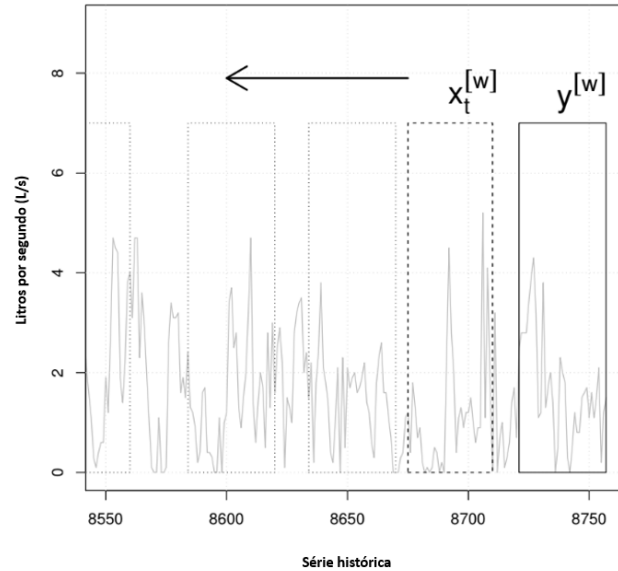
O método *DTSF* verifica uma série temporal usando uma janela fixa. O objetivo é encontrar a janela anterior nos dados que são semelhantes aos valores observados mais recentes. Portanto, uma estatística de teste, ou uma estatística de similaridade é calculada para cada janela. Além disso, uma função de similaridade é estimada para cada janela. Depois de detectar a maioria das janelas semelhantes, os valores de previsão são estimados usando as funções de similaridade e os valores observados que seguem as janelas selecionadas. Mais detalhes sobre *DTSF* são apresentados abaixo.

#### **4.15.2 Formulação do *Dynamic Time Scan Forecasting* (*DTSF*)**

O objetivo do *DTSF* inclui identificar padrões na série temporal que estão fortemente correlacionados com o vetor  $\mathbf{y}^{[w]}$  (vetor correspondente aos valores observados mais recente, onde o seu comprimento é dado por  $w$ ). Assim, o conjunto de vetores candidatos pode ser escrito como:

$$\mathbf{x}_t^{[w]} = [y_{t-w+1}; \dots; y_{t-w}] \quad (4.15.2.1)$$

em que,  $t = 1; \dots; N - 2.w$ . O limite superior da sequência de tempo ( $N - 2.w$ ) garante que o vetor  $\mathbf{x}_t^{[w]}$  não se sobrepõe com o vetor  $\mathbf{y}^{[w]}$ . A Figura 4.36 ilustra o método *DTSF*. Dado o último  $w$  valores observados, que compreende vetor  $\mathbf{y}^{[w]}$ , uma janela de varredura com o mesmo tamanho ( $\mathbf{x}_t^{[w]}$ ) é usada para verificar os valores anteriores das séries temporais.



**Figura 4.36** - Ilustração do procedimento de varredura da série de tempo *DTSF* (Adaptado de COSTA *et al.*, 2019)

O objetivo final do *DTSF* é fornecer uma previsão de  $K$  etapas da série temporal,  $y_{N+1}, \dots, y_{N+k}$ . Para atingir esse objetivo, o *DTSF* verifica as séries temporais para os melhores padrões  $\mathbf{x}_t^{[w]}$ . Os valores das séries temporais que seguem os melhores padrões são utilizados como valores de previsão:

$$y_{N+i} = f_{\mathbf{x}_t^{[w]}}(y_{t-w+i}) \quad (4.15.2.2)$$

em que  $f_{\mathbf{x}_t^{[w]}}$  é uma função que correlaciona os elementos do vetor  $\mathbf{x}_t^{[w]}$  e os elementos do vetor  $\mathbf{y}^{[w]}$ .

Uma primeira restrição pode ser imposta em  $k$ :  $1 \leq k \leq w$ . Essa restrição garante que, se a janela da série temporal mais correlacionada incluir os valores mais recentes, antes do vetor  $\mathbf{y}^{[w]}$ , os valores de previsão são uma função do vetor  $\mathbf{y}^{[w]}$ ,

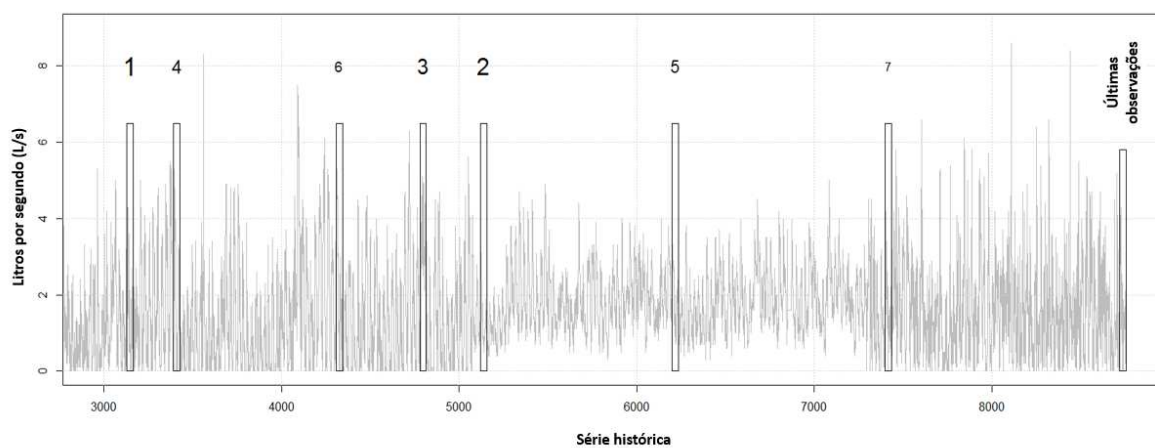
$$y_{N+i} = f_{\mathbf{x}_{N-2w}^{[w]}}(y_{N-w+i}) \quad (4.15.2.3)$$

Conforme observado nas equações 4.15.2.1 e 4.15.2.3, os valores de previsão dependem do comprimento da janela  $w$  e da função  $f_{\mathbf{x}_t^{[w]}}(\cdot)$ . Uma primeira proposta para a função  $f_{\mathbf{x}_t^{[w]}}(\cdot)$  é uma escala linear dos elementos do vetor  $\mathbf{x}_t^{[w]}$ , isto é, um modelo linear. Isso ocorre porque os valores anteriores podem ser supostamente semelhantes aos últimos valores, exceto para uma escala e/ou deslocamento de compensação. Assim, o método está buscando valores que possam ser semelhantes aos últimos valores, após a aplicação de uma função de similaridade.

Ao assumir a função de similaridade como um modelo linear, os parâmetros do modelo podem ser estimados para minimizar a soma dos quadrados entre os elementos do vetor  $\mathbf{y}^{[w]}$  e a equação linear:  $\beta_0^{[t]} + \beta_1^{[t]} \times \mathbf{x}_t^{[w]}$ . Além disso, a estatística de similaridade pode ser definida como o coeficiente de determinação  $R^2$  (MONTGOMERY, 2012):

$$R^2 = 1 - \frac{\sum_j (\mathbf{y}_j^{[w]} - \hat{\mathbf{y}}_j^{[w]})^2}{\sum_j (\mathbf{y}_j^{[w]} - \bar{\mathbf{y}}^{[w]})^2} \quad (4.15.2.4)$$

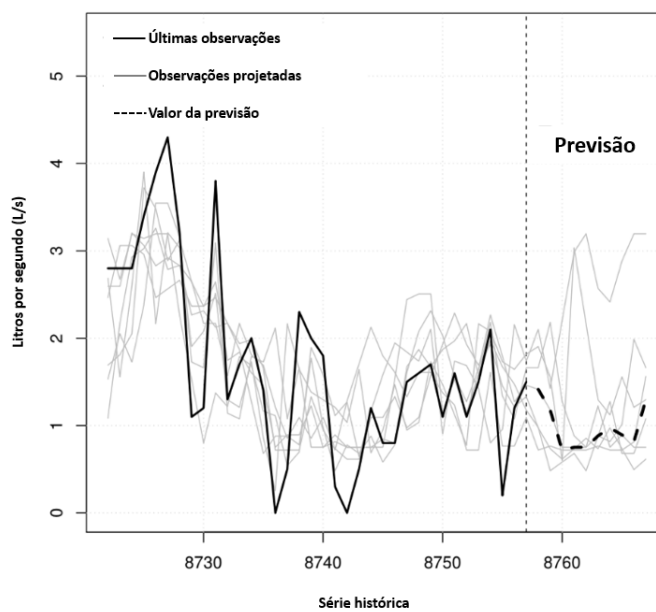
em que  $\mathbf{y}_t^{[w]}$  é o  $j$ -ésimo valor do vetor  $\mathbf{y}_j^{[w]}$  e  $\hat{\mathbf{y}}_j^{[w]}$  é o  $j$ -ésimo valor previsto usando a função linear estimada. Vale ressaltar que  $R^2$  está dentro do intervalo de unidade [0-1]. Se  $R^2 \rightarrow 1$  então os valores estimados são muito próximos dos valores observados, isto é, os valores observados passados localizados no tempo  $t$  são semelhantes aos últimos valores observados após a escala e a correção de deslocamento. O procedimento de digitalização é ilustrado na Figura 4.37 usando uma janela de comprimento 24 (horas). As 7 últimas janelas com estatísticas de alta similaridade ( $R^2$ ) são indicadas em retângulos. Para cada janela, um modelo linear (função de similaridade) foi estimado.



**Figura 4.37** - Exemplo de *DTSF* usando séries temporais de demanda de água (Adaptado de COSTA *et al.*, 2019)

Usando as funções de similaridade, os dados de acompanhamento das janelas selecionadas são usados como valores de previsão, conforme ilustrado na Figura 4.38. As estimativas de pontos são geradas usando uma função de agregação, como os valores médios. O método *DTSF* requer três parâmetros: o comprimento da janela de varredura, a especificação da função de similaridade e o número das melhores correspondências, ou seja, o número de janelas semelhantes encontradas nas séries temporais. A fim de melhorar a velocidade computacional, a função de similaridade linear é preferível, como equações lineares, quadráticas e cúbicas. O número de melhores correspondências pode ser selecionado dinamicamente usando, por exemplo, um limite para a estatística de similaridade. No presente estudo, o número de melhores correspondências (*match*) será previamente definidos.

Um requisito importante do método *DTSF* é a disponibilidade de uma grande série temporal. O *DTSF* é um método orientado a dados.



**Figura 4.38** - Previsão de estimativas usando as janelas mais semelhantes, selecionadas usando *DTSTF* (Adaptado de COSTA *et al.*, 2019)

## 5 MATERIAL E MÉTODOS

Os dados de vazão são monitorados por meio do dispositivo “*Data Logger*”, equipamento dotado de memória não volátil que realiza aquisição de dados com os mais variados tipos de sensores existentes. Com o *software LogChart II* é possível realizar a configuração de parâmetros do registrador, a coleta de dados no período de aquisição, a visualização destes dados em um gráfico temporal, a impressão deste gráfico, o salvamento dos dados de uma coleta em disco e muitos outros recursos. Estes equipamentos foram configurados para coletar os dados de vazão (L/s) a cada 5 minutos na saída da ETA-A e de 15 em 15 minutos no macromedidor que liga a ETA-B a ZA em estudo. Os dados coletados para o presente trabalho compreendem as vazões de 00:00 h do dia 01 de janeiro de 2015 até às 24:00 do dia 31 de dezembro de 2016.

Posteriormente, estes dados foram transformados na extensão “csv” (arquivos de texto com valores separados por vírgulas) e importados para o *software R* onde foram manipulados com vista à realização das previsões horárias da demanda de água. No período analisado houve dias em que não ocorreram coletas por falhas no dispositivo “*Data Logger*”. Estas falhas foram suprimidas empregando a imputação *Naive Bayes* na geração dos dados faltantes, possibilitando a realização da previsão. A escolha deste método se deve a sua simplicidade.

As principais ferramentas de estatísticas e de SC utilizadas neste trabalho encontram-se implementadas no *software R*. Os métodos rede neural autorregressiva não-linear (NNAR), Box e Jenkins (SARIMA), alisamento exponencial (ETS) e modelo de espaço de estado de suavização exponencial com transformação Box-Cox, erros ARMA, componente tendência e sazonal (TBATS) foram aplicadas empregando as funções *nnetar*, *auto.arima*, *ets*<sup>31</sup> e *tbats*, respectivamente, do pacote *Forecast* (HYDMAN e HANDAKAR, 2008; HYNDMAN *et al.*, 2017). Para todos os métodos deste pacote o treinamento empregará um único passo (HYNDMAN e ATHANASOPOULOS, 2014). O método ELM foi implementado utilizando a função *elm* do pacote *nnfor*<sup>32</sup> (Kourentzse, 2017). Já o modelo híbrido STL-ETS foi implementado empregando a função *hybridModel* do pacote *forecastHybrid* (SHAUB e ELLIS,

---

<sup>31</sup> O esquema heurístico de Hyndman *et al.* (2002) parece funcionar muito bem e computacionalmente é empregado no pacote *Forecast* (HYDMAN e HANDAKAR, 2008; HYNDMAN *et al.*, 2017), visto que segundo Makridakis *et al.* (2018)

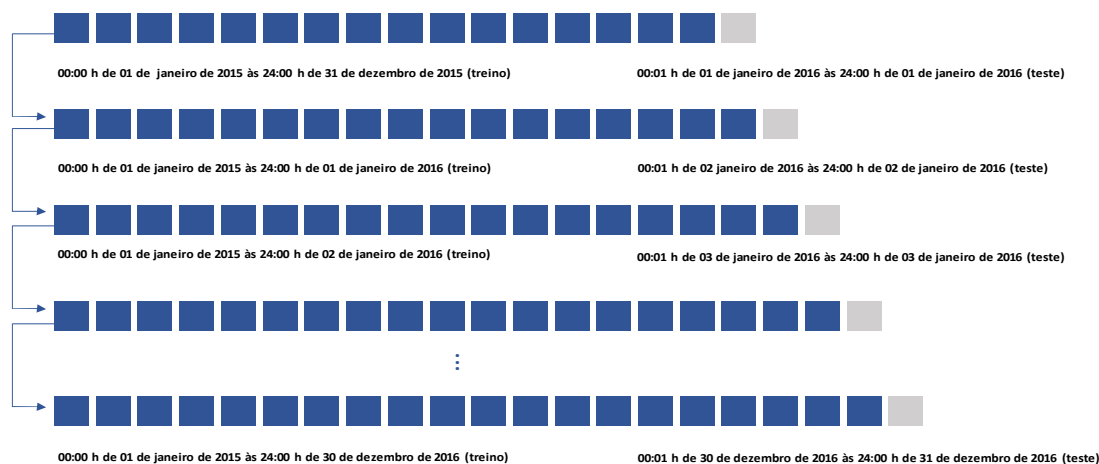
<sup>32</sup> Para maiores detalhes sobre este algoritmo ver Crone e Kourentzes (2010) e Kourentzes *et al.* (2014)

2019). Para o método DTSF utilizou a função *dtmf* do pacote *DTScanF* (COSTA e MINETTI, 2019).

Quando se trata de previsão em séries temporais, devido a correlação serial inerente e a potencial não estacionariedade da série, faz com que o uso da validação cruzada pareça problemática, uma vez que não leva em conta esses problemas (BERGMEIR e BENÍTEZ, 2012). Ainda, segundo estes autores, o problema das dependências dentro do treinamento e do conjunto de teste pode ser resolvido usando blocos de dados em vez de escolher os dados de forma aleatória. Na previsão tradicional o bloco final da série, reservado para teste, é denominado avaliação fora da amostra (TASHMAN, 2000). Para produzir as previsões fora da amostra, serão estimados os modelos empregando uma parte da amostra (treino), para em seguida, comparar com a demanda real não incluída na estimação. O período fora da amostra (teste) será utilizado para testar a capacidade preditiva dos modelos. Hyndman (2014), recomenda o uso da validação cruzada em séries temporais sempre que possível. A validação cruzada empregada no presente estudo concentra-se em um único horizonte de previsão para cada conjunto de teste, isto é, um passo à frente (24 horas).

A base de dados foi dividida em dois subconjuntos. O conjunto de treino empregará as médias horárias de vazão (L/s) de 01 de janeiro de 2015 das 00:00 h até às 24:00 do dia 31 de dezembro de 2015 e os dados de teste (fora da amostra) corresponderão as vazões horárias médias do dia 01 de janeiro de 2016 das 00:00 h até às 24:00 h. Para a previsão das próximas 24 horas do dia 02 de janeiro de 2016 serão utilizadas as vazões médias horárias de 00:00 h do dia 01/01/2015 até as 24:00 h do dia 01/01/2016 (treino) e das 00:00 h do dia 02/01/2016 até as 24:00 h do dia 02/01/2016 (teste); e assim sucessivamente.

A Figura 5.1 representa esquematicamente a recalibração de origem rolante, apresentada no sub-tópico (4.4.6), sendo o procedimento preferível (TASHMAN, 2000).



**Figura 5.1** – Exibição esquemática da recalibração de origem rolante

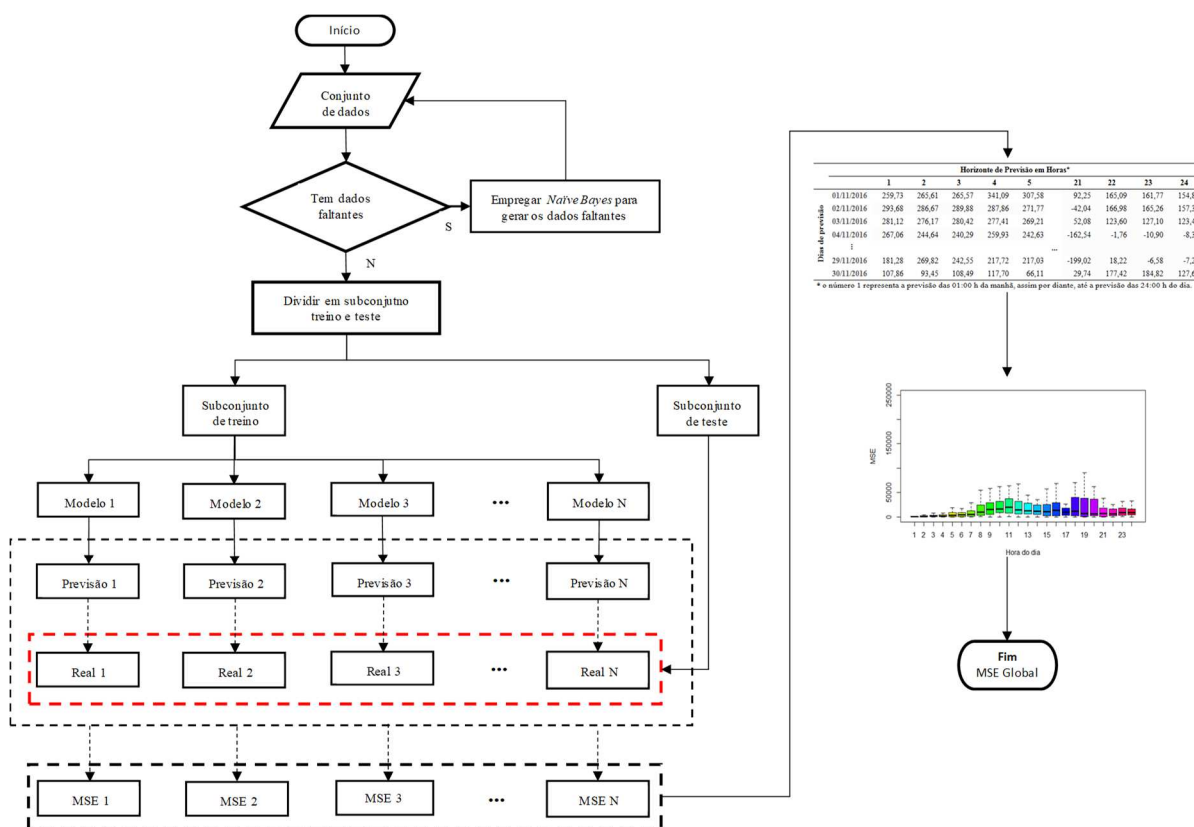
As previsões um passo à frente (horizonte de 24 horas) serão realizadas para todos os dias do ano de 2016 empregando os diversos métodos a serem testados. Posteriormente, será construída uma matriz de erros com dimensão  $(i \times j)$ , para qual  $i$  são os dias de previsão e  $j$  os horizontes de previsão em horas, conforme exemplo apresentado na Tabela 5.1. Os erros serão obtidos pela diferença entre os valores fora da amostra e os valores previstos.

**Tabela 5.1** – Exemplo de matriz de erros para os dias e horizonte de previsão

	Horizonte de Previsão em Horas*									
	1	2	3	4	5	21	22	23	24	
Dias de previsão	01/11/2016	259,73	265,61	265,57	341,09	307,58	92,25	165,09	161,77	154,89
	02/11/2016	293,68	286,67	289,88	287,86	271,77	-42,04	166,98	165,26	157,39
	03/11/2016	281,12	276,17	280,42	277,41	269,21	52,08	123,60	127,10	123,45
	04/11/2016	267,06	244,64	240,29	259,93	242,63	-162,54	-1,76	-10,90	-8,38
	⋮						⋮			
	29/11/2016	181,28	269,82	242,55	217,72	217,03	-199,02	18,22	-6,58	-7,21
	30/11/2016	107,86	93,45	108,49	117,70	66,11	29,74	177,42	184,82	127,66

\* o número 1 representa a previsão à 01:00 h da manhã, assim por diante, até a previsão das 24:00 h do dia.

Todo este procedimento pode ser resumido no fluxograma apresentado na Figura 5.2.



**Figura 5.2** – Fluxograma proposto para análise e comparação de todos os métodos a serem avaliados.

Este procedimento será empregado para os diversos métodos que a princípio serão analisados:

- i. *Naive Bayes*
- ii. *DTSF (Dynamic Time Scan Forecasting)*
- iii. *ELM (Extreme Learning Machine)*
- iv. *BNNAR (Rede neural autorregressiva não-linear com bootstrap)*
- v. *SARIMA (Box e Jenkins)*
- vi. *TBATS (Trigonometric Box-Cox transformation, ARMA errors, Trend, and Seasonal Components)*
- vii. *ETS (Alisamento exponencial)*
- viii. *STL-ETS (Modelo híbrido empregando filtro STL e o ETS)*

Um grande problema na literatura de *machine learning* é que a maioria dos estudos publicados fornecem previsões e afirmam que as acurácias são satisfatórias sem compará-los com métodos

estatísticos simples ou mesmo com previsão ingênua (*Naive Bayes*) (MAKRIDAKIS *et al.*, 2018).

## 5.1 Métricas de avaliação

Segundo Pellegrini (2012), durante os últimos anos inúmeras medidas para avaliar o ajuste de séries temporais e a acurácia da previsão foram propostas. Entretanto, seu uso indiscriminado pode levar a conclusões errôneas. Além disso, uma grande parte é inadequada quando se deseja medir a capacidade preditiva em uma grande quantidade de séries temporais. O erro médio absoluto em escala (MASE)<sup>33</sup> deve então ser utilizado. As métricas de avaliação podem ser caracterizadas em: medidas escala dependentes, medidas baseadas em erros percentuais, medidas baseadas em erros relativos, medidas relativas e medidas de erro de escala.

A seguir serão apresentadas as métricas utilizadas no presente estudo. A precisão de cada modelo será avaliada empregando as métricas de custo do erro quadrático médio (MSE), da raiz do erro quadrático médio (RMSE), do erro absoluto médio (MAE), do erro percentual absoluto médio (MAPE) e do *Model Fitting* (MF) para cada um dos horizontes e dias de previsão.

As medidas de erro MSE, RMSE, MAE e MAPE expõem as seguintes vantagens e desvantagens. O MAE e o MSE são bastantes eficazes na avaliação do erro total absoluto de previsão, mas são altamente sensíveis a erros extremos, o RMSE possui propriedades semelhantes às do MSE, mas é consideravelmente menos sensível a erros extremos e é mais estável. Já o MAPE, além de ser eficaz, é independente da escala de medida (ADHIKARI, 2015). Outra vantagem deste método é que ele fornece um rápido entendimento devido a sua representação em termos percentuais (%). Maiores detalhes sobre a estatísticas de erros, equações (5.1.1) a (5.1.4), podem ser obtidos em (MONTGOMERY *et al.*, 2012).

As funções de custo empregadas têm as seguintes formulações:

$$MSE = \frac{1}{n} \sum_{i=1}^n (O_i - P_i)^2 \quad (5.1.1)$$

---

<sup>33</sup> Métrica proposta formalmente por Hyndman e Koehler (2006)

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (O_i - P_i)^2} \quad (5.1.2)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |O_i - P_i| \quad (5.1.3)$$

$$MAPE = \frac{1}{n} \sum_{i=1}^n |(O_i - P_i)/O_i| \times 100 \quad (5.1.4)$$

em que  $O_i$  valor observado da demanda;  $P_i$  valor previsto da demanda; e  $n$  é o horizonte de previsão.

O *Model Fitting* (MF) também foi utilizado como uma medida que expressa o ajuste dos dados históricos ao modelo. MF representa o MSE das previsões de ajuste dos  $n - k$  modelos, normalizado pela média das séries temporais em análise. O MF é dado por:

$$Model\ Fitting\ (MF) = \frac{n \sum_{i=1}^n (O_i - P_i)^2}{(\sum_{i=1}^n P_i)^2} * 100\% \quad (5.1.5)$$

O custo computacional (CC) também foi usado para determinar o tempo necessário para treinar um modelo e usá-lo para previsões. O CC representa o tempo computacional médio requerido para prever uma série temporal empregando um determinado modelo dividido pelo tempo correspondente utilizado pelo método *Naive* para alcançar a mesma tarefa (MAKRIDAKIS *et al.*, 2018), ou seja, é a proporção do tempo computacional do modelo avaliado em relação ao método ingênuo.

Já o CC será dado pela Equação (5.1.6) que segue:

$$CC = \frac{\text{Tempo decorrido no modelo a ser avaliado}}{\text{Tempo decorrido na previsão empregando o Naive}} \quad (5.1.6)$$

O tempo computacional foi estimado usando um sistema com as seguintes características: CPU com processador da 7ª Geração Intel® Core™ i7-7500U @3.5GHz, memória de 16GB DDR4, processador baseado em 64x.

## **5.2 Intervalo de confiança (I.C.) empírico**

Os intervalos de confiança (I.C.), para cada um dos métodos avaliados, foram calculados empiricamente utilizando o seguinte critério. Inicialmente, para cada hora do dia foram calculados os erros históricos de de predição. Utilizando os erros históricos, foram definidos os seus respectivos percentis para cada hora do dia. O limite inferior de predição será calculado adicionando o percentil 5% (0,05) dos erros históricos ao valor previsto. De forma similar, o limite superior de predição será calculado adicionando o percentil 95% (0,95) dos erros históricos ao valor previsto, ou seja:

Limite Inferior (L.I.):  $\hat{y} +$  percentil 5% dos erros históricos e

Limite Superior (L.S.):  $\hat{y} +$  percentil 95% dos erros históricos,

implicando em um I.C. com aproximadamente 90% de probabilidade.

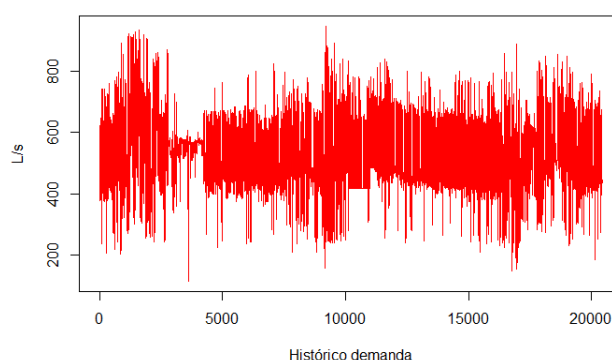
## 6 RESULTADOS E DISCUSSÃO

Neste capítulo serão descritos os resultados dos modelos de previsão empregados no presente estudo utilizando os dados da ZA expostos no tópico 5. O principal objetivo desse estudo, e discutido neste tópico, é comparar diferentes métodos estatísticos e de aprendizagem de máquina no tocante a previsão de demanda no curto prazo para sistemas de abastecimento de água urbano, verificando o desempenho de tais modelos em realizar prognósticos e compará-los, tanto na capacidade de generalização quanto no custo computacional demandado.

### 6.1 Estudo de caso

A ZA estudada está localizada na região centro sul de uma capital do Sudeste brasileiro e atende uma população de aproximadamente 230 mil pessoas, podendo ser comparada com uma cidade de médio porte. Esta região é predominantemente residencial (com diversos estratos sociais) e possui um amplo comércio. Segundo informações da concessionária responsável, a ZA em setembro de 2017, apresentava 35.710 ligações, 98.449 economias e uma vazão média distribuída de 861,05 L/s. Os volumes micromedido e o distribuído para o referido mês foram de 1.468.629 e 2.306.242, respectivamente, totalizando uma perda de 36,32%.

Na Figura 6.1 apresenta-se a demanda horária média desta ZA, onde é possível observar que a demanda permaneceu em média estável ao longo do período analisado, e na Tabela 6.1 as respectivas estatísticas descritivas.



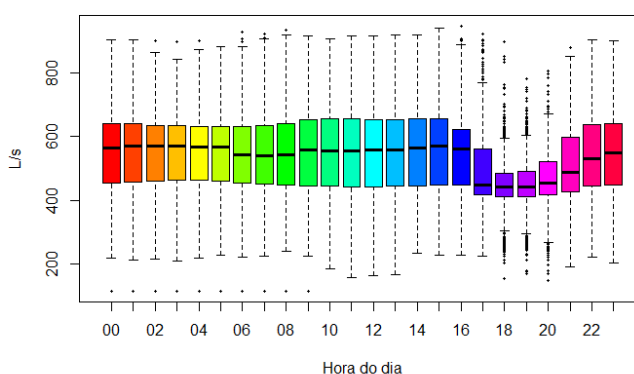
**Figura 6.1** – Série histórica de água demandada desde 00:00 de 01 de janeiro de 2015 às 24:00 de 31 de dezembro de 2016.

Na Tabela 6.1 são apresentadas as medidas descritivas da série histórica utilizada no presente estudo. A mediana de vazão, para o período analisado, foi de 522,81 (L/s) com um coeficiente de variação de 23,08%.

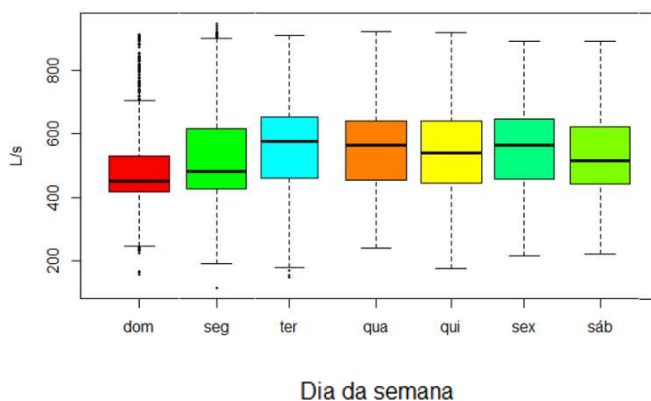
**Tabela 6.1** – Medidas descritivas da série histórica de demanda

<b>Estatística</b>	<b>Unidades</b>
Média	533,6307
Mediana	522,8083
Desvio Padrão	123,1604
Mínimo	116,5743
Máximo	947,1833
Primeiro Quartil	438,4812
Terceiro Quartil	629,6417
Coeficiente de variação	23,08%

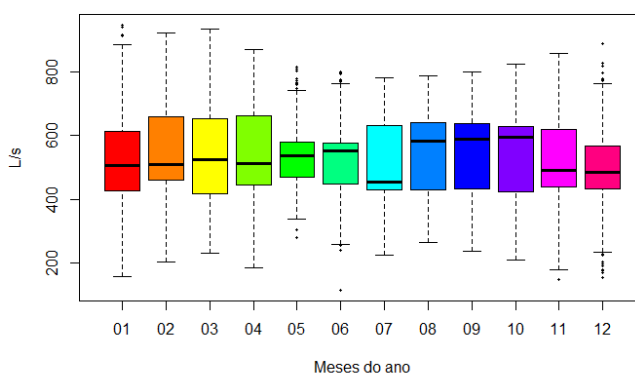
Ao analisar os dados horários, Figuras 6.2 a 6.4, observa-se que ao longo do dia a produção mantém-se estável variando somente nos horários de “ponta” onde o custo de energia elétrica é mais elevado. Contudo, uma dispersão é observada nas primeiras horas do dia. No tocante aos dias da semana, o domingo é o que possui menor consumo e maior dispersão, seguido pela segunda feira. Com relação aos meses do ano observa-se que no outono e no inverno são os períodos onde ocorrem os menores consumos e as maiores dispersões, seguido pelo mês de dezembro. O mês de julho possui a menor mediana, podendo ser associado ao período de férias escolares.



**Figura 6.2** – *Boxplot* dos dados para as 24 horas do dia empregando a vazão (L/s) no período de 00:00 de 01 de janeiro de 2015 às 24:00 de 31 de dezembro de 2016.

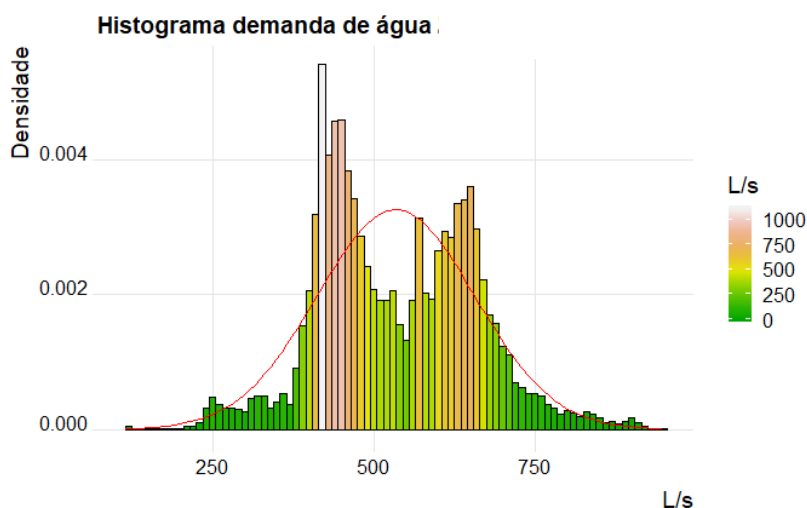


**Figura 6.3** – *Boxplot* dos dados para os dias da semana empregando a vazão (L/s) no período de 00:00 de 01 de janeiro de 2015 às 24:00 de 31 de dezembro de 2016.



**Figura 6.4** – *Boxplot* dos dados por mês empregando a vazão (L/s) no período de 00:00 de 01 de janeiro de 2015 às 24:00 de 31 de dezembro de 2016.

Observa-se, na Figura 6.5, que a demanda de água segue uma distribuição bimodal, isto é, uma distribuição de probabilidade contínua com duas modas diferentes. Estas modas aparecem como picos distintos (máximo local) na função de densidade de probabilidade da demanda de água da ZA em L/s.



**Figura 6.5** – Histograma e densidade de probabilidade da demanda de água da ZA em L/s.

## 6.2 Aplicação dos métodos de previsão

Um grande problema na literatura de *machine learning* é que a maioria dos estudos publicados fornecem previsões e afirmam que as acurácias são satisfatórias sem compará-los com métodos estatísticos simples ou mesmo com previsão ingênua (*Naive Bayes*) (MAKRIDAKIS *et al.*, 2018).

A seguir serão apresentados e discutidos os diversos modelos avaliados no presente estudo. O primeiro método avaliado foi o Box e Jenkins. Os ajustes de todos os modelos empregados no presente estudo estão apresentados no Anexo II.

### 6.2.1 Box e Jenkins

A modelagem Box e Jenkins foi realizada em três fases, conforme explicitado no tópico (4.6.9). Para a etapa de identificação empregou-se a seleção automática dos parâmetros  $(p, d, q) \times (P, D, Q)$  utilizando a função “*auto.arima*” do pacote *Forecast* (HYNDMAN *et al.*, 2017) do programa R. O primeiro parâmetro identificado foi o número de diferenciações necessárias para tornar a série temporal estacionária por meio do teste de raiz unitária KPSS. Posteriormente, foram determinados os parâmetros AR e MA( $p, q$ ) e os operadores sazonais dos parâmetros AR e MA de ordem  $(P, Q)$ . O algoritmo proposto por Hyndman e Khandakar

(2008) percorreu todo o espaço de busca com o objetivo de identificar os possíveis modelos e selecionar, empregando o critério AICc, o modelo que apresentou o menor critério. Este procedimento empregou o algoritmo *stepwise* com vista a reduzir o esforço computacional sem prejudicar a seleção do modelo mais parcimonioso para realizar a previsão.

O modelo selecionado, para os meses de janeiro a setembro de 2016, foi o SARIMA, (1,1,1)(0,0,2)[24], o que equivale afirmar que o modelo foi formado por um parâmetro autorregressivo, uma diferenciação para tornar a série estacionária, e um parâmetro de média móvel, além de dois parâmetros de média móvel de ordem sazonal, com uma sazonalidade de 24 horas. Contudo, para os meses de outubro a dezembro de 2016 o modelo selecionado foi o SARIMA (4,1,0)(2,0,0)[24]. Após a identificação, determinaram-se os valores dos parâmetros por meio do método de verossimilhança<sup>34</sup>. Os resultados dos parâmetros estimados dos modelos SARIMA estão apresentados na Tabela A.III.1.

Os diagnósticos dos resíduos, das funções de auto correlação residual, dos histogramas dos resíduos, dos gráficos de probabilidade Quantil-Quantil e das raízes dos polinômios que garantem a estacionariedade e a invertibilidade dos modelos SARIMA estão presentes nos Anexo III na Tabela A.III.2.

Os gráficos dos diagnósticos mostram a série dos resíduos, a função de autocorrelação residual (FAC) e o histograma dos resíduos. Os resultados obtidos demonstram que os desvios, tanto do modelo SARIMA (1,1,1)(0,0,2)[24] quanto do modelo SARIMA (4,1,0)(2,0,0)[24] não apresentam distribuição normal, com média zero e variância constante, o que é confirmado pelo teste de normalidade de Jarque Bera<sup>35</sup>(JB) (Tabela A.III.4) e pelo gráfico de probabilidade Quantil-Quantil (Q-Q) (Tabela A.III.2). A não normalidade observada ocorre pela curtose leptocúrtica<sup>36</sup>. Este gráfico é uma ferramenta muito útil para checar a adequação de distribuição de frequência dos dados a uma distribuição de probabilidades, usada para verificar se os resíduos apresentam distribuição normal. O gráfico Q-Q é melhor que o histograma e que a

---

<sup>34</sup> Para maiores detalhes ver Morettin e Toloi (2004).

<sup>35</sup> Trata-se de um teste assintótico, onde o procedimento do teste consiste em calcular os valores de assimetria e curtose da amostra e utilizar a seguinte estatística de teste:  $JB = T \left( \frac{\hat{\alpha}_1}{6} + \frac{(\hat{\alpha}_2 - 3)^2}{24} \right)$ , em que  $\hat{\alpha}_1$  e  $\hat{\alpha}_2$  são, respectivamente, os coeficientes amostrais de assimetria e de curtose e T é o tamanho da amostra. Este teste é utilizado para testar se a série tem distribuição normal baseada nas diferenças entre assimetria e curtose da distribuição da série em relação à normal. Segundo Granger e Newbold (1976), a heterocedasticidade não afeta a adequação da previsão, pois não implica em estimadores viesados, como no caso de uma regressão linear múltipla.

<sup>36</sup> Diz-se para as distribuições com caudas pesadas ou mais afunilada e concentrada que a distribuição normal.

distribuição acumulada empírica, pois é mais fácil verificar se uma reta se ajusta aos pontos do que se uma curva de densidade se ajusta a um histograma ou que uma curva de probabilidade acumulada se ajusta à acumulada empírica. Os gráficos Q-Q confirmam a leptocurtose (caudas pesadas) na medida em que os erros se assemelham a um “S” invertido, evidenciando a forma de uma distribuição mais “afilada” que a curva normal.

Outro ponto a ser abordado diz respeito à autocorrelação residual. O teste de Ljung-Box (L-B)<sup>37</sup> tem a hipótese nula de independência entre as observações considerando qualquer grau de defasagens entre elas. É comum usar este teste para verificar se os resíduos de um modelo de série temporal se assemelham ao ruído branco. Ver resultados na Tabela A.III.5.

Já a estabilidade do modelo foi testada pelo inverso das raízes dos polinômios característicos AR e MA. O inverso destas raízes, associadas aos componentes AR e MA, de fato pertencem ao círculo unitário, concluindo, portanto, que o modelo é estacionário e invertível (ver Tabela A.III.2).

Verifica-se, então, que os desvios no modelo não são normalmente distribuídos e são fortemente autocorrelacionados. Embora o modelo seja estacionário e invertível, necessita talvez de aplicar um pré-processamento para extrair toda informação contida nos dados.

**Tabela 6.2** - Modelos de previsão empregando SARIMA e suas respectivas métricas (RMSE, MAE e MAPE) para os meses de previsão

Mês	Modelo	RMSE	MAE	MAPE
jan/16	SARIMA(1,1,1)(0,0,2)[24]	214,723	162,299	0,381
fev/16	SARIMA(1,1,1)(0,0,2)[24]	173,828	130,608	0,266
mar/16	SARIMA(1,1,1)(0,0,2)[24]	155,864	115,328	0,231
abr/16	SARIMA(1,1,1)(0,0,2)[24]	149,156	112,378	0,217
mai/16	SARIMA(1,1,1)(0,0,2)[24]	131,096	98,502	0,193
jun/16	SARIMA(1,1,1)(0,0,2)[24]	134,915	98,710	0,202
jul/16	SARIMA(1,1,1)(0,0,2)[24]	135,242	98,178	0,208
ago/16	SARIMA(1,1,1)(0,0,2)[24]	126,745	92,346	0,189
set/16	SARIMA(1,1,1)(0,0,2)[24]	128,884	95,397	0,187
out/16	SARIMA(4,1,0)(2,0,0)[24]	144,261	106,244	0,225
nov/16	SARIMA(4,1,0)(2,0,0)[24]	152,601	113,928	0,250
dez/16	SARIMA(4,1,0)(2,0,0)[24]	146,398	107,750	0,250

<sup>37</sup> O teste de Ljung-Box baseia-se na estatística  $Q^* = T(T+2) \sum_{k=1}^h \frac{\hat{\rho}_k^2}{T-k}$ , em que  $T$  é o tamanho da amostra,  $\hat{\rho}_k$  é a autocorrelação da amostra na defasagem  $k$  e  $h$  é o número de defasagens a serem testados

O intervalo de confiança (I.C.) empírico obtido empregando o modelo SARIMA, juntamente com a mediana horária observada e prevista por mês ao longo do ano de 2016, está apresentado no Anexo III, Figura A.III.1.

### 6.2.2 Alisamento exponencial

O método de alisamento exponencial (ETS)<sup>38</sup> é considerado um procedimento automático de previsão, portanto não é necessário aplicar nenhuma estratégia de modelagem como, por exemplo, no método de Box e Jenkins no qual há necessidade de estacionariedade das séries. Os modelos de espaço de estados de suavização exponencial são todos não-estacionários. Outra vantagem dos modelos de suavização exponencial é que estes podem ser não-lineares. Assim, as séries que possuem características de não-linearidade, incluindo a heteroscedasticidade, podem ser melhor modeladas usando modelos de espaço de estados (HYNDMAN e KHANDAKAR, 2008). Ainda, segundo Hyndman e Khandakar (2008), a filosofia dos modelos ETS é de que o mundo é não-estacionário.

Assim como para o modelo SARIMA, o modelo ETS também empregou o pacote *Forecast*. Foi adotada a nomenclatura proposta por Hyndman *et al.* (2002), abordada no tópico (4.6.2). Baseados nestas considerações, Hyndman *et al.* (2002) propuseram o método ETS automático para selecionar os modelos de alisamento exponencial via espaço de estados por meio de critérios de informação. Assim como na função “*auto.arima*”, os critérios de informação empregados foram o AIC, o AICc e o BIC. O modelo automático emprega a função “*ets*” e considera todos os possíveis modelos dentro das 30 possibilidades de alisamento exponencial (combinando cinco formas de tendência por três componentes sazonais por dois erros) que descrevem uma série temporal (HYNDMAN *et al.*, 2002; GARDNER Jr., 2006). Por *default* o critério utilizado para selecionar o melhor modelo dentre as 30 possibilidades é o modelo que apresente o menor critério de Akaike (AIC).

O modelo selecionado automaticamente, empregando a função “*ets*” (Z,Z,Z), foi o ETS (A,Ad,A). Este modelo representa o Holt-Winters Aditivo, com erro aditivo, tendência

---

<sup>38</sup> Onde ETS tem como primeira variável o tipo de erro, a segunda variável o tipo de tendência e a terceira e última variável o tipo de sazonalidade da série em estudo.

amortecida e sazonalidade aditiva. O objetivo da seleção automática é encontrar o modelo que seja mais parcimonioso dentre as várias possibilidades, isto é, identificar o modelo com o menor número de parâmetros possíveis a serem estimados e que consigam explicar bem o comportamento da variável resposta (PELLEGRINI, 2012), refletindo um bom ajuste e consequentemente um bom poder de previsão.

Assim como observado para os modelos SARIMA, os resultados obtidos para os modelo ETS demonstram que os desvios não apresentam distribuição normal, com média zero e variância constante, o que é confirmado pelo teste de normalidade JB (Tabela A.III.4) e pelo gráfico de probabilidade Q-Q. Tanto o diagnóstico do ETS, quanto do modelo híbrido STL-ETS (sub tópico 6.2.3) estão apresentados na Tabela A.III.6.

Os resultados dos modelos são apresentados na Tabela 6.3 e na Tabela A.III.3

**Tabela 6.3** – Modelos de previsão empregando alisamento exponencial (ETS) e suas respectivas métricas (RMSE, MAE e MAPE) para os meses de previsão

Mês	Modelo	RMSE	MAE	MAPE
jan/16	ETS(A,Ad,A)*	213,707	162,322	0,378
fev/16	ETS(A,Ad,A)*	165,098	124,729	0,254
mar/16	ETS(A,Ad,A)*	148,331	110,633	0,225
abr/16	ETS(A,Ad,A)*	149,849	117,553	0,222
mai/16	ETS(A,Ad,A)*	133,367	108,852	0,208
jun/16	ETS(A,Ad,A)*	152,310	120,529	0,232
jul/16	ETS(A,Ad,A)*	131,575	98,838	0,204
ago/16	ETS(A,Ad,A)*	112,965	82,736	0,165
set/16	ETS(A,Ad,A)*	126,392	94,812	0,181
out/16	ETS(A,Ad,A)*	142,584	105,532	0,214
nov/16	ETS(A,Ad,A)*	147,768	113,436	0,242
dez/16	ETS(A,Ad,A)*	146,032	110,034	0,253

\* empregou-se transformação Box-Cox

O intervalo de confiança (I.C.) empírico obtido empregando o modelo ETS, juntamente com a mediana horária prevista e observada por mês ao longo do ano de 2016 está apresentado na Figura A.III.2 do Anexo III. O ETS possui uma grande capacidade de interpolação, o que pode ser observado nesta figura, principalmente no segundo semestre de 2016.

### 6.2.3 Modelo híbrido STL-ETS

O modelo híbrido heterogêneo (STL-ETS) foi calculado empregando o pacote *forecastHybrid* (SHAUB e HELLIS, 2019). A função “*hybridModel*” contida neste pacote combinam as diversas abordagens do pacote *Forecast* (HYNDMAN *et al.*, 2017). As previsões geradas a partir da função “*ets*” e a função “*stlm*” podem ser combinadas com pesos iguais, com pesos baseados em erros dentro da amostra (BATES e GRANGER, 1969), ou com pesos baseados em validação cruzada<sup>39</sup>. O padrão (*default*) é o emprego de pesos iguais, o que mostrou ser robusto e melhor em muitos casos do que dar mais peso aos modelos com melhor desempenho dentro da amostra (SHAUB e HELLIS, 2019).

O modelo híbrido heterogêneo (STL-ETS) construído empregou um peso igual à 50% para cada um dos algoritmos utilizados na sua composição. Os resultados são obtidos otimizando os recursos do modelo com base na minimização de erros e estão descritos na Tabela 6.4.

**Tabela 6.4** – Modelos de previsão empregando o STL-ETS e suas respectivas métricas (RMSE, MAE e MAPE) para os meses de previsão.

Mês	Modelo	RMSE	MAE	MAPE
jan/16	STL-ETS (Peso: 0,5 STL e 0,5 ETS)	216,071	166,601	0,385
fev/16	STL-ETS (Peso: 0,5 STL e 0,5 ETS)	167,871	127,170	0,268
mar/16	STL-ETS (Peso: 0,5 STL e 0,5 ETS)	143,909	111,058	0,222
abr/16	STL-ETS (Peso: 0,5 STL e 0,5 ETS)	147,702	117,951	0,223
mai/16	STL-ETS (Peso: 0,5 STL e 0,5 ETS)	117,948	89,126	0,171
jun/16	STL-ETS (Peso: 0,5 STL e 0,5 ETS)	126,296	95,176	0,192
jul/16	STL-ETS (Peso: 0,5 STL e 0,5 ETS)	121,445	93,172	0,193
ago/16	STL-ETS (Peso: 0,5 STL e 0,5 ETS)	104,619	75,636	0,153
set/16	STL-ETS (Peso: 0,5 STL e 0,5 ETS)	116,838	88,206	0,171
out/16	STL-ETS (Peso: 0,5 STL e 0,5 ETS)	138,741	102,991	0,215
nov/16	STL-ETS (Peso: 0,5 STL e 0,5 ETS)	148,777	110,931	0,247
dez/16	STL-ETS (Peso: 0,5 STL e 0,5 ETS)	141,399	103,414	0,237

Em ambos modelos que empregam alisamento exponencial (ETS e STL-ETS), os diagnósticos dos resíduos, as funções de auto correlação residual, os histogramas dos resíduos e os gráficos de probabilidade Q-Q estão presentes nos Anexo III na Tabela A.III.6. Diferentemente do modelo ETS, o modelo híbrido que emprega o filtro STL com o ETS demonstram que os

<sup>39</sup> Para maiores detalhes ver Shaub e Hellis (2019)

desvios apresentam distribuição normal, com média zero e variância constante, o que é confirmado pelo gráfico de probabilidade Q-Q.

Na Figura A.III.3 serão apresentados o intervalo de confiança (I.C.) empírico obtido empregando o modelo híbrido STL-ETS juntamente com a mediana horária observada e prevista por mês ao longo do ano de 2016.

#### 6.2.4 Redes Neurais não-lineares autorregressivas com *bootstrap* (BNNAR)

Para o modelo BNNAR empregou-se a função “*nnetar*” presente no pacote *Forecast* (HYNDMAN *et al.*, 2017). Esta função ajusta um modelo RNA do tipo *feedforward* com uma única camada oculta a uma série temporal defasada como entrada. Portanto, é um modelo autorregressivo não-linear.

Considerando que esta rede emprega a notação  $NNAR(p, P, k)[m]$  para indicar que a defasagem é de  $p$  observações, com ordens de sazonalidade  $P$  e  $m$  e  $k$  nós (neurônios) na camada oculta. Por *default*, a definição do número de neurônios é dada pelo valor inteiro arredondado da equação  $k = (p + P + 1)/2$ . Um conjunto de 100 redes foi treinado empregando, cada uma, pesos iniciais aleatórios. Posteriormente, a média foi calculada para realizar as previsões. Este algoritmo emprega o critério de Akaike (AIC) do modelo AR ( $p$ ) linear após ajuste sazonal com uma decomposição STL para a definição ótima do número de defasagens e uma transformação Box-Cox com  $\lambda = 0,5$  com o objetivo de garantir que os resíduos sejam aproximadamente homocedásticos.

Após o treino foi realizada a previsão para um horizonte de 24 horas. O emprego do argumento *bootstrap* permite que os erros sejam gerados aleatoriamente a partir dos erros históricos. Já o argumento *npaths* controla o número de simulações realizadas. Na presente análise foram empregadas 100 reamostragens *bootstrap* e 100 simulações.

Os resultados dos modelos BNNAR são apresentados na Tabela 6.5.

**Tabela 6.5** – Parâmetros dos modelos de previsão empregando redes neurais não-lineares autorregressivos com *bootstrap* e suas respectivas métricas (RMSE, MAE e MAPE)

Mês	Modelo	Nº Redes	Peso	Sigma estimado	RMSE	MAE	MAPE
jan/16	BNNAR(30,1,1)[24]	100	33	2950	151,753	112,491	0,285
fev/16	BNNAR(30,1,1)[24]	100	33	3002	135,642	105,629	0,214
mar/16	BNNAR(30,1,1)[24]	100	33	3087	115,860	95,741	0,186
abr/16	BNNAR(30,1,1)[24]	100	33	3010	103,786	78,048	0,138
mai/16	BNNAR(30,1,1)[24]	100	33	3088	101,266	79,276	0,144
jun/16	BNNAR(30,1,1)[24]	100	33	2950	104,291	84,596	0,164
jul/16	BNNAR(30,1,1)[24]	100	33	3023	107,118	90,339	0,181
ago/16	BNNAR(30,1,1)[24]	100	33	3062	97,933	81,933	0,154
set/16	BNNAR(30,1,1)[24]	100	33	3178	104,829	88,543	0,169
out/16	BNNAR(30,1,1)[24]	100	33	3127	108,647	92,146	0,186
nov/16	BNNAR(30,1,1)[24]	100	33	3121	121,898	97,024	0,213
dez/16	BNNAR(30,1,1)[24]	100	33	3079	113,015	82,346	0,202

O intervalo de confiança (I.C.) empírico obtido empregando o modelo híbrido BNNAR, juntamente com a mediana horária observada e prevista por mês ao longo do ano de 2016, está apresentado no Anexo III, Figura A.III.4. Na prática, a integralidade dos modelos avaliados, o I.C. é maior que os 90%. Para este modelo, que apresentou 16 pontos fora do I.C. (no horário de ponta para os meses de maio, junho, agosto e setembro) dentro dos 366 dias do ano de 2016, a confiança ainda é superior a 95%.

### 6.2.5 *Extreme learning machine* (ELM)

Para o modelo ELM utilizou-se a função “*elm*” presente no pacote *nnfor* (KOURENTZES, 2017). Este algoritmo automático<sup>40</sup>, que baseia-se nos trabalhos de Crone e Kourentzes (2010) e Kourentzes *et al.* (2014), é puramente orientado a dados, combinando, assim, as vantagens dos filtros e dos invólucros.

Este método ajusta uma SLFN à série de demanda horária defasada como entrada da rede, onde estas variáveis defasadas foram definidas automaticamente pelo algoritmo. A ELM possui uma única saída e foi treinada empregando 100 nós ocultos em 20 repetições (número de redes

<sup>40</sup> Para maiores detalhes ver <https://github.com/trnnick/nnfor>

treinadas onde a previsão será um *ensemble*). Os pares seno-cosseno foram aplicados como *dummies* determinísticas sazonais. A previsão combinada empregou o operador mediano e a estimativa de peso de saída utilizada foi a regularização Lasso<sup>41</sup> (TIBSHIRANI, 1996).

Os resultados dos modelos ELM são apresentados na Tabela 6.6.

**Tabela 6.6** – Modelos de previsão ELM, o número de *lags* empregados e suas respectivas métricas (RMSE, MAE e MAPE)

Mês	Modelo	Número de Lags	RMSE	MAE	MAPE
jan/16	ELM	60	145,077	111,832	0,281
fev/16	ELM	243	134,271	107,674	0,210
mar/16	ELM	127	123,607	101,783	0,199
abr/16	ELM	148	116,152	90,609	0,159
mai/16	ELM	112	95,900	80,430	0,152
jun/16	ELM	109	98,230	79,438	0,157
jul/16	ELM	125	101,832	84,061	0,165
ago/16	ELM	166	90,620	72,558	0,137
set/16	ELM	98	95,081	74,738	0,146
out/16	ELM	154	107,985	88,033	0,183
nov/16	ELM	122	115,048	91,792	0,203
dez/16	ELM	140	103,291	82,606	0,196

O mês de janeiro (verão) foi o mês em que ocorreu o pior ajuste. Já o mês de agosto foi o que obteve os melhores ajustes dadas as métricas de avaliação da capacidade preditiva com um RMSE de 90,620, um MAE de 72,558 e um MAPE de 13,7%.

A Figura A.III.5 apresenta o intervalo de confiança (I.C.) empírico obtido empregando o modelo híbrido ELM, juntamente com a mediana horária observada e prevista por mês ao longo do ano de 2016.

<sup>41</sup> Para maiores detalhes ver TIBSHIRANI, R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Serie B*, v. 58, n. 1, p. 267-288, 1996.

### 6.2.6 *Trigonometric Box-Cox transformation, ARMA errors, Trend and Seasonal components (TBATS)*

O TBATS, proposto por De Livera *et al.* (2011), é um método de previsão e decomposição de séries temporais para lidar com dados que apresentam padrões sazonais complexos. Método este que é semelhante ao de alisamento exponencial no contexto de espaço de estados (HYNDMAN *et al.*, 2008), apresentado no tópico 4.6.2, mas que incorpora uma transformação do tipo Box-Cox para lidar com a não-linearidade e a heterocedasticidade, um filtro do tipo ARMA para os resíduos e funções trigonométricas como séries de Fourier para tratar os padrões de sazonalidade.

O primeiro passo é realizar a filtragem da série de dados originais através do algoritmo TBATS, onde são separadas a estrutura principal da série e de seu resíduo. Como parâmetros para a decomposição foram escolhidos três ciclos sazonais de comprimento 24, 168 (24x7) e 8766 (24x365,25), que representam respectivamente o ciclo diário, semanal e a sazonalidade anual.

As séries dos resíduos dos modelos TBATS, as funções de auto correlação residual, os histogramas dos resíduos e as decomposições das séries empregando o TBATS estão presentes na Tabela A.III.7 constantes no Anexo III. Observa-se, na Tabela A.III.7 (a), que este modelo satisfaz as necessidades de sazonalidade usando dados de alta frequência. No entanto, existem comportamentos nos dados que não podem ser explicadas simplesmente sem referir-se a fatores externos. Isso mostra que outras variáveis explicativas devem ser incluídas no modelo visando capturar as informações nos dados com mais precisão.

A Tabela A.III.7 (c) apresenta a aplicação da decomposição TBATS. O primeiro gráfico apresenta os dados reais, o segundo apresenta o nível que é semelhante à tendência, o terceiro gráfico apresenta a inclinação, o quarto, quinto e sexto gráficos apresentam a sazonalidade diária, a sazonalidade semanal e a sazonalidade anual, respectivamente. A decomposição e os gráficos são gerados usando o *software* R, pacote de previsão *Forecast*, com a função “*tbats.components*”.

Após empregar o algoritmo TBATS, presente no pacote *Forecast*, obtiveram-se os modelos TBATS constantes da Tabela 6.7, além das suas respectivas métricas para os meses de previsão.

A ordem do modelo TBATS pode ser então representada pela notação TBATS  $(\emptyset, \{p, q\}, \omega, \{m_1, k_1\}, \{m_2, k_2\}, \dots, \{m_T, k_T\})$ . Os parâmetros obtidos foram

automaticamente ajustados pelo método de máxima verossimilhança. Ao confiar na estimativa de máxima verossimilhança, evitam-se as opções de inicialização “*ad hoc*” com propriedades estatísticas desconhecidas comumente usadas com suavização exponencial (De LIVERA *et al.*, 2011). O modelo estimado para janeiro de 2016 foi designado como TBATS (0,736, {2,2}, 1, {<24,11>, <168,4>, <8766,3>}), ou seja, tal modelo não empregou uma transformação Box-Cox ( $w = 1$ ), aplicou um filtro ARMA (2,2), utilizou três ciclos sazonais de comprimento 24, 168 e 8766 com 11, 4 e 3 termos de Fourier, respectivamente, para cada uma das sazonalidades e um amortecimento de tendência de  $\emptyset = 0,736$ . Valores de  $\emptyset \cong 1$  indicam que o amortecimento é lento e os harmônicos no modelo TBATS reduzem significativamente o número de valores de sementes a serem estimados.

**Tabela 6.7** – Modelos de previsão empregando o TBATS e suas respectivas métricas (RMSE, MAE e MAPE)

Mês	Modelo	RMSE	MAE	MAPE
jan/16	TBATS(0,736,{2,2},1,{<24,11>, <168,4>,<8766,3>})	174,200	129,920	0,298
fev/16	TBATS(0,692,{3,3},-,{<24,11>, <168,6>,<8766,6>})	131,353	102,755	0,212
mar/16	TBATS(0,657,{3,3},1,{<24,10>, <168,6>,<8766,3>})	118,280	94,931	0,190
abr/16	TBATS(0,681,{2,2},1,{<24,9>, <168,6>,<8766,1>})	132,075	105,810	0,205
mai/16	TBATS(0,709,{4,3},1,{<24,9>, <168,6>,<8766,1>})	109,265	86,394	0,170
jun/16	TBATS(0,664,{3,3},1,{<24,8>, <168,6>,<8766,1>})	121,736	92,953	0,193
jul/16	TBATS(0,650,{4,3},1,{<24,5>, <168,6>,<8766,6>})	115,083	87,799	0,186
ago/16	TBATS(0,746,{3,4},1,{<24,4>, <168,6>,<8766,1>})	106,305	79,481	0,164
set/16	TBATS(0,679,{5,4},1,{<24,4>, <168,4>,<8766,5>})	121,547	91,489	0,179
out/16	TBATS(0,695,{2,1},1,{<24,6>, <168,6>,<8766,5>})	123,633	95,830	0,196
nov/16	TBATS(0,673,{2,2},1,{<24,5>, <168,5>,<8766,5>})	132,453	98,412	0,210
dez/16	TBATS(0,661,{3,3},0,878,{<24,6>, <168,6>,<8766,7>})	123,433	92,259	0,208

O mês de agosto foi o que obteve os melhores ajustes dadas as métricas de avaliação da capacidade preditiva com um RMSE de 106,305, um MAE de 79,481 e um MAPE de 16,4%.

O intervalo de confiança (I.C.) empírico obtido empregando o modelo TBATS, juntamente com a mediana horária observada e prevista por mês ao longo do ano de 2016, está apresentado na Figura A.III.6.

### 6.2.7 *Dynamic Time Scan Forecasting (DTSF)*

Para o modelo *Dynamic Time Scan Forecasting* (DTSF) utilizou-se a função “*dtsf*” presente no pacote DTScanF (COSTA e MINETI, 2019). Os argumentos empregados na modelagem: “*window*” especifica o comprimento da janela, “*poli*” define o grau do modelo a ser usado, “*best*” define a quantidade de *matches* a serem empregados, “*forecast*” indica o número de pontos a serem previstos e “*reg*” indica se a regularização da série deve ser realizada.

Com o objetivo de identificar os padrões na série temporal de demanda de água que estão fortemente correlacionados com a janela de varredura das últimas 24 horas assumiu-se uma função de similaridade como um modelo linear, isto é, empregou-se um modelo de grau um. Os 24 melhores padrões da série temporal com estatísticas de alta similaridade ( $R^2$ ) foram, posteriormente, empregados na previsão das próximas 24 horas de demanda.

Os modelos de previsão empregando o DTSF e suas respectivas métricas (RMSE, MAE e MAPE) para os meses de previsão encontram-se na Tabela 6.8, em que (W,P,B,K) correspondem ao tamanho da janela de varredura, o grau do polinômio, o número de *matches* a serem empregados e o tamanho do horizonte de previsão, respectivamente.

**Tabela 6.8** – Modelos de previsão empregando o DTSF e suas respectivas métricas (RMSE, MAE e MAPE) para os meses de previsão.

Mês	Modelo	RMSE	MAE	MAPE
jan/16	DTSF (W,P,B,K) [24,1,24,24]	182,011	139,618	0,323
fev/16	DTSF (W,P,B,K) [24,1,24,24]	147,250	139,618	0,229
mar/16	DTSF (W,P,B,K) [24,1,24,24]	122,836	90,394	0,177
abr/16	DTSF (W,P,B,K) [24,1,24,24]	123,432	95,721	0,180
mai/16	DTSF (W,P,B,K) [24,1,24,24]	119,758	88,201	0,168
jun/16	DTSF (W,P,B,K) [24,1,24,24]	117,162	85,972	0,170
jul/16	DTSF (W,P,B,K) [24,1,24,24]	128,379	97,254	0,198
ago/16	DTSF (W,P,B,K) [24,1,24,24]	109,401	77,338	0,150
set/16	DTSF (W,P,B,K) [24,1,24,24]	118,224	90,596	0,172
out/16	DTSF (W,P,B,K) [24,1,24,24]	123,264	93,284	0,195
nov/16	DTSF (W,P,B,K) [24,1,24,24]	141,991	108,943	0,232
dez/16	DTSF (W,P,B,K) [24,1,24,24]	133,700	96,193	0,224

Assim como no TBATS, no ELM, no BNNAR, no ETS e no STL-ETS o mês de agosto foi o mês que o DTSF obteve os melhores ajustes dadas as métricas de avaliação da capacidade preditiva com um RMSE de 109,401, um MAE de 77,338 e um MAPE de 15%.

O intervalo de confiança (I.C.) empírico obtido empregando o modelo DTSF, juntamente com a mediana horária observada e prevista por mês ao longo do ano de 2016 são apresentados na Figura A.III.7.

### 6.2.8 Análise comparativa dos modelos empregando as diversas técnicas

Os modelos avaliados para descrever o processo gerador e o modelo proposto estão apresentados na Tabela 6.9. Para avaliar estes modelos foram empregadas as métricas raiz do erro quadrático médio (RMSE), erro absoluto médio (MAE), erro percentual médio (MAPE) além do CC, gerado para a previsão das próximas 24 horas ao longo do mês.

Segundo estudo realizado por Makridakis *et al.*, (2018), inúmeros métodos estatísticos e de *machine learning* foram utilizados para prever um passo à frente. Estes autores empregaram o MASE e a métrica erro percentual absoluto médio simétrico (sMAPE)<sup>42</sup> para avaliar a acurácia destes métodos. Ainda, segundo estes autores, o ETS foi o que obteve melhor acurácia entre os métodos avaliados.

Em uma análise comparativa entre o ETS e o modelo híbrido STL-ETS, para os meses de janeiro e fevereiro, todas métricas do modelo ETS se mostraram melhores que as do modelo híbrido STL-ETS. Já para o mês de abril somente RMSE do modelo híbrido (147,702) se mostrou melhor que o ETS (149,849). Contudo, para a métrica MAE e MAPE os resultados do ETS se mostraram ligeiramente melhores que as do modelo STL-ETS. Para o mês de março e para o período compreendido entre os meses de maio a setembro, o modelo com filtro obteve melhores resultados em todas as métricas avaliadas. Em outubro somente o MAPE do modelo ETS obteve um resultado ligeiramente melhor que a do modelo híbrido em 0,01 pontos percentuais. Para os meses novembro e dezembro o modelo híbrido STL-ETS também obtiveram resultados melhores para as métrica avaliadas RMSE, MAE e MAPE. Contudo, o

---

<sup>42</sup> Segundo Makidrakis *et al.*, (2018) esta métrica foi originalmente empregada na competição M3 para avaliar os métodos participantes.

filtro STL empregado no modelo híbrido, implicou em um gráfico de probabilidade Q-Q que se aproxima da normal.

Quando comparamos o modelo proposto DTFSF *versus* o ETS, observa-se que o DTFSF se mostrou superior ao alisamento exponencial em todas as métricas avaliadas (RMSE, MAE e MAPE) para todos meses do ano de 2016. O mesmo resultado foi obtido quando comparado com o modelo SARIMA. Quando comparado com o modelo híbrido STL-ETS, este se mostrou superior à métrica RMSE para os meses de maio, julho, agosto e setembro e para a métrica MAPE se mostrou ligeiramente superior somente nos meses de julho e setembro, com percentuais de 19,3% e 17,1% frente aos 19,8% e 17,2% obtidos pelo DTFSF. Quando realizada a comparação do DTFSF frente ao TBATS para a métrica RMSE, o TBATS foi superior para os meses de janeiro, fevereiro, março, maio, julho, agosto, novembro e dezembro. Entretanto, quando comparado frente a métrica MAPE o DTFSF se mostrou mais preciso no período de março a junho e no trimestre compreendido pelos meses de agosto a outubro.

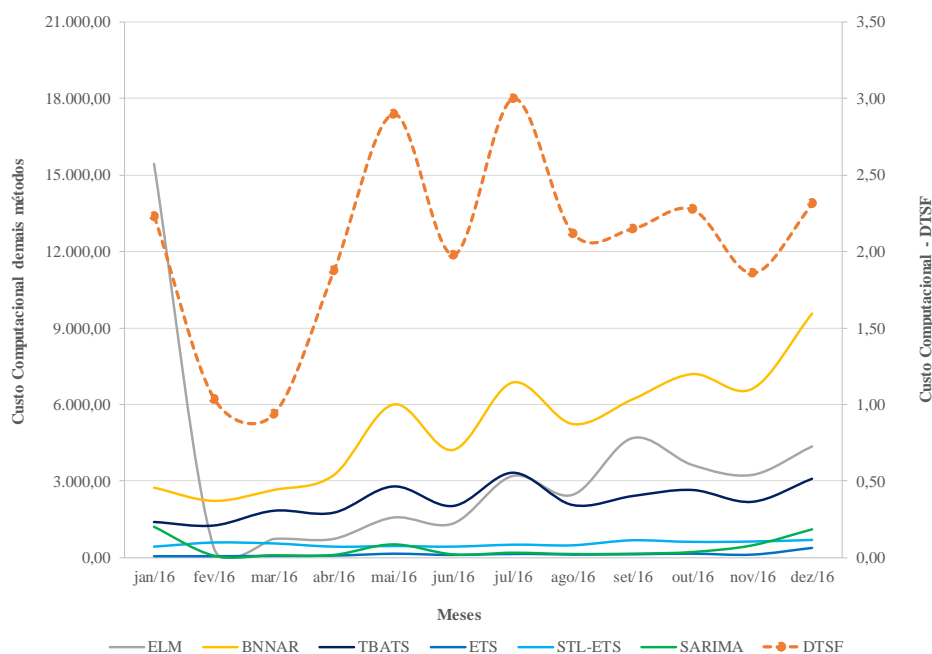
**Tabela 6.9** – Avaliação dos métodos de previsão empregando as métricas raiz do erro quadrático médio (RMSE), erro absoluto médio (MAE), erro percentual médio (MAPE), complexidade computacional (CC) para um horizonte de previsão horário

Mês de previsão	DTSF				ELM				BNNAR				TBATS			
	RMSE	MAE	MAPE	CC	RMSE	MAE	MAPE	CC	RMSE	MAE	MAPE	CC	RMSE	MAE	MAPE	CC
jan/16	182,011	139,618	0,323	<b>2,230</b>	<b>145,077</b>	<b>111,832</b>	<b>0,281</b>	15.433,79	151,753	112,491	0,285	2.734,73	174,200	129,920	0,298	1.395,87
fev/16	147,250	112,660	0,229	<b>1,036</b>	134,271	107,674	<b>0,210</b>	390,77	135,642	105,629	0,214	2.215,38	<b>131,353</b>	<b>102,755</b>	0,212	1.247,59
mar/16	122,836	<b>90,394</b>	<b>0,177</b>	<b>0,936</b>	123,607	101,783	0,199	735,93	<b>115,860</b>	95,741	0,186	2.641,90	118,280	94,931	0,190	1.837,43
abr/16	123,432	95,721	0,180	<b>1,873</b>	116,152	90,609	0,159	742,17	<b>103,786</b>	<b>78,048</b>	<b>0,138</b>	3.220,66	132,075	105,810	0,205	1.751,24
mai/16	119,758	88,201	0,168	<b>2,900</b>	<b>95,900</b>	80,430	0,152	1.582,85	101,266	<b>79,276</b>	<b>0,144</b>	6.005,79	109,265	86,394	0,170	2.797,26
jun/16	117,162	85,972	0,170	<b>1,978</b>	<b>98,230</b>	<b>79,438</b>	<b>0,157</b>	1.343,13	104,291	84,596	0,164	4.219,84	121,736	92,953	0,193	2.021,75
jul/16	128,379	97,254	0,198	<b>2,999</b>	<b>101,832</b>	<b>84,061</b>	<b>0,165</b>	3.202,72	107,118	90,339	0,181	6.865,31	115,083	87,799	0,186	3.333,40
ago/16	109,401	77,338	0,150	<b>2,118</b>	<b>90,620</b>	<b>72,558</b>	<b>0,137</b>	2.460,95	97,933	81,933	0,154	5.238,27	106,305	79,481	0,164	2.058,82
set/16	118,224	90,596	0,172	<b>2,148</b>	<b>95,081</b>	<b>74,738</b>	<b>0,146</b>	4.689,41	104,829	88,543	0,169	6.203,08	121,547	91,489	0,179	2.412,09
out/16	123,264	93,284	0,195	<b>2,276</b>	<b>107,985</b>	<b>88,033</b>	<b>0,183</b>	3.632,87	108,647	92,146	0,186	7.199,26	123,633	95,830	0,196	2.652,74
nov/16	141,991	108,943	0,232	<b>1,858</b>	<b>115,048</b>	<b>91,792</b>	<b>0,203</b>	3.249,07	121,898	97,024	0,213	6.612,75	132,453	98,412	0,210	2.185,04
dez/16	133,700	96,193	0,224	<b>2,311</b>	<b>103,291</b>	<b>82,606</b>	<b>0,196</b>	4.362,90	113,015	82,346	0,202	9.574,77	123,433	92,259	0,208	3.094,29

Mês de previsão	ETS				STL_ETS				SARIMA				NB			
	RMSE	MAE	MAPE	CC	RMSE	MAE	MAPE	CC	RMSE	MAE	MAPE	CC	RMSE	MAE	MAPE	CC
jan/16	213,707	162,322	0,378	52,82	216,071	166,601	0,385	419,01	214,723	162,299	0,381	1.215,40	209,745	156,491	0,379	1,00
fev/16	165,098	124,729	0,254	51,73	167,871	127,170	0,268	577,08	173,828	130,608	0,266	75,52	183,982	145,982	0,301	1,00
mar/16	148,331	110,633	0,225	59,73	143,909	111,058	0,222	542,01	155,864	115,328	0,231	83,81	166,761	121,163	0,233	1,00
abr/16	149,849	117,553	0,222	70,58	147,702	117,951	0,223	415,32	149,156	112,378	0,217	97,01	141,742	102,277	0,185	1,00
mai/16	133,367	108,852	0,208	149,17	117,948	89,126	0,171	439,34	131,096	98,502	0,193	511,58	143,721	104,316	0,191	1,00
jun/16	152,310	120,529	0,232	102,69	126,296	95,176	0,192	420,30	134,915	98,710	0,202	117,17	149,416	107,567	0,208	1,00
jul/16	131,575	98,838	0,204	143,76	121,445	93,172	0,193	492,91	135,242	98,178	0,208	182,39	146,528	106,845	0,216	1,00
ago/16	112,965	82,736	0,165	117,99	104,619	75,636	0,153	473,56	126,745	92,346	0,189	128,82	123,719	81,809	0,154	1,00
set/16	126,392	94,812	0,181	127,45	116,838	88,206	0,171	668,20	128,884	95,397	0,187	144,64	133,974	93,607	0,180	1,00
out/16	142,584	105,532	0,214	149,55	138,741	102,991	0,215	604,27	144,261	106,244	0,225	213,13	148,850	113,865	0,234	1,00
nov/16	147,768	113,436	0,242	112,07	148,777	110,931	0,247	624,49	152,601	113,928	0,250	477,35	171,804	139,047	0,290	1,00
dez/16	146,032	110,034	0,253	383,01	141,399	103,414	0,237	677,74	146,398	107,750	0,250	1.103,02	152,384	115,530	0,267	1,00

Em seguida apresenta-se uma análise comparativa do DTSF *versus* os modelos neuronais ELM e BNNAR. O invólucro que empregou o ELM se mostrou mais acurado não só frente ao DTSF, mas também ao BNNAR para todo o segundo semestre de 2016 para as métricas RMSE, MAE e MAPE. O BNNAR obteve resultados superiores frente ao ELM e o DTSF para os meses de abril e maio para a métrica MAPE. No tocante a esta métrica o ELM foi superior ao DTSF e ao BNNAR para o primeiro trimestre do ano. Entretanto, os métodos neuronais possuem uma complexidade computacional muito superior ao DTSF. O invólucro ELM possui vantagens (melhores resultados), mas adiciona uma camada adicional de complexidade na tarefa de modelagem aumentando o custo computacional. Somente para efeito comparativo, usando o CC de dezembro de 2016, o BNNAR e o ELM tiveram um custo computacional em relação ao DTSF de 4.143 e 1.887 vezes, respectivamente.

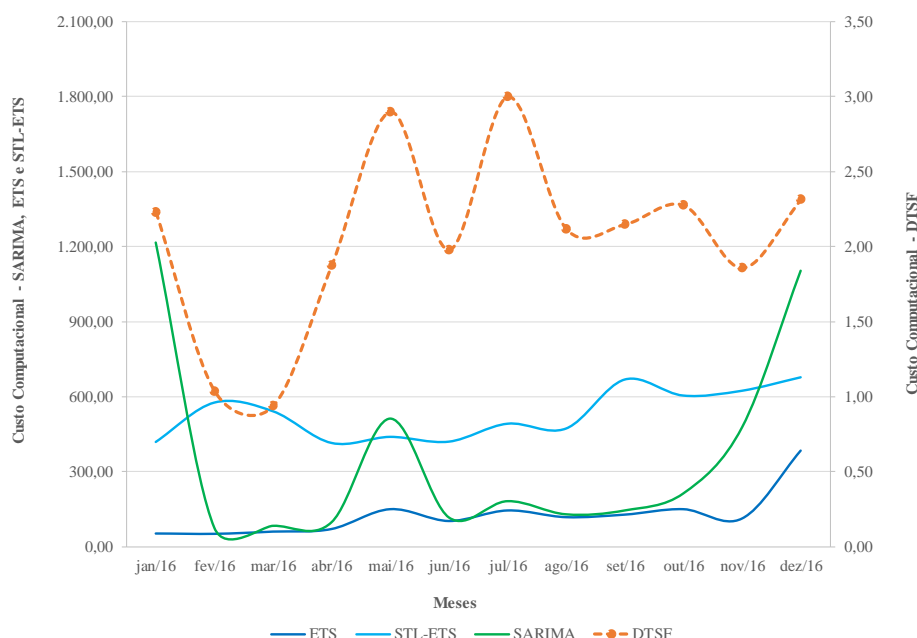
Em comparação com as diversas abordagens estatísticas e de SC, o método DTSF se mostrou extremamente rápido. Os resultados obtidos mostram que este método fornece valores de previsão semelhantes ou melhorados quando comparados com métodos como ELM, BNNAR, SARIMA, ETS, TBATS e STL-ETS, todavia, usando uma fração do tempo.



**Figura 6.6** – Custo computacional do DTSF *versus* demais métodos empregados ao longo de 2016

A Figura 6.6 ilustra o comparativo do custo computacional do DTSF *versus* os outros métodos empregados no presente estudo. O DTSF, ao longo de 2016, teve o seu custo variando entre 0,936 e 2,999 (eixo secundário do gráfico). Contudo, o segundo método com menor complexidade computacional foi o ETS. Os valores deste variaram entre 51,73 a 383. Já os métodos neuronais ELM e BNNAR e o método estatístico TBATS tiveram um crescimento acentuado à medida que houveram incorporações de novos dados para a previsão, recalibração de origem rolante, procedimento este o preferível segundo Tashman (2000). Os CC dos respectivos métodos variaram de 390 a 15.433, 2.215 a 9.574 e 1.247 a 3.094. Os menores valores se concentraram no mês de fevereiro devido ao menor número de dias de previsão, com exceção do DTSF que ocorreu em março e do STL-ETS que ocorreu em abril. Pode-se observar a semelhança do comportamento do CC ao longo de 2016 para os métodos DTSF, ELM, BNNAR e TBATS, métodos estes que obtiveram as melhores capacidades de generalização.

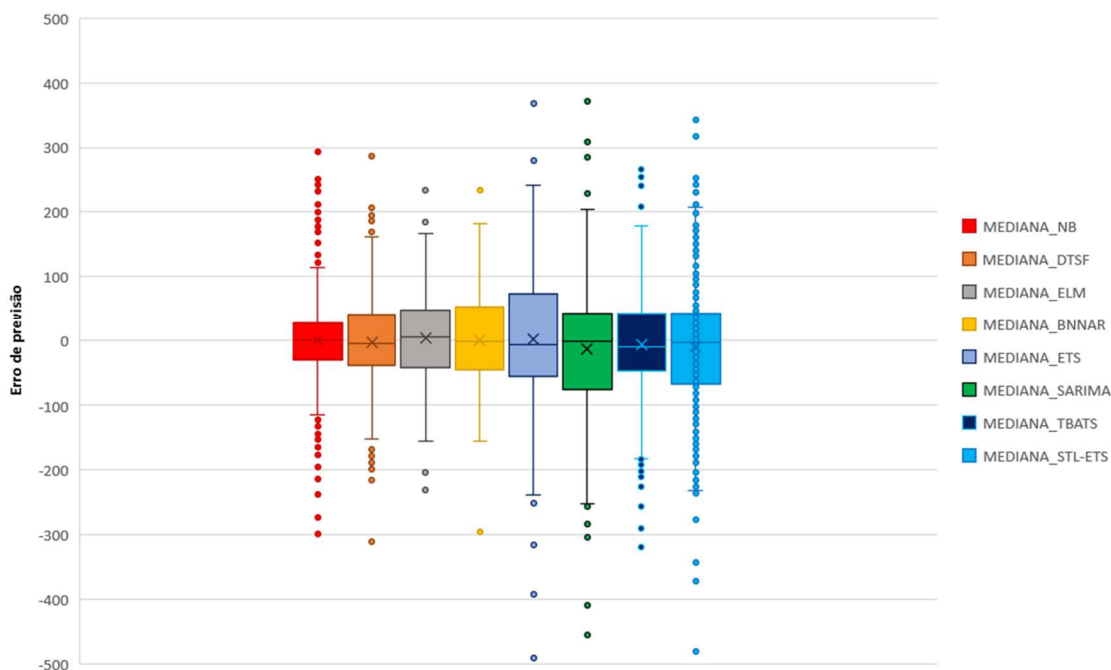
Em virtude da diferença de escala do CC dos métodos neuronais e do TBATS (eixo principal) da Figura 6.6, não ficou claro como o CC dos métodos SARIMA, ETS e o modelo híbrido STL-ETS se comportaram frente ao modelo proposto DTSF. A Figura 6.7 ilustra o CC do DTSF *versus* estes métodos.



**Figura 6.7** – Custo computacional do DTSF *versus* os métodos SARIMA, ETS e STL-ETS empregados ao longo de 2016

Os resultados das previsões um passo à frente (horizonte de 24 horas) realizadas para todos os dias do ano de 2016 geraram matrizes de erros com dimensão  $(i \times j)$ , onde  $i$  são os dias de previsão e  $j$  os horizontes de previsão em horas. Os erros foram obtidos pela diferença entre os valores fora da amostra e os valores previstos. As tabelas A.III.8 e A.III.9 apresentam o *bloxplot* dos erros quadráticos médios da matriz de erros das previsões (24 horas para os todos os dias do mês) para os modelos *Naive Bayes*, DTSF, ELM, BNNAR, TBATS, SARIMA, ETS e o modelo híbrido STL-ETS.

A Figura 6.8 apresenta a mediana dos erros de previsão para os modelos avaliados. Observa-se que os valores do segundo quartil estão bem próximos a zero. O modelo que apresentou a menor dispersão dos erros foi o NB, seguido pelo BNNAR e DTSF. As maiores amplitudes interquartis são observadas nos modelos ETS e SARIMA, o mesmo acontecendo com os comprimentos das caudas de distribuição.

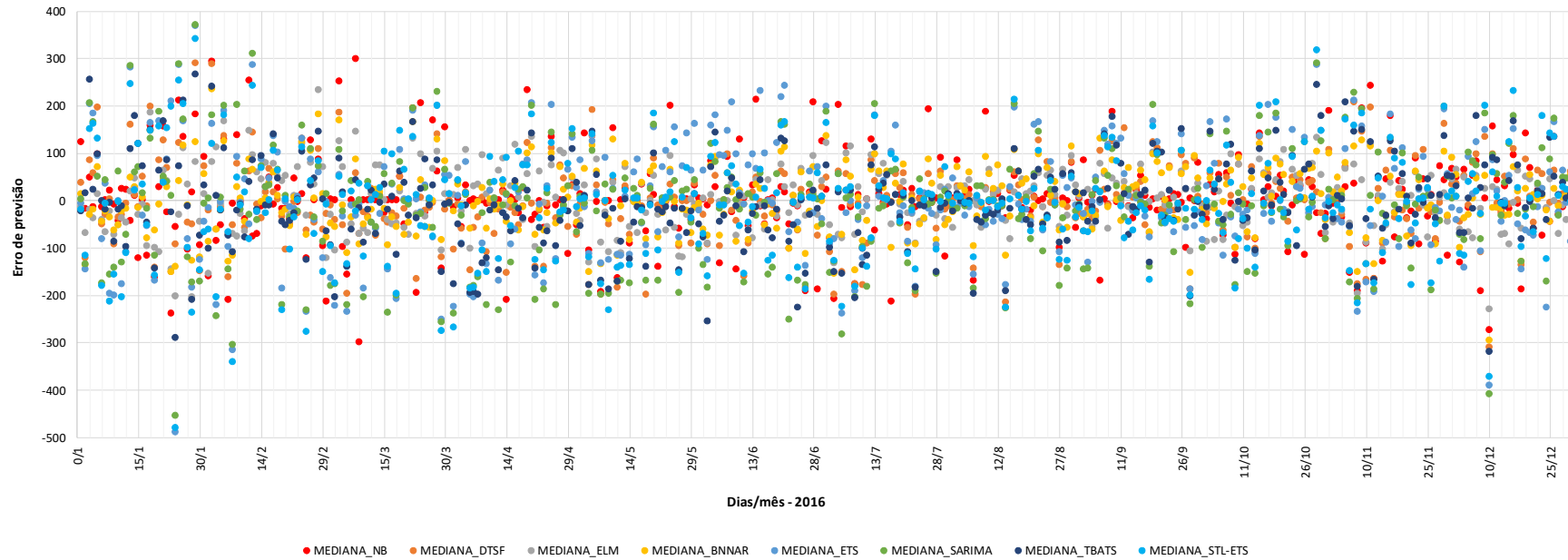


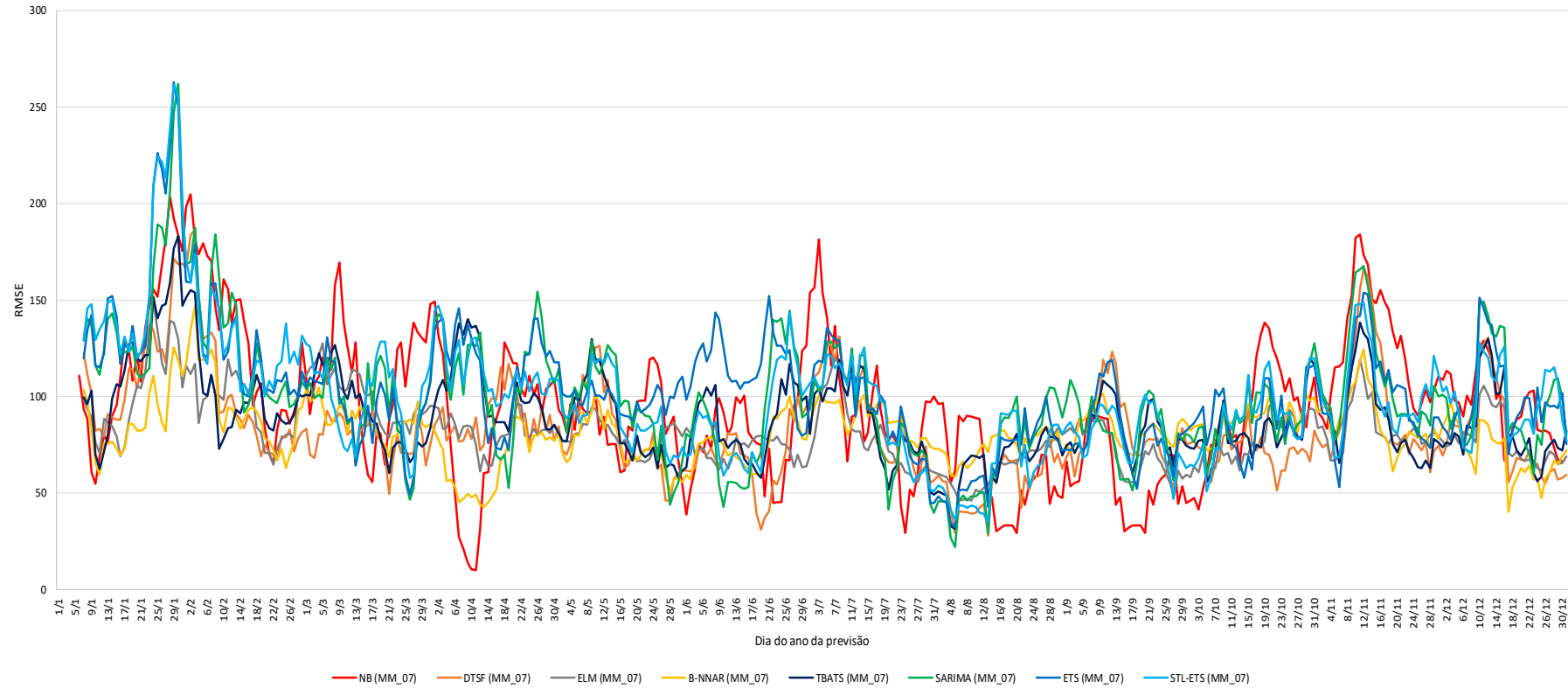
**Figura 6.8** – *Bloxplot* da mediana dos erros de previsão para as próximas 24 horas, dos modelos avaliados (01 de janeiro a 31 de dezembro de 2016)

A Figura 6.9 demonstra uma grande dispersão da mediana dos erros de previsão para todos os modelos testados ao longo do ano de 2016. Observa-se uma maior dispersão nos meses de verão e outono. Contudo, a dispersão observada não possibilita, de forma clara,

afirmar qual modelo é melhor ao longo dos dias. Objetivando ter uma visão mais clara da acurácia dos modelos estudados aplicou-se uma média móvel de 7 dias para a mediana dos MSE. O resultado desta suavização está apresentado na Figura 6.10, possibilitando uma compreensão mais clara de como os modelos se comportam ao longo do ano. Este resultado indica que, em um horizonte semanal, o erro de previsão horária para os diferentes modelos varia bastante ao longo do ano. Portanto, conclui-se que não há um melhor modelo para prever a demanda de água todos os dias do ano, assim como apresentado em Groppo *et al.* (2019). Esta suavização usando médias móveis sugere uma possível estratégia para chaveamento de modelos. Pode-se utilizar a tendência histórica do erro suavizado de um modelo para definir um período na qual a sua previsão seja, em média, preterível.

**Figura 6.9** – Evolução da mediana dos erros de previsão, para as próximas 24 horas, de todos modelos avaliados





**Figura 6.10** – Evolução da mediana dos RMSE, para as próximas 24 horas, após aplicar uma média móvel de 7 dias para todos os modelos avaliados

Após aplicar a média móvel de 7 dias para a mediana dos erros de previsão obtiveram-se os seguintes resultados apresentados na Tabela 6.10.

**Tabela 6.10** - Número e percentual dos melhores ajustes de previsão de cada um dos modelos para as próximas 24 horas ao longo do ano de 2016.

<b>Modelo</b>	<b>Número das melhores previsões</b>	<b>Percentual das melhores previsões</b>
NB	77	21,39%
DTSF	69	19,17%
ELM	63	17,50%
BNNAR	81	22,50%
ETS	10	2,78%
STL-ETS	13	3,61%
TBATS	20	5,56%
SARIMA	27	7,50%

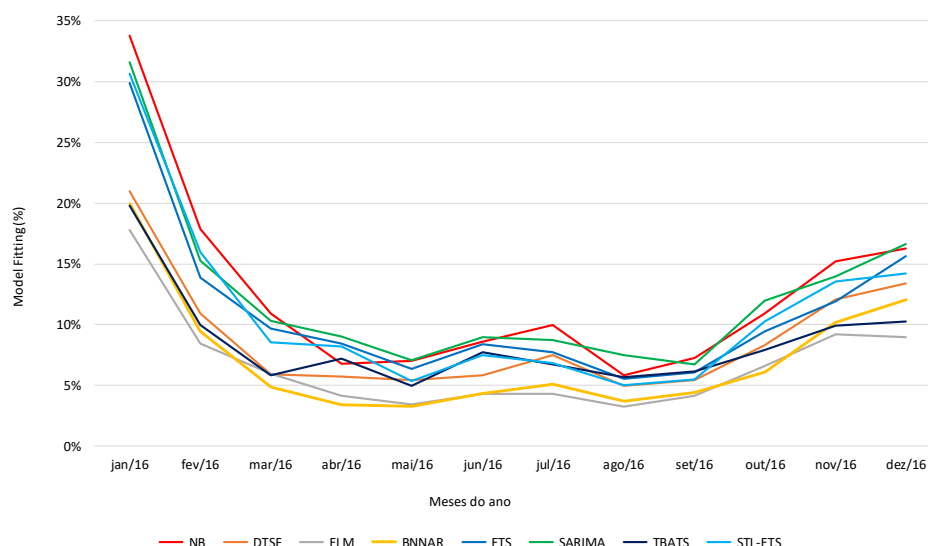
Os resultados indicam que em um horizonte de uma semana com previsão horária, o erro de previsão para os diferentes modelos varia bastante. Não há, portanto, um melhor modelo para prever a demanda de água todos os dias do ano. O melhor modelo na previsão de demanda de água ao longo do ano de 2016 foi o BNNAR com 22,5% das melhores previsões enquanto os outros modelos representam somados 77,5%. Seguido pelo método *Naive Bayes* com 21,39%, pelo DTSF com 19,17% e pelo ELM com 17,50%. Os outros modelos estatísticos e o modelo híbrido empregando o filtro STL apresentaram as piores performances após a aplicação de uma média móvel semanal. Saber que um determinado método sofisticado não é tão preciso quanto um método muito mais simples é perturbador do ponto científico, já que o método mais sofisticado requer uma grande dose de conhecimento acadêmico e um grande CC a ser aplicado (MAKIDRAKIS *et al.*, 2018).

Empregando a métrica MF os dois modelos que apresentaram os melhores ajustes ao longo dos meses analisados foi o ELM e o BNNAR. Estes resultados estão apresentados na Tabela 6.11 e na Figura 6.11. Na Figura 6.11 demonstra-se a acurácia dos modelos ao longo dos meses. Observa-se que os piores ajustes se deram nos meses de verão (janeiro, fevereiro e dezembro)

e primavera (setembro a novembro), enquanto os melhores ajustes ocorreram nos meses do outono e inverno, meses de menor demanda de água. Os resultados dos dois melhores modelos (ELM e BNNAR) possuem um comportamento muito semelhante ao longo dos meses. Já os modelos DTSF e TBATS alternam a performance ao longo do período analisado como se pode observar na Figura 6.11.

**Tabela 6.11** – Performance da previsão empregando o *Model Fitting* (MF) de 01 de janeiro a 31 de dezembro de 2016

Mês de previsão	NB	DTSF	ELM	BNNAR	ETS	STL-ETS	TBATS	SARIMA
jan/16	0,3382	0,2102	<b>0,1782</b>	0,1996	0,2989	0,3067	0,1979	0,3162
fev/16	0,1786	0,1093	<b>0,0842</b>	0,0947	0,1387	0,1599	0,0999	0,1526
mar/16	0,1092	0,0589	0,0595	<b>0,0486</b>	0,0968	0,0857	0,0584	0,1035
abr/16	0,0678	0,0572	0,0411	<b>0,0342</b>	0,0841	0,0818	0,0720	0,0905
mai/16	0,0703	0,0542	0,0341	<b>0,0329</b>	0,0638	0,0539	0,0496	0,0705
jun/16	0,0861	0,0584	<b>0,0429</b>	0,0433	0,0839	0,0746	0,0774	0,0894
jul/16	0,0996	0,0749	<b>0,0430</b>	0,0511	0,0772	0,0687	0,0673	0,0875
ago/16	0,0584	0,0498	<b>0,0322</b>	0,0368	0,0552	0,0503	0,0565	0,0749
set/16	0,0723	0,0542	<b>0,0414</b>	0,0438	0,0605	0,0547	0,0616	0,0672
out/16	0,1093	0,0833	0,0661	<b>0,0612</b>	0,0942	0,1026	0,0788	0,1196
nov/16	0,1519	0,1208	<b>0,0923</b>	0,1017	0,1189	0,1356	0,0993	0,1398
dez/16	0,1626	0,1341	<b>0,0895</b>	0,1207	0,1565	0,1423	0,1025	0,1663



**Figura 6.11** – Acurácia dos modelos em prever ao longo dos meses empregando o *Model Fitting* (MF) no período de 01 de janeiro de 2016 a 31 de dezembro de 2016.

## 7 CONCLUSÕES

No trabalho avaliou o desempenho de diferentes algoritmos univariados empregados tendo em vista a sua capacidade de generalização e realização de previsão de curtíssimo prazo (horário) para demanda de água urbana, além de levar em conta a complexidade computacional. Algoritmos que apresentam altíssimo custo computacional muitas vezes não se aplicam no mundo real em uma situação de larga escala. Neste estudo os métodos empregados foram: os modelos neuronais BNNAR e ELM, os modelos estatísticos SARIMA, TBATS, ETS, o modelo híbrido STL-ETS, além do método proposto DTSF.

Existem inúmeros métodos de previsão de demanda, porém, nenhum dos métodos apresentados na revisão de literatura se mostrou eficaz e preciso para um amplo universo de zonas de abastecimento estudadas. Não existe um melhor modelo para todas as situações, devendo ser avaliado cada caso separadamente (GHALEHKHONDABI *et al.*, 2017; GROPPPO *et al.*, 2019). Resultado semelhante foi obtido nesse trabalho.

Na seção anterior foram apresentados os resultados das previsões sob diferentes metodologias de estimação da demanda de água urbana no curto prazo para uma dada região metropolitana do Sudeste brasileiro. Como critérios de avaliação foram empregadas as seguintes métricas: RMSE, MAE, MAPE, CC e MF. Os resultados obtidos demonstram que tanto no verão quanto no outono não houve um modelo que superasse os outros em termos de precisão de previsão. Todavia, para os meses de junho a novembro (inverno e primavera), o método ELM obteve os melhores resultados.

No presente estudo, o invólucro, conhecido por utilizar a “força bruta” (GUYON e ELISSEEFF, 2003) foi o que apresentou as melhores capacidades de generalização. Os invólucros resultam em uma estrutura induzida com alta precisão preditiva. Contudo, a sua complexidade computacional é uma das grandes desvantagens do método. O outro método neuronal empregado, BNNAR, foi o que obteve os melhores resultados para as métricas RMSE, MAE, MAPE e MF. Este método, assim como o invólucro ELM, apresentou uma complexidade computacional que cresce de forma explosiva ao empregar a validação cruzada de origem deslizante com recalibração.

Em relação à estatística CC, o algoritmo DTSF foi extremamente rápido quando comparado aos outros métodos estudados para todos os meses do ano, usando uma fração do tempo. A grande vantagem deste método, dado que é um método orientado a dados, é que quanto maior

a quantidade de dados empregada melhor será a sua capacidade de generalização e, em um ambiente conhecido como a era do *big data*, este obterá vantagem frente aos outros métodos.

Quando da previsão para todos os dias do ano de 2016 os resultados indicam que, em um horizonte de uma semana com previsão horária, o erro de previsão para os diferentes modelos varia bastante. Portanto, não há um melhor modelo para prever a demanda de água todos os dias do ano.

## 8 RECOMENDAÇÕES E TRABALHOS FUTUROS

Como possibilidades de trabalhos futuros, temos algumas alternativas que são descritas a seguir:

- Avaliar os métodos empregados no presente estudo para prever a demanda de água urbana para periodicidades semanais, mensais e anuais visto que a maioria da literatura referenciada aborda previsões de curto prazo;
- Empregar técnicas de pós-processamento para estimar os intervalos de confiança;
- Aplicar os métodos utilizados no presente estudo para outras ZA que possuam comportamentos distintos da série temporal de demanda, como área por exemplo, áreas onde possuam uma predominância de consumidores industriais;
- Ampliar o número de modelos híbridos heterogêneos utilizados já que a integração de múltiplos métodos tem a vantagem de melhorar a capacidade de generalização;
- Utilizar outros filtros descritos na literatura no pré-processamento das séries temporais de demanda de água visto que diversos pesquisadores concluíram que sem o devido pré-processamento os métodos de *machine learning* podem se tornar instáveis e produzir resultados abaixo do ideal. As grandes vantagens dos filtros dizem respeito a sua robustez a superajustes, melhorando o desempenho dos modelos e a sua execução rápida;
- Empregar o DTSF acoplado ao simulador hidráulico EPANET 2 (ROSSMAN, 2000), desenvolvido pela *Environmental Protection Agency* (EPA), para simular o comportamento hidráulico de redes de abastecimento pressurizadas; e
- Como não há um método ideal para prever em todas as situações e para todos os dias do ano sugere a criação de um chaveamento. Chaveamento este que irá buscar acompanhar os menores MSE criando uma envoltória que minimizará o erro de previsão.

## REFERÊNCIAS

ADAMOWSKI, J. F. Peak Daily Water Demand Forecast Modeling Using Artificial Neural Networks. *Journal of Water Resources Planning and Management*, v. 134, n. 2, p. 119-128, 2008.

ADAMOWSKI, J.; CHANG, H. F. A wavelet neural network conjunction model for groundwater level forecasting. *Journal of Hydrology*, v. 407, p. 28-4, 2011.

ADAMOWSKI, J.; CHANG, H. F.; PRASCHER, S. O.; OZGA-ZIELINSKI, B.; SLIUSARIEVA, A. Comparison of multiple linear and nonlinear regression. autoregressive integrated moving average. artificial neural network. and wavelet artificial neural network methods for urban water demand forecasting in Montreal. Canada. *Water Resources Research*, v. 47, p. 1-14, 2012.

ADAMOWSKI, J.; KARAPATAKI, C. Comparison of multivariate regression and artificial neural networks for peak urban water-demand forecasting: Evaluation of different ANN learning algorithms. *Journal of Hydrologic Engineering*, v. 15, n. 10, p. 729-743, 2010.

ADAMOWSKI, K.; ADAMOWSKI, J. F.; SEIDOU, O.; OZGA-ZIELIŃSKI, B. Weekly urban water demand forecasting using a hybrid wavelet–bootstrap–artificial neural network approach. In: ANNALS OF WARSAW UNIVERSITY OF LIFE SCIENCES – SGGW LAND RECLAMATION, v. 46, n. 3, p. 197-204, 2014.

ADHIKARI, R. A neural based linear ensemble framework for time series forecasting. *Neurocomputing*, v. 157, p. 231-242, 2015.

ADHIKARI, R.; AGRAWAL, R. K. A Homogeneous Ensemble of Artificial Neural Networks for Time Series Forecasting. *International Journal of Computer Applications*, v. 32, n. 7, p. 1-8, 2011.

AGUIAR, J. L. N. Medidas de similaridade entre séries temporais. 2016. 75 p. Dissertação (Mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2016.

AKAIKE, H. Fitting autoregressive models for prediction. *Annals of the Institute of Statistical Mathematics*, v. 21, n. 1, p. 243-247, 1969.

AKAIKE, H. Maximum likelihood identification of Gaussian autoregressive moving average models. *Biometrika*, v. 60, n. 2, p. 255-265, 1973.

- ALBARD, M. A. A.; TIUN, S. Extreme Learning Machine: A review. *International Journal of Applied Engineering Research*, v. 12, n. 14, p. 4610-4623, 2017.
- ALTUNKAYNAK, A.; ÖZGER, M.; ÇAKMAKCI, M. Water consumption prediction of Istanbul city by using fuzzy logic approach. *Water Resources Management*, v. 19, p. 641-654, 2005.
- ALTUNKAYNAK, A.; NIGUSSIE, T. A. Monthly Water Consumption Prediction Using Season Algorithm and Wavelet Transform–Based Models. *Journal of Water Resource Planning and Management*, v. 143, n. 6, 2017.
- AL-ZAHRANI, M. A.; ABO-MONASAR, A. Urban residential water demand prediction based on artificial neural networks and time series models. *Water Resources Management*, v. 29, p. 3651–3662, 2015.
- ALVES, M. F.; LOTUFO, A. D. P.; LOPES, M. L. M. Seleção de variáveis stepwise aplicadas em redes neurais artificiais para previsão de demanda de cargas elétricas. *Proceeding Series of the Brazilian Society of Applied and Computational Mathematics*, v. 1, n. 1, 2013.
- ARANDIA, E.; BA, A.; ECK, B.; MCKENNA, S. Tailoring seasonal time series models to forecast short-term water demand. *Journal of Water Resource Planning and Management*, v. 142, n. 3, 2016.
- ARBUÉS, F; VALIÑAS, M. A. G.; ESPINEIRA, R. M. Estimation of residential water demand: a state-of-the-art review. *The Journal of Socio-Economics*, v. 32, p. 81-102, 2003.
- ARMSTRONG, J. S. Combining Forecasts. In: ARMSTRONG, J. S. (ed.) Principles of Forecasting: A Handbook for Researchers and Practitioners. Springer Science Business Media, New York, 2001.
- AVNIMELECH, R.; INTRATOR, N. Boosting regression estimators. *Neural Computation*, v. 11, p. 499-520, 1999.
- AZADEH, A.; NESHAT, N.; HAMIDIPOUR, H. Hybrid Fuzzy regression - artificial neural network for improvement of short-term water consumption estimation and forecasting in uncertain and complex environments: case of a large metropolitan city. *Journal of Water Resource Planning and Management*, v. 138, n. 1, p. 71-75, 2012.

BABEL, M. K.; MARPON, N.; SHINDE, V.R. Incorporating future climate and socioeconomic variables in water demand forecasting: A case study Bangkok. *Water Resources Management*, v. 28, p. 2049–2062, 2014.

BAI, Y.; WANG, P.; LI, C.; XIE, J.; WANG, Y. A multi-scale relevance vector regression approach for daily urban water demand forecasting. *Journal of Hydrology*, v. 517, p. 236-245, 2014.

BARTLETT, P. L. The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network. *IEEE Transactions on Information Theory*, v. 44, n. 2, p. 525-536, 1998.

BARROW, D. K.; CRONE, S. F. Cross-validation aggregation for combining autoregressive neural network forecasts. *International Journal of Forecasting*, v. 32, n. 4, p. 1120-1137, 2016.

BATES, J. M.; GRANGER, C. W. J. Combination of forecasts. *Operations Research Quarterly*, v. 20, n. 4, p. 451-468, 1969.

BATISTA, G. E. A. P. A.; WANG, X.; KEOGH, E. J. A complexity-invariant distance measure for time series. In: PROCEEDINGS OF THE ELEVENTH SIAM INTERNATIONAL CONFERENCE ON DATA MINING, SDM, 2011. p. 699-710.

BAXT, W. G. Improving the accuracy of an artificial neural network using multiple differently trained networks. *Neural Computation*, v. 4, p. 772-780, 1992.

BECKENBACK, E.; BELLMAN, R. *Inequalities*. Berlin: Springer-Verlag, 1961. 198 p.

BERGMEIR, C. N. *New approaches in time series forecasting: methods, software and evaluation procedures*. 2013. Thesis (Doctoral) - University Granada, Granada, Espanha, 2013.

BERGMEIR, C.; BENÍTEZ, J. M. On the use of cross-validation for time series predictor evaluation. *Information Sciences*, v. 191, p. 192-213, 2012.

BERGMEIR, C.; HYNDMAN, R. J.; BENÍTEZ, J. M. Bagging exponential smoothing methods using STL decomposition and Box-Cox transformation. *International Journal of Forecasting*, v. 32, p. 303-312, 2016.

BILLAH, B.; KING, M. L.; SNYDER, D.; KOEHLER, C. Exponential smoothing model selection for forecasting. *International Journal of Forecasting*, v. 22, n. 2, p. 239-247, 2006.

BIRKHOFF, G. D. Proof of the ergodic theorem. *Proceedings of the National Academy of Sciences of the United States of America (PNAS)*, v. 17, n. 12, p. 656-660, 1931.

- BISHOP, C. M. *Pattern Recognition and Machine Learning*. New York: Springer-Verlag, 2006.
- BLUM, A.; LANGLEY, P. Selection of relevant features and examples of machine learning. *Artificial Intelligence*, v. 97, n. 1-2, p. 245-271, 1997.
- BOCCATO, L.; SOARES, E. S.; FERNANDES, M. M. L. P.; SORIANO, D. C.; ATTUX, R. Unorganized Machines: From Turing's Ideas to Modern. Connectionist Approaches. *International Journal of Natural Computing Research*, v. 2, n. 4, p. 1-16, 2011.
- BONISSONE, P. P. Soft computing: convergence of emerging reasoning technologies. *Soft Computing*, v. 1, p. 6-18, 1997.
- BORGES, V. M. N. A. *Acomplamento de um modelo de previsão de demanda de água a um modelo simulador em tempo real – um estudo de caso: sistema adutor metropolitano de São Paulo*. 2003. 205 p. Dissertação (Mestrado) – Escola Politécnica, Universidade de São Paulo, São Paulo, 2003.
- BOROVYKH, A.; BOHTE, S.; OOSTERLEE, C. W. Conditional Time Series Forecasting with Convolutional Neural Networks. *Machine Learning*. Cornell University Library, 2017. <https://arxiv.org/pdf/1703.04691.pdf>.
- BOUGADIS, J.; ADAMOWSKI, K.; DIDUCH, R. Short-term municipal water demand forecasting. *Hydrology Processes*, v. 19, p. 137-148, 2005.
- BOWMN, A.; AZZALINI, A. *Applied smoothing techniques for data analysis: The Kernel approach with S-Plus illustrations*. Oxford: Oxford University press, 1997.
- BOX, G. E. P.; COX, D. R. An analysis of transformations. *Journal of the Royal Statistical Society, Series B*, v. 26, n. 2, p. 211-252, 1964.
- BOX, G. E. P.; JENKINS, G. M. *Time Series Analysis: Forecasting and Control*. San Francisco: Holden-Day, 2<sup>th</sup> Edition, 1976.
- BOX, G. E. P.; JENKINS, G. M; REINSEL, G. *Time series analysis: Forecasting and control*. Oxford: Wiley, 4<sup>th</sup> Edition, 2008.
- BRAGA, A. P.; CARVALHO, A. C. P. F.; LUDERMIR, T. B. *Fundamentos de redes neurais artificiais*. Rio de Janeiro: 11<sup>a</sup> Escola de Computação/Universidade Federal do Rio de Janeiro, 1998. 246 p.

- BREIMAN, L. Heuristics of instability and stabilization in model selection. *Technical Report* N° 416, 1994. Disponível em:  
<<https://pdfs.semanticscholar.org/cb6d/cc7f60ac837359433f5bb42001c0d8557180.pdf>>  
Acessado em 21/11/2018.
- BREIMAN, L. Bagging predictors. *Machine Learning*, v. 24, n. 2, p. 123-140, 1996.
- BREIMAN, L. Arcing Classifiers. *The Annals of Statistics*, v. 26, n. 3, p. 801-849, 1998.
- BREIMAN, L. Random Forest. *Machine Learning*, v. 45, n. 1, p. 5-32, 2001.
- BREIMAN, L.; FRIEDMAN, J. H.; OSHEN, R. A.; SRONE, C. J. *Classification and regression trees*. New York: Chapman & Hall, 1984.
- BRENTAN, B. M.; LUZIVOTO, E. JR.; HERRERA, M.; IZQUIERDO, J.; PÉREZ-GARCIA, R. Hybrid regression model for near real-time urban water demand forecasting. *Journal of Computational and Applied Mathematics*, 2016. [doi:10.1016/j.cam.2016.02.009](https://doi.org/10.1016/j.cam.2016.02.009).
- BROCKWELL, P. J.; DAVIS, R. A. *Introducing to Time Series and Forecasting*. New York: Springer-Verlag, 2<sup>th</sup> Edition, 2002. 437 p.
- BROWN, R. G. *Statistical forecasting for inventory control*. New York: McGraw-Hill, 1959.
- BROWN, G.; WYATT, J. L.; HARRIS, R.; YAO, X. Diversity creation methods: A survey and categorization. *Journal of Information Fusion*, v. 6, n. 1, p. 1-28, 2005.
- BUJA, A.; STUETZLE, W. *Observations on bagging*. 2000. Disponível em:  
<<http://stat.wharton.upenn.edu/~buja/PAPERS/sinica-bagging-buja-stuetzle.pdf>> Acessado em: 21/11/2018.
- CAIADO, J. Performance of combined double seasonal univariate time series models for forecasting water demand. *Journal of Hydrologic Engineering*, v. 15, n. 3, p. 215-222, 2010.
- CAMPISI-PINTO, S.; ADAMOWSKI, J.; ORON, G. Forecasting urban water demand via wavelet-denoising and neural network models. Case study: City of Syracuse, Italy. *Water Resources Management*, v. 26, p. 3539-3558, 2012.
- CAO, J.; LIN, Z. Extreme Learning Machines on High Dimensional and Large Data Applications: A Survey. *Mathematical Problems in Engineering*, v. 2015, p. 1-13, 2015.
- CERNY, V. Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm. *Journal of Optimization Theory and Applications*, v. 45, p. 41-51, 1985.

CHA, S. H. Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences*, v. 1, n. 4, p. 300-307, 2007.

CHANG, H.; PARANDVASH, H.; SHANDAS, V. Spatial variations of single-family residential water consumption in Portland, Oregon. *Urban Geography*, v. 31, p. 953-972, 2010.

CHATTOPADHYAY, A.; NABIZADEH, E.; HASSANZADEH, P. Analog forecasting of extreme-causing weather patterns using deep learning. 2019. DOI: <https://arxiv.org/abs/1907.11617>

CHEN, G; LONG, T.; XIONG, J.; BAI, Y. Multiple random forest modelling for urban water consumption forecasting. *Water Resources Management*, v. 31, n. 15, p. 4715-4729, 2017.

CHEN, X.; DONG, Z. Y.; MENG, K.; XU, Y.; WONG, K. P.; NGAN, H. W. Electricity price forecasting with extreme learning machine and bootstrapping. *IEEE Transactions on Power Systems*, v. 27, n. 4, p. 2055-2062, 2012.

CHRISTODOULOS, C.; MICHALAKELIS, C.; VAROUTAS, D. Forecasting with limited data: Combining ARIMA and diffusion models. *Technological Forecasting & Social Change*, v. 77, n. 4, p. 558-565, 2010.

CLEMEN, R. T. Combining forecasts: A review and annotated bibliography. *International Journal of Forecasting*, v. 5, p. 559-583, 1989.

CLEVELAND, W. S. Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, v. 74, n. 368, p. 829-836, 1979.

CLEVELAND, R. B.; CLEVELAND, W. S.; McRAE, J.; TERPENNING, I. STL: A seasonal-trend decomposition procedure based on Loess. *Journal of Official Statistics*, v. 6, p. 3-73, 1990.

CLEVELAND, W. S.; LOADER, C. Smoothing by Local Regression: Principles and Methods. In: HÄRDLE, W.; SCHIMEK, M. G. (eds) *Statistical Theory and Computational Aspects of Smoothing. Contributions to Statistics*. Heidelberg: Physica-Verlag HD, 1996.

COSTA, M. A.; MINETI, L. B. DTScanF: Dynamic Time Scan Forecasting (version v0.1). Zenodo. 2019. DOI: <http://doi.org/10.5281/zenodo.2603008>

COSTA, M. A.; MINETI, L. B.; PRATES, M. O.; CÁRDENAS, R. R. Dynamic Time Scan Forecasting. 2019. DOI: <http://arxiv.org/abs/1906.05399v1>

- CRONE, S. F.; KOURENTZES, N. Feature selection for time series prediction – A combined filter and wrapper approach for neural networks. *Neurocomputing*, v. 73, p. 1923-1936, 2010.
- CYBENKO, G. Approximation by superpositions of a sigmoid function. *Mathematics of Control, Signals and Systems*, v. 2, p. 303-314, 1989.
- DAHL, C. M.; HYLLEBERG, S. Flexible regression models and relative forecast performance. *International Journal of Forecasting*, v. 20, n. 2, p. 201-207, 2004.
- De LIVERA, A. M.; HYNDMAN, R. J.; SNYDER, R. D. Forecasting time series with complex seasonal patterns using exponential smoothing. *Journal of the American Statistical Association*, v. 106, n. 496, p. 1513-1527, 2011.
- DENG, W.; ZHENG, Q.; CHEN, L. Regularized extreme learning machine. In: IEEE SYMPOSIUM ON COMPUTATIONAL INTELLIGENCE AND DATA MINING (CIDM 2009), Nashville. 2009. p. 389-395.
- DI, C.; YANG, X.; WANG, X. A four-stage hybrid model for hydrological time series forecasting. *PLoS One*, v. 9, n. 8, p. 1-18, 2014.
- DICKEY, D. A.; FULLER, W. A. Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American Statistical Association*, v. 74, n. 366, p. 427-31, 1979.
- DICKEY, D. A.; FULLER, W. A. A likelihood ratio statistics for autoregressive time series with a unit root. *Econometrica*, v. 49, n. 4, p. 1057-72, 1981.
- DIEBOLD, F. X.; PAULY, P. The use of prior information in forecast combination. *International Journal of Forecasting*, v. 6, n. 4, p. 503-508, 1990.
- DIETTERICH, T. G. Machine learning research: four current direction. *AI Magazine*, v. 18, n. 4, p. 97-136, 1997.
- DIETTERICH, T. G. Ensemble methods in machine learning. *Multiple Classifier Systems Lecture Notes in Computer Science*, v. 1857, p. 1–15, 2000.
- DING, S.; ZHAO, H.; ZHANG, Y.; XU, X.; NIE, R. Extreme learning machine: algorithm. theory and applications. *Artificial Intelligence Review*, v. 44, n. 1, p. 103-115, 2015.
- DONKOR, E. A.; MAZZUCHI, T. A.; SOYER, R.; ROBERSON, J. A. Urban water demand forecasting: Review of methods and models. *Journal of Water Resources Planning and Management*, v. 140, n. 2, p. 146-159, 2014.

DU, K.; ZHAO, Y.; LEI, J. The incorrect usage of singular spectral analysis and discrete wavelet transform in hybrid models to predict hydrological time series. *Journal of Hydrology*, v. 552, p. 44-51, 2017.

EFRON, B. Bootstrap methods: another look at the jackknife. *The Annals of Statistics*, v. 7, n. 1, p. 1-26, 1979.

EVARISTO, R. M. Métodos de reamostragem de séries temporais baseados em wavelets. 2010. 53 p. Dissertação (Mestrado) – Escola Politécnica da Universidade de São Paulo, São Paulo, 2010.

FAHLMAN, S. E. An empirical study of learning speed in back-propagation networks. *Technical Report*, CMU-CS-88-162, Carnegie Mellon University, Computer Science Department, Pittsburgh, Pennsylvania, 1988.

FAHLMAN, S. E. Faster-learning variations on back-propagation: an empirical study. In: PROCEEDINGS OF THE 1988 CONNECTIONIST MODELS SUMMER SCHOOL, Pittsburgh. PA: Morgan Kaufmann, 1989.

FALKENBERG, A. V. Previsão de consumo urbano de água de curto prazo. 2005. 89 p. Dissertação (Mestrado) – Universidade Federal do Paraná, Curitiba, 2005.

FAVA, V. L. Metodologia de Box-Jenkins para Modelos Univariados. In: VASCONCELLOS, M. A. S.; ALVES, D. *Manual de econometria: nível intermediário*. São Paulo: Atlas, 2000. 308 p.

FAWAZ, H. I.; FORESTIER, G.; WEBER, J.; IDOUMGHAR, L.; MULLER, P. - A. Deep learning for time series classification: a review. *Data Mining and Knowledge Discovery*, v. 33, p. 917-963, 2019.

FERRERO, C. A. Algoritmo k-NN para previsão de dados temporais: funções de previsão e critérios de seleção de vizinhos próximos aplicados a variáveis ambientais em limnologia. 2009. 129 p. Dissertação (Mestrado) - Universidade de São Paulo, São Carlos, 2009.

FILDES, R.; HIBON, M.; MAKRIDAKIS, S.; MEADE, N. Generalising about univariate forecasting methods: further empirical evidence. *International Journal of Forecasting*, v. 14, p. 339-358, 1998.

- FIRAT, M.; TURAN, M. E.; YURDUSEV, M. A. Comparative analysis of fuzzy inference systems for water consumption time series prediction. *Journal of Hydrology*, v. 374, p. 235-241, 2009a.
- FIRAT, M.; TURAN, M. E.; YURDUSEV, M. A. Comparative analysis of neural network techniques for predicting water consumption time series. *Journal of Hydrology*, v. 384, p. 46-51, 2010.
- FIRAT, M.; YURDUSEV, M. A.; TURAN, M. E. Evaluation of Artificial Neural Network Techniques for Municipal Water Consumption Modeling. *Water Resources Management*, v. 23, p. 617-632, 2009b.
- FULLER, W. A. *Introduction to statistical time series*. New York: John Wiley, 1976. 424 p.
- FULLERTON, T. M. JR.; CEBALLOS, A.; WALKE, A. G. Short-term forecasting analysis for municipal water demand. *Journal American Water Works Association*, v. 108, n. 1, p. 27-38, 2016.
- FREITAS, A. A. C. *Previsão de séries temporais via seleção de variáveis, reconstrução dinâmica, ARMA-GARCH e redes neurais artificiais*. 2007. 160 p. Tese (Doutorado) - Universidade Estadual de Campinas, Campinas, 2007.
- FREITAS, M. A. V.; SANTOS, A. H. M. Importância da Água e da Informação Hidrológica. In: *O Estado das Águas no Brasil*. Brasília. Ed. ANEEL/MME/MMA-SRH/OMM, 1 ed., 1999. p. 13-16.
- FREUND, Y.; SCHAPIRE, R. E. Experiments with a new boosting algorithm. *In Proceedings of the International Conference on Machine Learning*, p. 148-156, 1996.
- FRIEDMAN, J. H. Multivariate adaptive regression splines. *The Annals of Statistics*, v. 19, n. 1, p. 1-67, 1991.
- GAGLIARDI, F., ALVISI, S., KAPELAN, Z., FRANCHINI, M. A Probabilistic Short-Term Water Demand Forecasting Model Based on the Markov Chain. *Water*, v. 9, n. 507, 2017.
- GARDNER Jr., E. S. Exponential smoothing: the state of the art. *Journal of Forecasting*, v. 4, n. 1, p. 1-28, 1985.
- GARDNER Jr. E. S. Exponential smoothing: the state of the art - Part II. *International Journal of Forecasting*, v. 22, p. 637-666, 2006.

GASHLER, M.; GIRAUD-CARRIER, C.; MARTINEZ, T. Decision Tree Ensemble: Small Heterogeneous Is Better Than Large Homogeneous. In: 7<sup>TH</sup> INTERNATIONAL CONFERENCE ON MACHINE LEARNING AND APPLICATIONS, San Diego. 2008.

GEMAN, S.; BIENENSTOCK, E.; DOURSAT, R. Neural network and the Bias/Variance Dilemma. *Neural Computation*, v. 4, p. 1-58, 1992.

GHALEHKHONDABI, I.; ARDJMAND, E.; YONG II, W. A.; WEECKMAN, G. R. Water demand forecasting: review of soft computing methods. *Environmental Monitoring and Assessment*, p. 189-313, 2017.

GHIASSI, M.; ZIMBRA, D. K.; SAIDANE, H. Urban water demand forecasting with a dynamic artificial neural network model. *Journal of Water Resource Planning and Management*, v. 134, n. 2, p. 138-146, 2008.

GIUSTI, R.; BATISTA, G. E. A. P. A. An empirical comparison of dissimilarity measures for time series classification. In: BRAZILIAN CONFERENCE ON INTELLIGENT SYSTEMS, 2013. p. 82-88.

GLAZ, J.; POZDNYAKOV, V.; WALLENSTEIN, S. *Scan statistics: methods and applications*. Boston: Birkhäuser Publishing Ltd., 2009. 394 p.

GOUTTE, C. Note on free lunches and cross-validation. *Neural Computation*, v. 9, n. 6, p. 1245-1249, 1997.

GRANGER, C. W. J.; NEWBOLD, P.  $R^2$  and the transformation of regression variables. *Journal of Econometrics*, v. 4, p. 205-210, 1976.

GRANGER, C. W. J. Combining forecast – Twenty years later. *Journal of Forecasting*, v. 8, p. 167-173, 1989.

GRANGER, C. W. J.; RAMANATHAN, R. Improved methods of combining forecast. *Journal of Forecasting*, v. 3, p. 197-204, 1984.

GRANITTO, P. M.; VERDES, P. F.; CECCATTO, H. A. Neural network ensembles: evaluation of aggregation algorithms. *Artificial Intelligence*, v. 163, p. 139-162, 2005.

GROPPO, G. S.; COSTA, M. A.; LIBÂNIO, M. Predicting water demand: a review of the methods employed and future possibilities. *Water Supply*, ws2019122, 2019.

GUERRERO, V. M. Time-series analysis supported by power transformations. *Journal of Forecasting*, v. 12, n. 1, p. 37-48, 1993.

- GUJARATI, D. N. *Econometria básica*. São Paulo: Makron Books, 2000. 845 p.
- GUYON, I.; ELISSEEFF, A. An introduction to variable and feature selection. *Journal of Machine Learning Research*, v. 3, p. 1157-1182, 2003.
- HAGIWARA M.; SATO A., Analysis of momentum term in backpropagation, *IEICE Transactions on Information and Systems*, v. E78-D, n° 8, 1995.
- HAYKIN, S. *Redes Neurais: princípios e prática*. Porto Alegre: Bookman, 2ª edição., 2001. 898 p.
- HANSEN, L. K.; SALAMON, P. Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 12, n. 10, p. 993-1001, 1990.
- HASSANI, H. Singular Spectrum Analysis: Methodology and Comparison. *Journal of Data Science*, v. 5, p. 239-257, 2007.
- HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. *Elements of Statistical Learning*. New York: Springer, 2<sup>th</sup> Edition, 2009. 764 p.
- HECHT-NIELSEN, R. *Neurocomputing*. New York: Addison-Wesley Publishing Company, 1990. 433 p.
- HEINERMANN, J.; KRAMER, O. Machine learning ensembles for wind power prediction. *Renewable Energy*, v. 89, p. 671-679, 2016.
- HENDRY, D. F.; CLEMENTS, M. P. Polling of forecasts. *The Econometric Journal*, v. 7, p. 1-31, 2004.
- HERRERA, M.; TORGO, L.; IZQUIERDO, J.; PÉREZ-GARCÍA, R. Predictive models for forecasting hourly urban water demand. *Journal of Hydrology*, v. 387, p. 141-150, 2010.
- HOLT, C. C. Forecasting seasonals and trends by exponentially weighted moving average. *Office of Naval Research (ONR)*, v. 52, p. 297-307, 1957.
- HORNIK, K. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, v. 4, n. 2, p. 251-257, 1991.
- HORNIK, K.; STINCHCOMBE, M.; WHITE, H. Multilayer feedforward networks are universal approximators. *Neural Network*, v. 2, n. 5. P. 359-366, 1989.

- HOUSE-PETERS, L.; PRATT, B.; CHANG, H. Effects of urban spatial structure, sociodemographics, and climate on residential water consumption in Hillsboro, Oregon. *Journal of the American Water Resource Association*, v. 46, n. 3, p. 461-472, 2010.
- HSU, H.H.; HSIEH, C.CH.; LU, M. D. Hybrid feature selection by combining filters and wrappers. *Expert Systems with Applications*, v. 38, p. 8144-8150, 2011.
- HUANG, G. B.; CHEN, L. Enhanced random search based incremental extreme learning machine. *Neurocomputing*, v. 71, n. 16-18, p. 3460-3468, 2008.
- HUANG, G.; HUANG, G. B.; SONG, S.; YOU, K. Trends in extreme learning machines: A review. *Neural Network*, v. 61, p. 32-48, 2015.
- HUANG, G. B.; ZHOU, H.; DING, X.; ZHANG, R. Extreme Learning Machines for Regression and Multiclass Classification. *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, v. 42, n. 2, p. 513-529, 2012.
- HUANG, G. B.; ZHU, Q. Y.; SIEW, C. K. Extreme Learning Machine: a new learning scheme of feedforward neural network. *IEEE International Joint Conference Neural Network*, v. 4, n. 1, p. 985-990, 2004.
- HUANG, G. B.; ZHU, Q. Y.; SIEW, C. K. Extreme Learning Machine: theory and applications. *Neurocomputing*, v.70, p. 489-501, 2006.
- HUANG, G. B.; WANG, D. H.; LAN, Y. Extreme learning machines: a survey. *International Journal Machine Learning & Cybernetics*, v. 2, p. 107-122, 2011.
- HUANG, L.; ZHANG, C.; PENG, Y.; ZHOU, H. Application of a combination model based on wavelet transform and KPLS-ARMA for urban annual water demand forecasting. *Journal of Water Resources Planning and Management*, v. 140, n. 8, 2014.
- HUANG, N. E.; SHEN, Z.; LONG, S. R.; WU, M. C.; SHIH, H. H.; ZHENG, Q.; YEN, N.; TUNG, C. C.; LIU, H. H. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proceedings of the Royal Society A – Mathematical, Physical and Engineering Sciences*, v. 454, n. 1971, p. 903-995, 1998.
- HURVICH, C. M.; TSAI, C. L. Regression and time series model selection in small samples. *Biometrika*, v. 76, n. 2, p. 297-307, 1989.
- HURVICH, C. M.; TSAI, C. L. Bias of the corrected AIC criterion for underfitted regression and time series models. *Biometrika*, v. 78, n. 3, p. 499-509, 1991.

HYLLEBERG, S.; ENGLE, R.; GRANGER, C.; YOO, B. Seasonal Integration and Cointegration. *Journal of Econometrics*, v. 44, n. 1-2, p. 215-238, 1990.

HYNDMAN, R. J. Measuring forecast accuracy. 2014. Disponível em: <<https://pdfs.semanticscholar.org/af71/3d815a7caba8dff7248ecea05a5956b2a487.pdf>> acessado em 13/10/2018.

HYNDMAN, R. J.; ATHANASOPOULOS, G. *Forecasting: Principles and Practice*. OTexts. 2014. ISBN-10: 0987507109

HYNDMAN, R. J.; KHANDAKAR, Y. Automatic time series forecasting: the forecast package for R. *Journal of Statistical Software*, v. 27, n. 3, 2008.

HYNDMAN, R. J.; KOEHLER, A. Another look at measures of forecast accuracy. *International Journal of Forecasting*, v. 22, p. 679-688, 2006.

HYNDMAN, R. J.; KOEHLER, A.; SNYDER, R.; GROSE, S. A state space framework for automatic forecasting using exponential smoothing methods. *International Journal of Forecasting*, v. 18, n. 3, p. 439-454, 2002.

HYNDMAN, R. J.; KOEHLER, A.; ORD, K.; SNYDER, R. *Forecasting with exponential smoothing: the state space approach*. Berlin: Springer, 2008. 360 p.

HYNDMAN, R. J.; O'HARA-WILD, M.; BERGMEIR, C.; RAZBASH, S.; WANG, E. *Forecasting Functions for Time Series and Linear Models*, 2017. R package version 8.2.

IBRAHIM, M.; JEMEI, S.; WIMMER, G.; HISSEL, D. Nonlinear autoregressive neural network in an energy management strategy for battery/ultra-capacitor hybrid electrical vehicles. *Electric Power System Research*, v. 136, p. 262-269, 2016.

JACOBS, R. A. An adaptive least square algorithm for the efficient training of artificial neural networks. *IEEE Transactions on Circuits and Systems*, v. 36, p. 1092-1101, 1988.

JAMES, G.; WITTEN, D.; HASTIE, T.; TIBSHIRANI, R. *An Introduction to Statistical Learning with Application in R*. New York: Springer. 2013. 426 p.

JARQUE, C. M.; BERA, A. K. A test for normality of observations and regression residuals. *International Statistical Review*, v. 55, p. 163-172, 1987.

JOHN, H. G.; KOHAVI, R.; PFLEGER, K. Irrelevant Features and the subset selection problem. In: MACHINE LEARNING: PROCEEDINGS OF THE ELEVENTH INTERNATIONAL CONFERENCE, 1994. p. 121-129.

JOSE, V. R. R.; WINKLER, R. L. Simple robust averages of forecasts: Some empirical results. *International Journal of Forecasting*, v. 24, n. 1, p. 163–169, 2008.

KALMAN, R. E. A new approach to linear filtering and prediction problems. *Transactions of the ASME – Journal of Basic Engineering*, v. 82, n. 1, p. 35-45, 1960.

KAMARTHI S. V.; PITTNER S., Accelerating neural network training using weight extrapolations, *Neural Networks*, v. 12, p. 1285-1299, 1999.

KARUNASINGHA, D. S. K.; LIONG, S. Y. Enhancement of chaotic hydrological time series prediction with real-time noise reduction using Extended Kalman Filter. *Journal of Hydrology*, v. 565, p. 737-746, 2018.

KAY, S. M.; MARPLE, S. L. Spectrum analysis – a modern perspective. *Proceedings of the IEEE*, v. 69, n. 11, p. 1380-1419, 1981.

KEOGH, E. J.; PAZZANI, M. J. Relevance feedback retrieval of time series data. In: PROCEEDINGS OF THE 22<sup>ND</sup> ACM SIGIR ANNUAL INTERNATIONAL CONFERENCE ON RESEARCH AND DEVELOPMENT IN INFORMATION RETRIEVAL, 1999. p. 183-190.

KEOGH, E. J.; PAZZANI, M. J. Scaling up dynamic time warping for datamining applications. In: 6<sup>TH</sup> INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING, 2000. p. 285-289.

KHASHEI, M.; BIJARI, M. An artificial neural network  $(p, d, q)$  model for timeseries forecasting. *Expert Systems with Applications*, v. 37, p. 479-489, 2010.

KIRKPATRICK, S.; GELATT, C. D.; VECCHI, M. P. Optimization by simulated annealing, *Science*, v. 220, p. 671-680, 1983.

KOHAVI, R.; JOHN, G. H. Wrappers for feature subset selection. *Artificial Intelligence*, v. 7, p. 273-324, 1997.

KOURENTZES, N. nnfor: Time Series Forecasting with Neural Networks, 2017. R package version 0.9.2.

KOURENTZES, N; BARROW, D. K.; CRONE, S. F. Neural network ensemble operators for time series forecasting. *Expert Systems with Applications*, v. 41, p. 4235-4244, 2014.

KROGH, A; VEDELSBY, J. Neural network ensembles, cross validation, and active learning. *Advances in Neural Information Processing Systems*, v. 7, p. 231-238, 1995.

KULAIF, A. C. P. *Técnicas de regularização para máquinas de aprendizado extremo*. 2014. 70 p. Dissertação (Mestrado) – Universidade de Campinas, Campinas, 2009.

KULLDORFF, M. A spatial scan statistic. *Communications in Statistics - Theory and methods*, v. 26, n. 6, p. 1481-1496, 1997.

KULLDORFF, M. Prospective time periodic geographical disease surveillance using a scan statistic. *Journal of the Royal Statistical Society, Series A (Statistics in Society)*, v. 164, p. 61-72, 2001.

KULLDORFF, M.; ATHAS, W. F.; FEURER, E. J.; MILLER, B. A.; KEY, C. R. Evaluating cluster alarms: a space-time scan statistic and brain cancer in Los Alamos. New Mexico. *American Journal of Public Health*, v. 88, p. 1377-1380, 1998.

KÜNSCH, H. R. The jackknife and the bootstrap for general stationary observations. *Annals of Statistics*, v. 17, n. 3, p. 1217-1241, 1989.

KWIATKOWSKI, D; PHILLIPS, P. C. B.; SCHMIDT, P.; SHIN, Y. Testing the null hypothesis of stationarity against the alternative of a unit root. *Journal of Econometrics*, v. 54, p. 159-178, 1992.

LI, Y; MOURA, J. M. F. Forecaster: A Graph Transformer for Forecasting Spatial and Time-Dependent Data. 2019. DOI: <https://arxiv.org/pdf/1909.04019.pdf>

LIN, L.; WANG, F.; XIE, X.; ZHONG, S. Random forest-based extreme learning machine ensemble for multi-regime time series prediction. *Expert Systems with Applications*, v. 83, p. 164-178, 2017.

LIU, J. Q.; CHENG, W. P.; ZHANG, T. Q. Principal factor analysis for forecasting diurnal water-demand pattern using combined Rough-set and Fuzzy-Clustering technique. *Journal of Water Resource Planning Management*, v. 139, n. 1, p. 23-33, 2013.

LIU, R. Y.; SINGH, K. Efficiency and robustness in resampling. *The Annals of Statistics*, v. 20, n. 1, p. 370-384, 1992.

MACKAY, D. J. C. *Information Theory, Inference, and Learning Algorithms*. Cambridge: Cambridge University Press, 2003. 640 p.

MALDONADO, S.; LÓPEZ, J. An embedded feature selection approach for support vector classification via second-order cone programming. *Intelligent Data Analysis*, v. 19, n.6, p. 1259-1273, 2015.

MAKRIDAKIS, S.; ANDERSEN, A.; CARBONE, R.; FILDES, R.; HIBON, M.; LEWANDOWSKI, R.; NEWTON, J.; PARZEN, E.; WINKLER, R. The accuracy of extrapolation (time series) methods: Results of a forecasting competition. *Journal of Forecasting*, v. 1, p. 111-153, 1982.

MAKRIDAKIS, S.; HIBON, M. The M3-Competition: results, conclusions and implications. *International Journal of Forecasting*, v. 16, p. 451-476, 2000.

MAKRIDAKIS, S.; WINKLER, R. L. Averages of forecasts: Some empirical results. *Management Science*, v. 29, p. 987-996, 1983.

MAKRIDAKIS, S.; SPILIOTIS, E.; ASSIMAKOPOULOS, V. Statistical and machine learning forecasting methods: Concerns and way forward. *PLoS ONE*, v. 13, n. 3, p. 1-16, 2018.

MAIER, H. R.; DANDY, G. C. Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. *Environmental Modelling & Software*, v. 15, p. 101-124, 2000.

MAO, W.; ZHAO, S.; UM, S.; WANG, H. Multi-dimensional extreme learning machine. *Neurocomputing*, v. 149, p. 160-170, 2015.

MARIA ANDRÉ, D.; CARVALHO, J. R. Spatial determinants of urban residential water demand in Fortaleza. Brazil. *Water Resources Management*, v. 28, p. 2401-2414, 2014.

MARTÍNEZ-MARTÍNEZ, J. M.; ESCANDELL-MONTERO, P.; SORIA-OLIVAS, E.; MARTÍN-GUERRERO, J. D.; MAGDALENA-BENEDITO, R.; GÓMEZ-SANCHIS, J. Regularized extreme learning machine for regression problems. *Neurocomputing*, v. 74, n. 17, p. 3716-3721, 2011.

MARTÍNEZ, R. O.; ZAMPROGNO, B. Comparação de algumas técnicas de previsão em análise de séries temporais. *Revista Colombiana de Estadística*, v. 26, n. 2, p. 129-157, 2003.

MCDONALD, R. I.; WEBER, K.; PADOWSKI, J.; FLÖRKE, M.; SCHNEIDER, C.; GREEN, P. A.; GLESSON, T.; ECKMAN, S.; LEHNER, B.; BALK, D.; BOUCHER, T.; GRILL, G.; MONTGOMERY, M. Water on an urban planet: Urbanization and the reach of urban water infrastructure. *Global Environmental Change*, v. 27, p. 96-105, 2014.

MELIN, P.; SOTO, J.; CASTILLO, O.; SORIA, J. A new approach for time series prediction using ensembles of ANFIS models. *Expert Systems with Applications*, v. 39, p. 3494-3506, 2012.

MENDES-MOREIRA, J.; SOARES, C.; JORGE, AL. M.; SOUSA, J. F. Ensemble approaches for regression: A survey. *ACM Computing Surveys*, v. 45, n. 1, article 10, 40 p., 2012.

MICHE, Y.; SORJAMAA, A.; BAS, P.; SIMULA, O.; JUTTEN, C.; LENDASSE, A. OP-ELM: Optimally-pruned extreme learning machine. *IEEE Transactions on Neural Networks*, v. 21, n. 1, p. 158-162, 2010.

MICHE, Y.; VAN HEESWIJK, M.; BAS, P.; SIMULA, O.; LENDASSE, A. Trop-elm: A double-regularized elm using lars and Tikhonov regularization. *Neurocomputing*, v. 74, n. 16, p. 2413-2421, 2011.

MITCHELL, T. M. *Machine Learning*. McGraw-Hill, 1<sup>th</sup> Edition, 1997 414 p.

MONTGOMERY, D. C.; JENNINGS, C. L.; KULAHCI, M. *Introduction to time series analysis and forecasting*. Wiley-Interscience, 2<sup>th</sup> Edition, 2015 672 p.

MONTGOMERY, D. C.; PECK, E. A.; VINING, G. G. *Introduction to linear regression analysis*. Nova Jersey: John Wiley & Sons, 2<sup>th</sup> Edition, 2012. 544 p.

MOONEY, C. Z. *Monte Carlo simulation*, v. 116, Sage Publications, 1997.

MORETTIN, P. A.; TOLOI, C. M. C. *Análise de séries temporais*. São Paulo: Editora Edgard Blücher, 2004. 564 p.

NADARAYA, E. A. On estimating regression. *Theory Probability and its Applications*, v. 9, p. 141-142, 1964.

NAUGES, C.; WHITTINGTON, D. Estimation of water demand in developing countries: An overview. *The World Bank Research Observer*, v. 25, n. 2, 2010.

NAUS, J. I. The distribution of the size of the maximum cluster of points on a line. *Journal of the American Statistical Association*, v. 60, p. 532-538, 1965.

NASSERI, M.; MOEINI, A.; TABESH, M. Forecasting monthly urban water demand using Extended Kalman Filter and Genetic Programming. *Expert Systems with Applications*, v. 38, p. 7387-7395, 2011.

NEWBOLD, P.; GRANGER, C. W. J. Experience with forecasting univariate time series and the combination of forecasts. *Journal of the Royal Statistical Society, Series A*, v. 137, p. 131-146, 1974.

NIELS, R. *Dynamic Time Warping - An intuitive way of handwriting recognition*. 2004. 95 p. Dissertação (Mestrado) - Radboud University, Nijmegen, 2004.

NOURANI, V.; BAGHANAM, A. H.; ADAMOWSKI, J.; KISI, O. Applications of hybrid wavelet–Artificial Intelligence models in hydrology: A review. *Journal of Hydrology*, v. 514, p. 358-377, 2014.

OBA, S.; SATO, M.; TAKEMASA, I.; MONDEN, M.; MATSUBARA, K.; ISHII, S. A. Bayesian missing value estimation method. *Bioinformatics*, v. 19, n. 16, p. 2088-2096, 2003.

ODAN, F. K. *Estudo de confiabilidade aplicado à otimização da operação em tempo real de redes de abastecimento de água*. 2013. 210 p. Tese (Doutorado) - Universidade de São Paulo, São Carlos, 2010.

ODAN, F. K.; REIS, L. F. R. Hybrid Water Demand Forecasting Model Associating Artificial Neural Network with Fourier Series. *Journal of Water Resources Planning and Management*, v. 138, p. 245-256, 2012.

OORD, A. V. D.; DIELEMAN, S.; ZEN, H.; SIMONYAN, K.; VINYALS, O.; GRAVES, A.; KALCHBRENER, N.; SENIOR, A.; KAVUKCUOGLU, K. WaveNet: A Generative Model for Raw Audio. Sound. Cornell University Library, 2016. <https://arxiv.org/pdf/1609.03499.pdf>

ORD, J. K.; KOEHLER, A. B.; SNYDER, R. D. Estimation and prediction for class of dynamic nonlinear statistical models. *Journal of American Statistical Association*, v. 92, n. 440, p. 1621-1629, 1997.

PARK, J.; SANDBERG, I. W. Universal Approximation using Radial-Basis-Function Networks. *Neural Computation*, v. 3, p. 246-257, 1991.

PARTAL, T.; KISI, O. Wavelet and neuro-fuzzy conjunction model for precipitation forecasting. *Journal of Hydrology*, v. 342, n. 1-2, p. 199-212, 2007.

PEGELS, C. C. Exponential forecasting: some new variations. *Management Science*, v. 12, p. 311-315, 1969.

PELLEGRINI, T. R. *Uma avaliação de métodos de previsão aplicados à grande quantidade de séries temporais univariadas*. 2012. 76 p. Dissertação (Mestrado). Universidade Federal de São Carlos (UFSCar), São Carlos, 2012.

PEÑA-GUZMÁN, C.; MELGAREJO, J.; PRATS, D. Forecasting water demand in residential, commercial, and industrial zones in Bogotá, Colombia, using Least-Squares Support Vector Machines. *Mathematical Problems in Engineering*, v. 2016, 2016.

PERRON, P. The great crash. the oil price shock and the unit root hypothesis. *Econometrica*, v. 75, n. 6, p. 1361-401, 1989.

PERRON, P. Testing for a unit root in a time series with a changing mean. *Journal of Business & Economic Statistics*, v. 8, n. 2, p. 153-162, 1990.

PERRON, P. Further Evidence on Breaking Trend Functions in Macroeconomic Variables. *Journal of Econometrics*, v. 80, n. 2, p. 355-385, 1997.

PERRONE, M. P. *Improving Regression Estimation: Averaging Methods for Variance Reduction with Extensions to General Convex Measure Optimization*. 1993. 82 p. Thesis (Doctoral). Brown University, Providence, 1993.

PERRONE, M. P.; COOPER, L. N. When networks disagree: Ensemble methods for hybrid neural networks. In: MAMMONE, R. J., *Neural Networks for Speech and Image Processing*, chapter 10. Chapman-Hall, 1993.

PHANSALKAR V. V.; SASTRY P. S., Analysis of the back-propagation algorithm with momentum, *IEEE Transactions Neural Networks*, v. 5, p. 505-506, 1994.

PHILLIPS, P. C. B.; PERRON, P. Testing for a unit root in time series regression. *Biometrika*, v. 75, n. 2, p. 335-346, 1988.

PIRAMUTHU, S. Evaluating feature selection methods for learning in data mining applications. *European Journal of Operational Research*, v. 156, n. 2, p. 483-494, 2004.

POLI, I.; JONES, R. D. A neural net model for prediction. *Journal of American Statistical Association*, v. 89, n. 425, p. 117-121, 1994.

PULIDO-CALVO, I.; GUTIÉRREZ-ESTRADA, J. C. Improved irrigation water demand forecasting using a soft-computing hybrid model. *Biosystems Engineering*, v. 102, p. 202-218, 2009.

PULIDO-CALVO, I; ROLDÁN, J; LÓPEZ-LUQUE, R; GUTIÉRREZ-ESTRADA, J. C. Demand forecasting for irrigation water distribution system. *Journal of Irrigation and Drainage Engineering*, v. 129, p. 422-431, 2003.

PRECHELT, L. Automatic early stopping using cross validation: quantifying the criteria. *Neural Network*, v. 11, n. 4, p. 761-767, 1998.

PYAYT, A. L.; KOZIONOV, A. P.; KUSHERBAEVA, V. T.; MOKHOV, I. I.; KRZHIZHANOVSKAYA, V. V.; BROEKHIJDEN, B. J.; MEIJER, R. J.; SLOOT, P. M. A.

Signal analysis and anomaly detection for flood early warning systems. *Journal of Hydroinformatics*, v.16, n. 15, p. 1025-1043, 2014.

QIU, X.; ZHANG, L.; REN, Y.; SUGANTHAN, P. N.; AMARATUNGA, G. Ensemble deep learning for regression and time series forecasting. In: SYMPOSIUM ON COMPUTATIONAL INTELLIGENCE IN ENSEMBLE LEARNING (CIEL), 2014.

QUINLAN, J. R. *Bagging, Boosting and C 4.5*. University of Sidney, 2006. Disponível em: <<https://www.cs.utah.edu/~piyush/teaching/Quinlan.AAAI96.pdf>> acessado em: 19/11/2018.

RAO, C. R.; MITRA, S. K. *Generalized inverse of matrices and its applications*. New York: Wiley, 1971.

REIDMILLER, M.; BRAUN, H. A direct adaptive method for faster backpropagation learning: The RPROP algorithm. *IEEE International Conference on Neural Networks*, v. 1, p. 586-591, 1993.

REN, Y.; ZHANG, L.; SUGANTHAN, P. M. Ensemble Classification and Regression-Recent Developments, Applications and Future Directions. *IEEE Computational Intelligence Magazine*, v. 11, p. 41-53, 2016.

ROMANO, G.; SALVATI, N.; GUERRINI, A. An empirical analysis of the determinants of water demand in Italy. *Journal of Cleaner Production*, v. 130, p. 74-81, 2016.

ROMANO, M.; KAPELAN, Z. Adaptive water demand forecasting for near real-time management of smart water distribution systems. *Environmental Modelling & Software*, v.60. p. 265-276. 2014.

ROONEY, N.; PATTERSON, D.; ANAND, S.; TSYMBAL, A. Dynamic integration of regression models. In: PROCEEDINGS OF THE INTERNATIONAL WORKSHOP ON MULTIPLE CLASSIFIER SYSTEMS. *Lecture Notes in Computer Science*, v. 3181, p. 164-173, 2004.

ROSENBLATT, F. The Perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, v. 65, n. 6, p. 386-408, 1958.

ROSSMAN, L. A. EPANET 2 USERS MANUAL. U.S. Environmental Protection Agency, Washington, D.C., EPA/600/R-00/057, 2000.

RUJIS, A.; ZIMMERMANN, A.; VAN DEN BERG, M. Demand and distributional effects of water pricing policies. *Ecological Economics*, v. 66, p. 506-516, 2008.

- RUIZ, L. G. B.; CUÉLLAR, M. P.; CALVO-FLORES, M. D.; JIMÉNEZ, M. D. C. P. An application of non-linear autoregressive neural network to predict energy consumption in public buildings. *Energies*, v. 9, n. 684, p. 1-21, 2016.
- RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. *Nature*, v. 323, p. 533-536, 1986.
- SAEYS, Y.; INZA, I.; LARRAÑAGA, P. A review of feature selection techniques in bioinformatics. *Bioinformatics*, v. 23, n. 19, p. 2507-2517, 2007.
- SAKOE, H.; CHIBA, S. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*, v. 26, n. 1, p. 43-49, 1978.
- SANTHUSITHA, D.; KARUNASINGHA, K.; LIONG, S. Y. Enhancement of chaotic hydrological time series prediction with real-time noise reduction using Extended Kalman Filter. *Journal of Hydrology*, v. 565, p. 737-746, 2018.
- SANTOS, C. C.; PEREIRA, A. J. FILHO. Water demand forecasting model for the metropolitan area of São Paulo, Brazil. *Water Resources Management*, v. 28, p. 4401-4414, 2014.
- SARLE, W. S. Stopped training and other remedies for overfitting. In: SYMPOSIUM ON THE INTERFACE COMPUTING SCIENCE AND STATISTICS, 1995. p. 352-360.
- SCHEICH, J.; HILLENBRAND, T. Determinants of residential water demand in Germany. *Ecological Economics*, v. 68, p. 1756-1769, 2009.
- SCHWARZ, G. Estimating the dimension of a model. *The Annals of Statistics*, v. 6, p. 461-464, 1978.
- SHABRI, A.; SAMSUDIN, R. Empirical mode decomposition–Least Squares Support Vector Machine based for water demand forecasting. *International Journal of Advances in Soft Computing and its Application*, v. 7, n. 2, 2015.
- SHAMSELDIN, A. Y.; O'CONNOR, K. M.; LIANG, G. C. Methods for combining the outputs of different rainfall-runoff models. *Journal of Hydrology*, v. 197, p. 203-229, 1997.
- SHARKEY, A. J. C. On combining artificial neural nets. *Journal Connection Science*, v. 8, n. 3-4, p. 299-314, 1996.

SHAUB, D.; ELLIS, P. *forecastHybrid*. Title Convenient Functions for Ensemble Time Series Forecasts, 2019. R package version 4.2.17.

SHEPHERD, A. J. *Second-order methods for neural network: fast and reliable methods for Multi-Layer Perceptrons*. New York: Springer, 1997. 145 p.

SILVA, C. S. *Previsão Multivariada da Demanda Horária de Água em Sistemas Urbanos de Abastecimento*. 2003. 303 p. Tese (Doutorado) - Universidade de Campinas, Campinas, 2003.

SILVA, F. M.; ALMEIDA, L. B. Speeding up Backpropagation. In: ECKMILLER, R. (Editor), *Advanced Neural Computers*. North Holland: Elsevier Science Publishers, 1990.

SINGH, R.; BALASUNDARAM, S. Application of extreme learning machine method for time series analysis. *International Journal of Computer and Information Engineering*, v. 1, n. 11, p. 3407-3413, 2007.

SIQUEIRA, H. V. *Máquinas desorganizadas para previsão de séries de vazões*. 2013. 218 p. Tese (Doutorado) - Universidade Estadual de Campinas, Campinas, 2013.

SOLLICH, P. KROGH, A. Learning with ensembles: How overfitting can be useful. In: TOURETZKY, D.; MOZER, M.; HASSELMO, M. (Editors) *Advances in Neural Information Processing Systems*, v. 8, p. 190-196. The MIT Press, 1996.

SORJAMAA, A.; HAO, J.; REYHANI, N.; JI, Y.; LENDASSE, A. Methodology for long-term prediction of time series. *Neurocomputing*, v. 70, p. 2861-2869, 2007.

SPECHT, D. F. A general regression neural network. *IEEE Transactions on Neural Networks*, v. 2, n. 6, p. 568-576, 1991.

STAŃCZYK, U. Feature evaluation by filter, wrapper, and embedded approaches. In: STAŃCZYK, U.; JAIN, L. C. (Editors.) *Feature Selection for Data and Pattern Recognition. Studies in Computational Intelligence*, v. 584, p. 29-44 Springer, 2015.

STONE, C. J. Consistent nonparametric regression. *The Annals of Statistics*, v. 5, n. 4, p. 595-620, 1977.

STONE, M. Cross-validation choice and assessment of statistical predictions. *Journal of the Royal Statistical Society, Serie B*, v. 2, n. 36, p. 111-147, 1974.

TASHMAN, L. J. Out-of-sample tests of forecasting accuracy: an analysis and review. *International Journal of Forecasting*, v. 16, p. 437-450, 2000.

TAYLOR, J. W. Exponential smoothing with a damped multiplicative trend. *International Journal of Forecasting*, v. 19, n. 4, p. 715-725, 2003.

TEALAB, A.; HEFNY, H.; BADR, A. Forecasting of nonlinear time series using ANN. *Future Computing and Informatics Journal*, v. 2, p. 39-47, 2017.

TIBSHIRANI, R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Serie B*, v. 58, n. 1, p. 267-288, 1996.

TIWARI, M. K.; ADAMOWSKI, J. Urban water demand forecasting and uncertainty assessment using ensemble wallet-bootstrap-neural network models. *Water Resources Research*, v. 49, p. 6486-6507, 2013.

TIWARI, M. K.; ADAMOWSKI, J. Medium term water demand forecasting with limited data using an ensemble Wavelet-Bootstrap-Machine-Learning approach. *Journal of Water Resources Planning and Management*, v. 141, n. 2, 2015.

TIWARI, M. K.; ADAMOWSKI, J.; ADAMOWSKI, K. Water demand forecasting using extreme learning machines. *Journal of Water and Land Development*, n. 28, p. 37–52, 2016.

TRAUTWEIN Jr., B. Avaliação de métodos para previsão de consumo de água para curtíssimo prazo: um estudo de caso para a empresa de saneamento. 2004. 107 p. Dissertação (Mestrado) – Pontifícia Universidade Católica do Paraná, Curitiba, 2004.

TSUTIYA, M. T. *Abastecimento de Água*. Departamento de Engenharia Hidráulica e Sanitária da Escola Politécnica da Universidade de São Paulo. São Paulo: 3ª Edição, 2006. 659 p.

TURING, A. Intelligent machinery. In: EVANS. C. R.; ROBERTSON. A. D. (Editors). *Cybernetics: key papers*. Baltimore Md. and Manchester: University Park Press, 1968.

UNCU, Ö.; TÜRKSEN, I. B. A novel feature selection approach: Combining feature wrappers and filters. *Information Sciences*, v. 177, p. 449-466, 2007.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMES, A. N.; KAIZER, L.; POLOSUKHIN, I. Attention Is All You Need. 31<sup>st</sup> *Conference on Neural Information Processing Systems (NIPS)*, Long Beach, CA, USA, 2017. Disponível em: <<https://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf>> acessado em 27/12/2019.

VOGL T. P; MANGIS J. K.; RIGLER A. K.; ZINK W. T.; ALKON D. L. Accelerating the Convergence of the Back-Propagation Method. *Biological Cybernetics*, v. 59, p. 257-263, 1988.

- VÖRÖSMARTY, C. J.; GREEN, P.; SALISBURY, J.; LAMMERS, R. B. Global Water Resources: Vulnerability from Climate Change and Population Growth. *Science*, v. 289, n. 5477, p. 284-288, 2000.
- WALLIS, K. F. Combining forecast – forty years later. *Applied Financial Economics*, v. 21, p. 33-41, 2011.
- WANG, L.; WANG, Z.; QU, H.; LIU, S. Optimal forecast combination based on neural network for time series forecasting. *Applied Soft Computing*, v. 66, p. 1-17, 2018.
- WATSON, G. S. Smooth regression analysis. *Sankhya: The Indian Journal of Statistics, Series A*, v. 26, n. 4, p. 359-372, 1964.
- WEBB, G. I; ZHENG, Z. Multi-strategy ensemble learning: Reducing error combining ensemble learning techniques. *IEEE Transactions on Knowledge and Data Engineering*, v. 16, n. 8, p. 980-991, 2004.
- WELCH, G.; BISHOP, G. An introduction to the Kalman Filter. University of North Carolina at Chapel Hill, Department of Computer Science, TR 95-041, 2004.
- WEIGEND, A. S.; HUBERMAN, B. A.; RUMELHART, D. E. Predicting the future: A connectionist approach. *International Journal of Neural Systems*, v. 1, n. 3, p. 193-209, 1990.
- WENTZ, E. A.; GOBER, P. Gober Determinants of small area water consumption for the city of Phoenix. Arizona. *Water Resources Management*, v. 21, p. 1849-1863, 2007.
- WICHARD, J. D.; MERKWIRTH, C.; OGORZALEK, M. Building ensembles with heterogeneous models. 1990. Disponível em: <[www.j-wichard.de/publications/salermo\\_Incs\\_2003.pdf](http://www.j-wichard.de/publications/salermo_Incs_2003.pdf)> acessado em; 30/11/2018.
- WIDROW, B.; HOFF, M. E. Adaptive Switching Circuits, Technical Report N° 1553-1, 1960.
- WILAMOWSKI, B. M.; YU, H. Neural Network Learning without Backpropagation. *IEEE Transactions on Neural Networks*, v. 21, n. 11, p. 1793-1803, 2010.
- WINTERS, P. R. Forecasting Sales by Exponentially Weighted Moving Averages. *Management Science*, v. 6, p. 324-342, 1960.
- YANG, D.; ALESSANDRINI, S. An ultra-fast way of searching weather analogs for renewable energy forecasting. *Solar Energy*, v. 185, p. 255-261, 2019.
- YU, X. H.; CHEN, G. A. Efficient Backpropagation Learning Using Optimal Learning Rate and Momentum. *Neural Networks*, v. 10, n. 3, p. 517-527, 1997.

ZANCHETTIN, C. *Otimização global de redes neurais artificiais*. 2008. 146 p. Tese (Doutorado) - Universidade Federal de Pernambuco, Recife, 2008.

ZHANG, G. P. An investigation of neural networks for linear time-series forecasting. *Computers & Operations Research*, v. 28, n. 12, p. 1183-1202, 2001.

ZHANG, G. P. Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*, v. 50, p. 159-175, 2003.

ZHANG, G. P. A neural network ensemble method with jittered training data for time series forecasting. *Information Sciences*, v. 177, p. 5329-5346, 2007.

ZHANG, G. P.; QI, M. Neural network forecasting for seasonal and trend time series. *European Journal of Operational Research*, v. 160, n. 2, p. 501-514, 2005.

ZHOU, S. L.; McMAHON, T. A.; WALTON, A.; LEWIS, J. Forecasting daily urban water demand: A case study of Melbourne. *Journal of Hydrology*, v. 236, p. 153-164, 2000.

ZIVOT, E.; ANDREWS, D. W. K. Further evidence on the great crash the oil price shock and the unit root hypothesis. *Journal of Business and Economic Statistics*, n. 10, p. 251-270, 1992.

**PRODUÇÃO CIENTÍFICA:**

GROPPO, G. S.; COSTA, M. A.; LIBÂNIO, M. Predicting water demand: a review of the methods employed and future possibilities. *Water Supply*, ws2019122, 2019.

GROPPO, G. S.; COSTA, M. A.; LIBÂNIO, M. Previsão de demanda de água em regiões metropolitanas utilizando métodos de *machine learning* e *dynamic time scan forecasting*. In: LI SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL (SBPO), PO E INTELIGÊNCIA COMPUTACIONAL, Limeira, São Paulo, v.2, 2019 – 107292

## **ANEXO I**

### A.I.1. - Pseudo inversa Moore-Penrose generalizada

A solução de um sistema linear geral de equações

$$Ax = y$$

em que  $A$  pode ser uma matriz singular ou retangular podendo ser obtida pelo uso da pseudo inversa Moore-Penrose generalizada,

**Definição 1** (Rao e Mitra, 1971): A matriz  $G$  de tamanho  $n \times m$  é chamada de pseudo inversa generalizada Moore-Penrose de uma dada matriz  $A$  de tamanho  $m \times n$ , se

$$AGA = A, GAG = G, (AG)' = AG, (GA)' = GA$$

Por questão de conveniência, denota-se  $A$  como  $A^\dagger$ ,

**Definição 2:** Para o sistema linear geral dado  $Ax = y$ , onde  $A$  é uma matriz de tamanho  $m \times n$ ,  $y$  é um vetor em  $R^m$  e  $\hat{x}$  é um vetor em  $R^n$ , Este é chamado de solução de mínimos quadrados se

$$\|A\hat{x} - y\| = \min_x \|Ax - y\|$$

em que  $\|\cdot\|$  é a norma no espaço Euclidiano,

**Definição 3:** O vetor  $\hat{x} \in R^n$  é dito ser uma solução de mínimos quadrados da norma mínima do sistema linear geral  $Ax = y$  para qualquer  $y \in R^m$

$$\|\hat{x}\| \leq \|x\|$$

Isso significa que se a solução  $\hat{x}$  for considerada uma solução de mínimos quadrados se tiver a menor norma entre as possíveis soluções de mínimos quadrados  $x \in R^n$

**Teorema 1** (Rao e Mitra, 1971): Seja  $G$  uma matriz de tamanho  $n \times m$ , Então,  $\hat{x} = Gy$  é uma solução de mínimos quadrados da norma mínima do sistema linear geral  $Ax = y$  se, e somente se,  $G = A^\dagger$  for uma Moore-Penrose inversa generalizada da matriz  $A$ ,

A partir deste teorema fica claro que  $\hat{x} = A^\dagger y$  é a solução mínima de mínimos quadrados da norma mínima,

### A.I.2. - Desigualdade de Cauchy-Schwarz

Um ponto  $x \in \mathbb{R}^n$  é também indicado pelo vetor  $x = (x_1, x_2, \dots, x_n)$ . Dados  $x, y \in \mathbb{R}^n$ , o produto interno em  $\mathbb{R}^n$  é definido por

$$x \cdot y = x_1y_1 + x_2y_2 + \dots + x_ny_n$$

A norma euclidiana de  $x$ , ou do módulo de  $x$  é:

$$|x| = \sqrt{x \cdot x} = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2}$$

**Teorema (Desigualdade de Cauchy-Schwarz):** Vale a desigualdade

$$|x \cdot y| \leq |x||y|$$

Quaisquer que sejam  $x, y \in \mathbb{R}^n$ .

**Prova:** Temos,

$$\begin{aligned} |x \cdot y|^2 &= (x_1y_1 + x_2y_2 + \dots + x_ny_n)^2 = \sum_i x_i^2 x_i^2 + \sum_{i < j} 2 x_i y_i x_j y_j \\ &\leq \sum_i x_i^2 y_i^2 + \sum_{i < j} (x_i^2 y_j^2 + x_j^2 y_i^2) \\ &= \sum_{i,j} x_i^2 y_j^2 \\ &= (x_1^2 + \dots + x_n^2)(y_1^2 + \dots + y_n^2) \blacksquare \end{aligned}$$

A norma  $|x| = \sqrt{x \cdot x}$  definida sobre  $\mathbb{R}^n$  possui as propriedades abaixo

- ✓  $|x| \geq 0$ , para todo  $x$ , e tem-se  $|x| = 0$  se e somente se  $x = 0$ .
- ✓  $|\lambda x| = |\lambda||x|$ , para todo  $\lambda \in \mathbb{R}$  e para todo  $x \in \mathbb{R}^n$ .
- ✓  $|x + y| \leq |x| + |y|$ , quaisquer que sejam  $x, y \in \mathbb{R}^n$  (desigualdade triangular).

A desigualdade triangular segue da desigualdade de Cauchy-Schwarz. Pois,

$$\begin{aligned} |x + y|^2 &= (x + y) \cdot (x + y) = x \cdot x + 2x \cdot y + y \cdot y \\ &\leq |x|^2 + 2|x||y| + |y|^2 = (|x| + |y|)^2 \blacksquare \end{aligned}$$

## **ANEXO II**

## **Ajustes dos modelos**

Em aprendizagem de máquinas empregam-se o *tuning* (ajustes) para ajustar os modelos. Os modelos foram treinados empregando os seguintes códigos:

### **i. DTSF (*Dynamic Time Scan Forecasting*)**

```
fit_DTSF = dtsf(ts = dados1$L.S.A, poli = 1, best = 24, window = horizonte, forecast =
horizonte, reg = T)
```

### **ii. ELM (*Extreme Learning Machine*)**

```
fit_ELM = elm(serie, m = frequency(serie), hd = NULL, type = c("lasso", "ridge", "step",
"lm"), reps = 20, comb = c("median", "mean", "mode"), lags = NULL, keep = NULL,
difforder = 0, sel.lag = c(TRUE), direct = c(FALSE), allow.det.season = c(TRUE,
FALSE), det.type = c("auto", "bin", "trg"), barebone = c(FALSE, TRUE))
```

### **iii. BNNAR (*Rede Neural Autorregressivo não linear*)**

```
fit_nnetar = nnetar(serie, size=1, repeats = 100, xreg = NULL, lambda = TRUE, model
= NULL, subset = NULL, scale.inputs = TRUE)
```

### **iv. SARIMA (*Box e Jenkins*)**

```
fit_arima = auto.arima(serie, stationary = FALSE, seasonal = TRUE, stepwise = TRUE,
ic = c("aicc", "aic", "bic"), test = c("kpss", "adf", "pp"), allowdrift = FALSE, lambda =
TRUE )
```

### **v. TBATS (*Trigonometric Box-Cox transformation ARMA erros, Trend, and Seasonal Components*)**

```
fit_TBATS = tbats(serie, use.box.cox = TRUE, use.parallel = TRUE, num.cores =
NULL, use.trend = NULL, use.damped.trend = NULL, use.arma.errors = TRUE,
model= NULL)
```

### **vi. ETS (*Alisamento Exponencial*)**

```
fit_ETS_ZZZ = ets(serie, model= "ZZZ", alpha = NULL, beta = NULL, gamma =  
NULL, additive.only = FALSE, lambda = NULL, biasadj = FALSE, opt.crit = c("lik",  
"amse", "mse", "sigma", "mae"), ic = c("aicc", "aic", "bic"))
```

**vii. STL-ETS (Modelo híbrido empregando filtro STL e o ETS)**

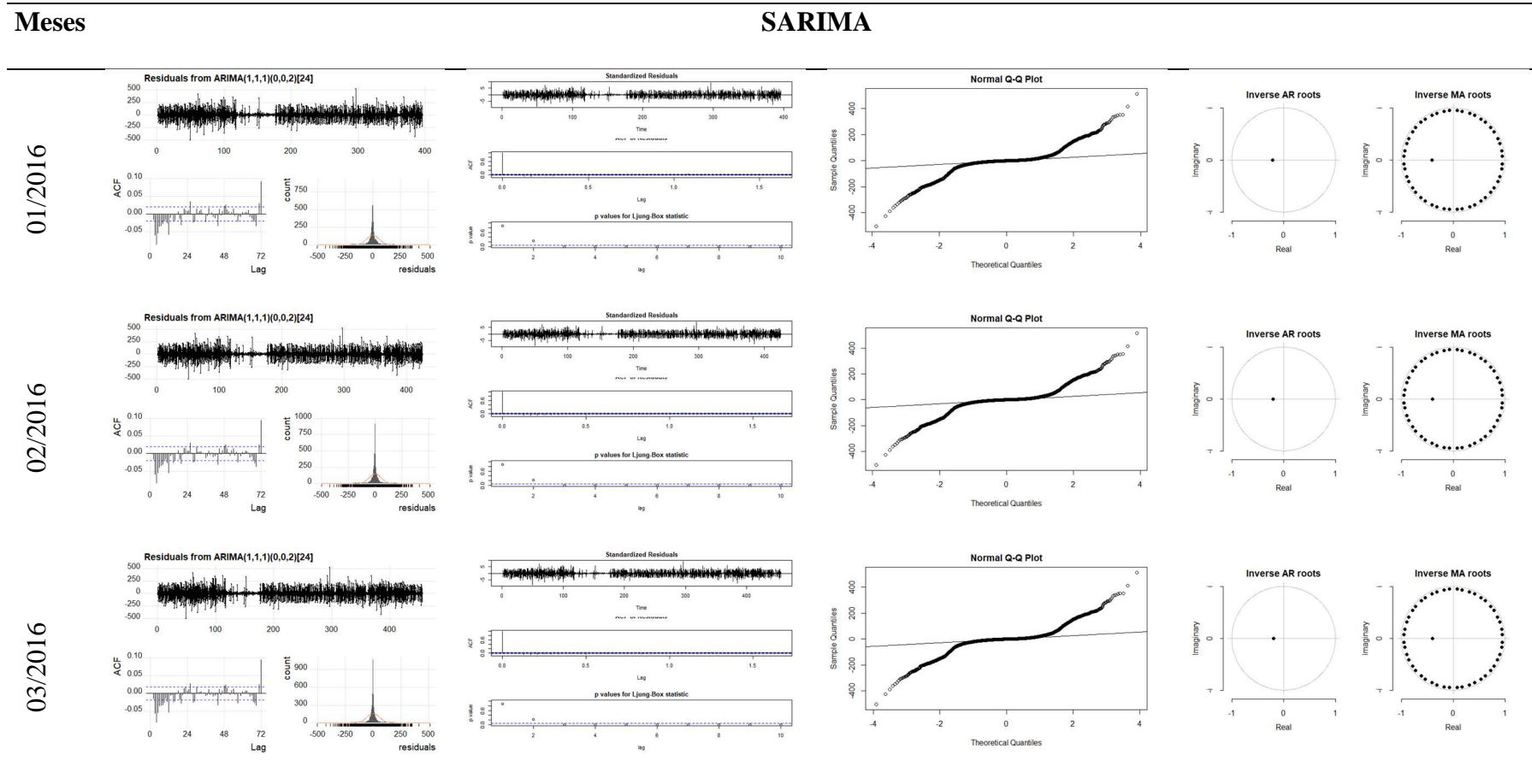
```
fit_hybrid = hybridModel(serie, models = "es", e.args = list(model= "ZZZ", damped =  
TRUE, alpha = NULL, beta = NULL, gamma = NULL, phi = NULL, additive.only =  
FALSE, lambda = NULL, biasadj = FALSE, opt.crit = c("lik", "amse", "mse", "sigma",  
"mae"), ic = c("aicc", "aic", "bic")), s.args = list(robust = TRUE), weights = c("equal",  
"insample.errors", "cv.errors"), errorMethod = c("RMSE", "MAE", "MASE"),  
cvHorizon = frequency(serie))
```

### **ANEXO III**

Tabela A.III. 1 Parâmetros estimados dos modelos SARIMA

Mês	Modelo	Descrição	Parâmetro estimado	Erro padrão
jan/16	SARIMA(1,1,1)(0,0,2)[24]	ar1	-0,2089	0,0457
		ma1	0,4097	0,0426
		sma1	0,1687	0,0103
		sma2	0,0941	0,0094
fev/16	SARIMA(1,1,1)(0,0,2)[24]	ar1	-0,1997	0,0440
		ma1	0,4018	0,4110
		sma1	0,1675	0,1000
		sma2	0,0913	0,0091
mar/16	SARIMA(1,1,1)(0,0,2)[24]	ar1	0,1949	0,0419
		ma1	0,3998	0,0391
		sma1	0,1635	0,0096
		sma2	0,0902	0,0087
abr/16	SARIMA(1,1,1)(0,0,2)[24]	ar1	-0,1946	0,0420
		ma1	0,3907	0,0394
		sma1	0,1590	0,0093
		sma2	0,0900	0,0085
mai/16	SARIMA(1,1,1)(0,0,2)[24]	ar1	0,2131	0,0402
		ma1	0,4067	0,0375
		sma1	0,1612	0,0090
		sma2	0,0920	0,0082
jun/16	SARIMA(1,1,1)(0,0,2)[24]	ar1	-0,2125	0,0382
		ma1	0,4090	0,0356
		sma1	0,1621	0,0088
		sma1	0,0909	0,0080
jul/16	SARIMA(1,1,1)(0,0,2)[24]	ar1	0,2146	0,0373
		ma1	0,4076	0,3480
		sma1	0,1596	0,0085
		sma2	0,0913	0,0078
ago/16	SARIMA(1,1,1)(0,0,2)[24]	ar1	-0,2317	0,0361
		ma1	0,4241	0,0335
		sma1	0,1603	0,0083
		sma2	0,0894	0,0076
set/16	SARIMA(1,1,1)(0,0,2)[24]	ar1	-0,2306	0,0347
		ma1	0,4250	0,0322
		sma1	0,1625	0,0081
		sma2	0,0896	0,0074
out/16	SARIMA(4,1,0)(2,0,0)[24]	ar1	0,1768	0,0079
		ar2	-0,1123	0,0080
		ar3	-0,0300	0,0008
		ar4	-0,0974	0,0079
nov/16	SARIMA(4,1,0)(2,0,0)[24]	sar1	0,1695	0,0079
		sar2	0,1065	0,0079
		ar1	0,1762	0,0077
		ar2	-0,1101	0,0078
dez/16	SARIMA(4,1,0)(2,0,0)[24]	ar3	-0,0342	0,0078
		ar4	-0,0975	0,0077
		sar1	0,1717	0,0077
		sar2	0,1093	0,0077
		ar1	0,1757	0,0075
		ar2	-0,1125	0,0076
		ar3	-0,0359	0,0076
		ar4	-0,0977	0,0075
		sar1	0,1734	0,0075
		sar2	0,1111	0,0075

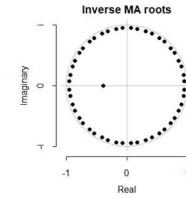
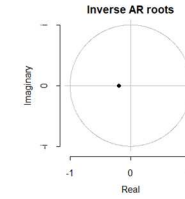
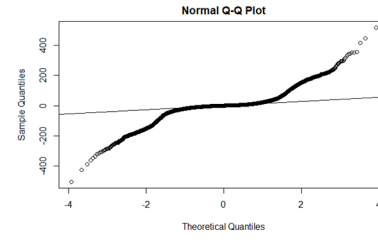
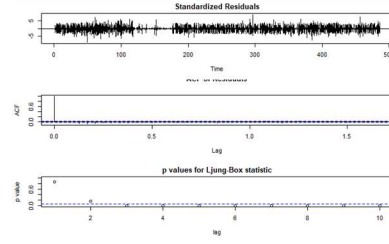
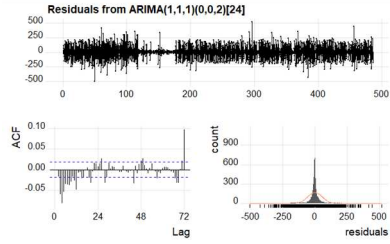
**Tabela A.III. 2 – Diagnósticos dos resíduos, das funções de autocorrelação residual, histogramas dos resíduos, teste de Ljung-Box, gráfico de probabilidade Quantil-Quantil e raízes dos polinômios que garantem a estacionariedade e a invertibilidade do modelo SARIMA**



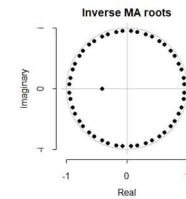
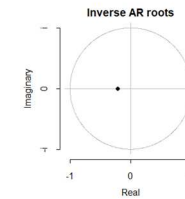
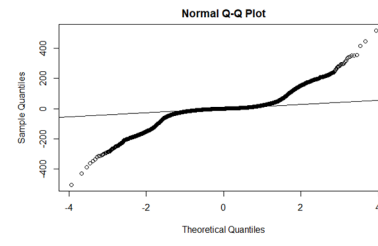
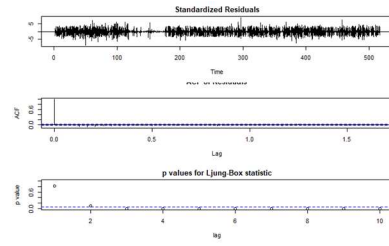
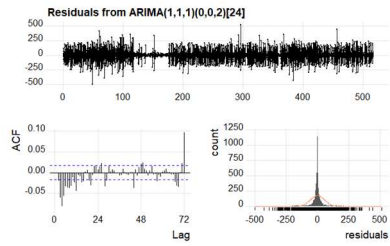
Meses

SARIMA

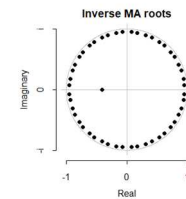
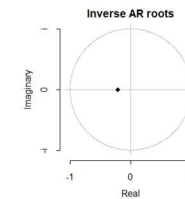
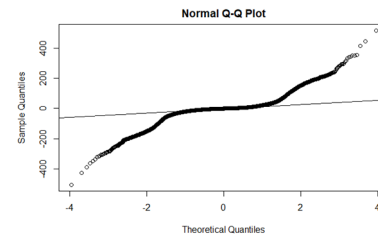
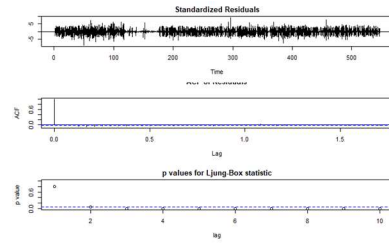
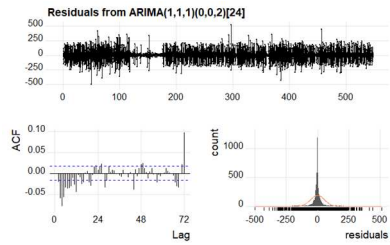
04/2016



05/2016



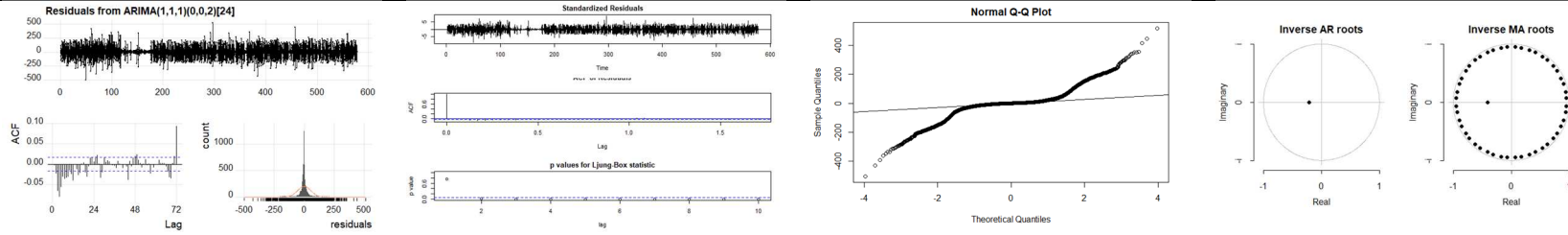
06/2016



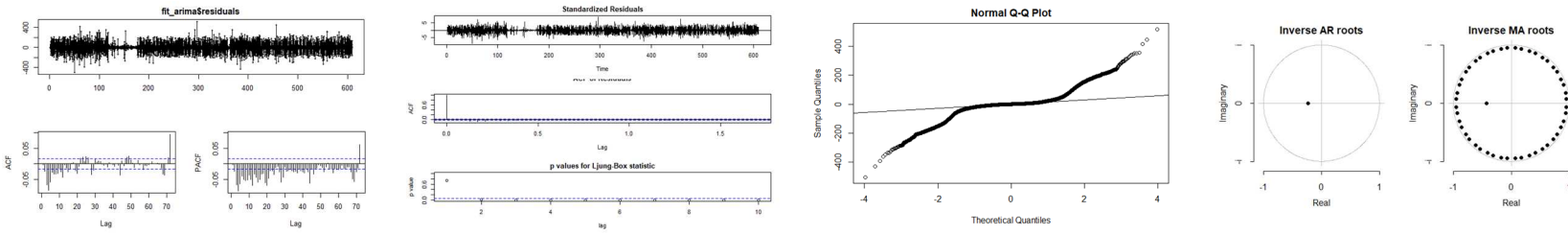
Meses

SARIMA

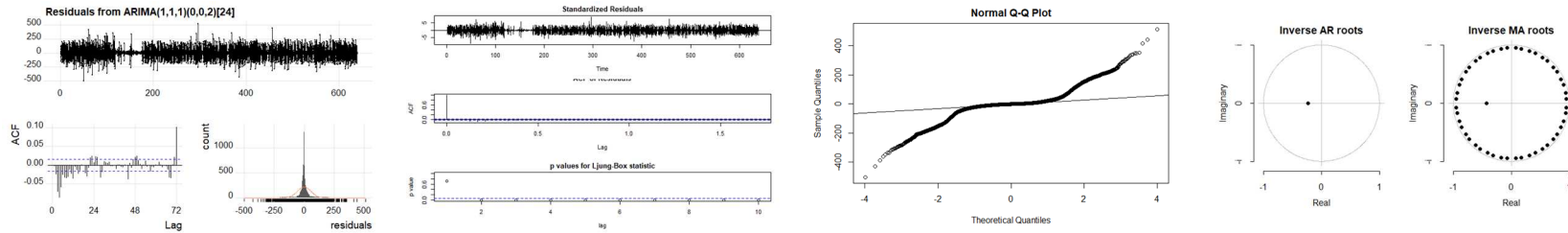
07/2016



08/2016



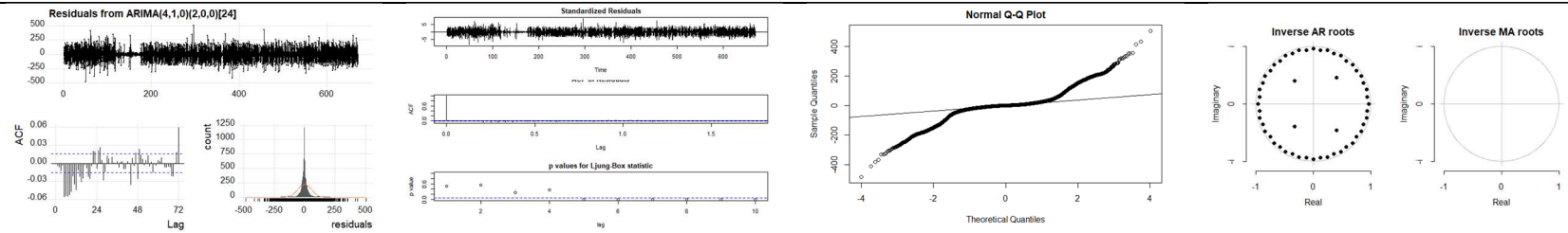
09/2016



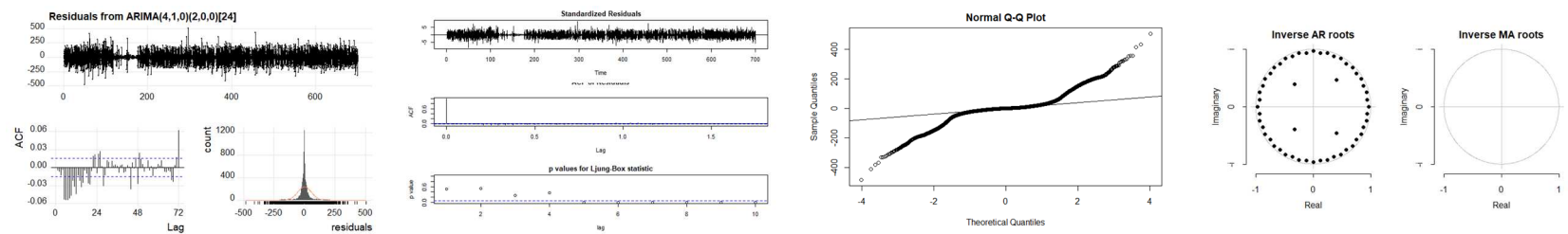
Meses

SARIMA

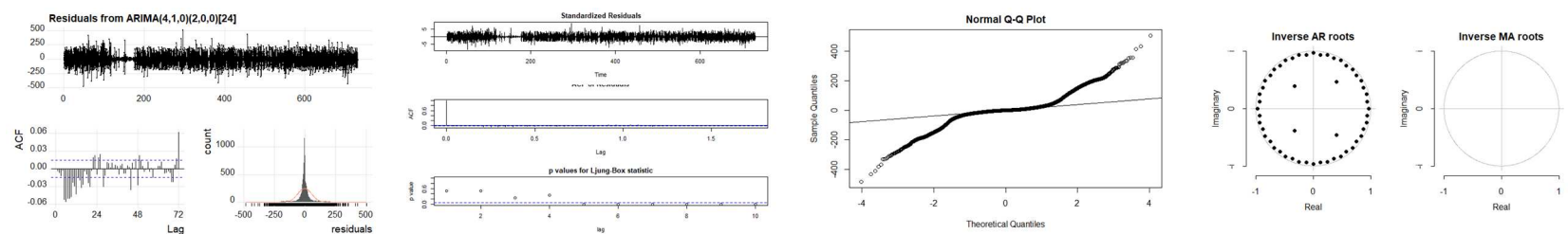
10/2016



11/2016



12/2016



**Tabela A.III.3 –** Parâmetros de suavização e critérios de informação para os modelos estimados

Mês	Modelo	Parâmetros de suavização				sigma	Critérios de informação		
		alpha	beta	gamma	phi		AIC	AICc	BIC
jan/16	ETS(A,Ad,A)*	0,7891	0,0024	0,0037	0,9383	59,0671	164.188,70	164.188,90	164.403,40
fev/16	ETS(A,Ad,A)*	0,9952	0,0002	0,0048	0,9774	56,5692	176.081,00	176.081,20	176.297,80
mar/16	ETS(A,Ad,A)*	0,9081	0,0001	0,0919	0,9789	59,4702	190.814,20	190.814,40	191.033,20
abr/16	ETS(A,Ad,A)*	0,8926	0,0001	0,1017	0,8669	59,7613	204.249,10	204.249,20	204.469,90
mai/16	ETS(A,Ad,A)*	0,7066	0,0032	0,0761	0,9173	62,6022	219.218,70	219.218,90	219.441,50
jun/16	ETS(A,Ad,A)*	0,8942	0,0040	0,1053	0,8892	59,5939	231.410,90	231.411,10	231.635,40
jul/16	ETS(A,Ad,A)*	0,7070	0,0053	0,0810	0,8530	62,3699	246.572,70	246.572,90	246.798,80
ago/16	ETS(A,Ad,A)*	0,7056	0,0053	0,0768	0,8532	62,4914	260.683,00	260.638,10	260.865,60
set/16	ETS(A,Ad,A)*	0,6869	0,0026	0,0711	0,9320	63,0700	274.515,70	274.515,80	274.744,80
out/16	ETS(A,Ad,A)*	0,8936	0,0001	0,1025	0,9194	59,9353	286.976,50	286.976,60	287.207,00
nov/16	ETS(A,Ad,A)*	0,6830	0,0072	0,0923	0,8280	60,5276	300.909,10	300.909,20	301.140,90
dez/16	ETS(A,Ad,A)*	0,9022	0,0019	0,0976	0,8125	59,9845	314.696,40	314.696,50	314.929,50

\* empregou-se transformação Box-Cox

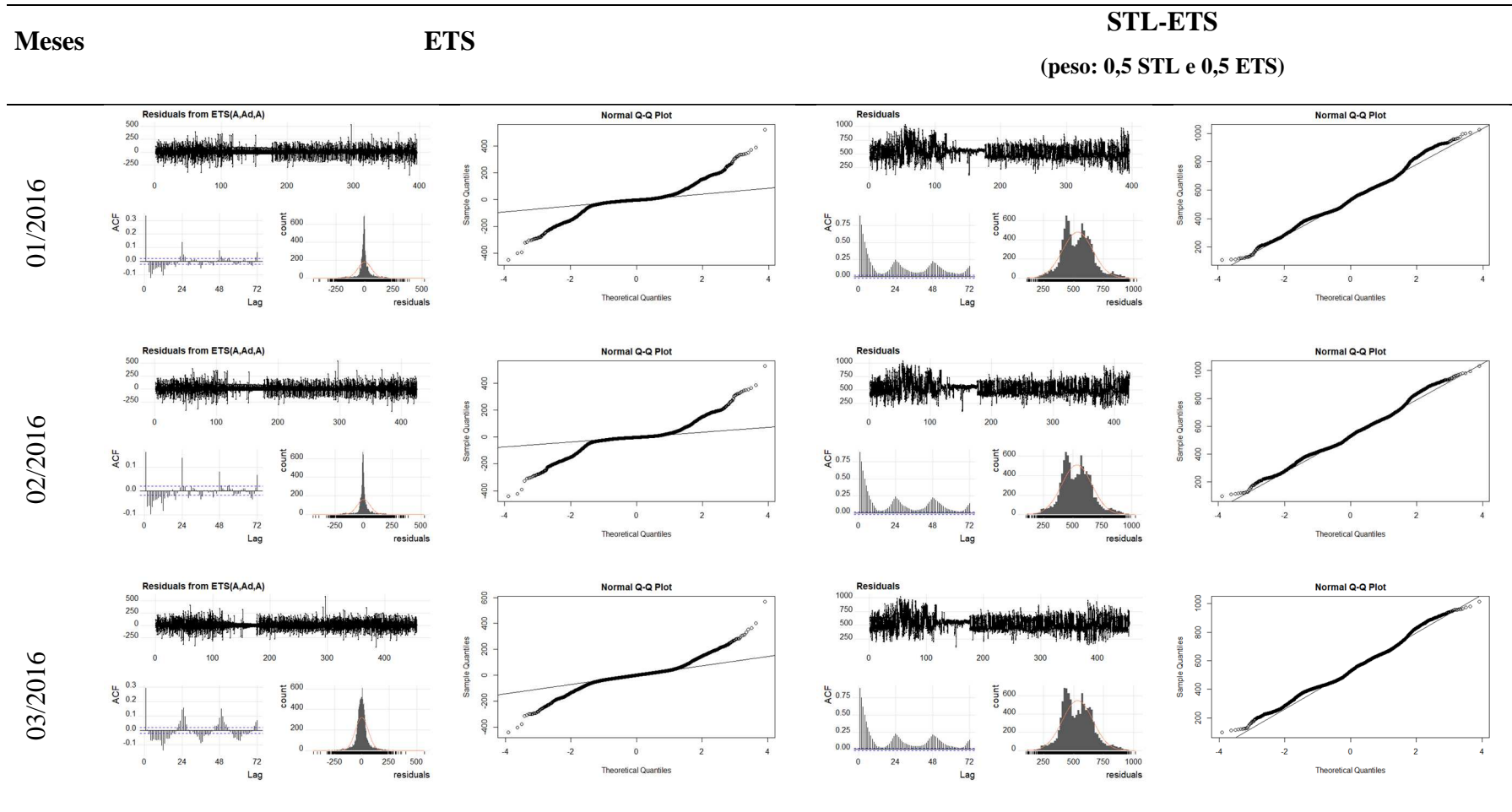
**Tabela A.III.4 –** Teste de normalidade Jarque-Bera (JB) para os modelos ETS, STL-ETS e SARIMA

Mês	Modelo ETS		Modelo STL-ETS		Modelo SARIMA	
	Teste JB	<i>p-value</i>	Teste JB	<i>p-value</i>	Teste JB	<i>p-value</i>
jan/16	X <sup>2</sup> = 21256	< 2.2e-16	X <sup>2</sup> = 254,71	< 2.2e-16	X <sup>2</sup> = 38351	< 2.2e-16
fev/16	X <sup>2</sup> = 31496	< 2.2e-16	X <sup>2</sup> = 196,6	< 2.2e-16	X <sup>2</sup> = 37305	< 2.2e-16
mar/16	X <sup>2</sup> = 15260	< 2.2e-16	X <sup>2</sup> = 208,07	< 2.2e-16	X <sup>2</sup> = 39549	< 2.2e-16
abr/16	X <sup>2</sup> = 16464	< 2.2e-16	X <sup>2</sup> = 222,19	< 2.2e-16	X <sup>2</sup> = 42033	< 2.2e-16
mai/16	X <sup>2</sup> = 12341	< 2.2e-16	X <sup>2</sup> = 241,9	< 2.2e-16	X <sup>2</sup> = 43490	< 2.2e-16
jun/16	X <sup>2</sup> = 17497	< 2.2e-16	X <sup>2</sup> = 252,57	< 2.2e-16	X <sup>2</sup> = 44948	< 2.2e-16
jul/16	X <sup>2</sup> = 13338	< 2.2e-16	X <sup>2</sup> = 236,99	< 2.2e-16	X <sup>2</sup> = 45618	< 2.2e-16
ago/16	X <sup>2</sup> = 13335	< 2.2e-16	X <sup>2</sup> = 205,8	< 2.2e-16	X <sup>2</sup> = 45468	< 2.2e-16
set/16	X <sup>2</sup> = 13001	< 2.2e-16	X <sup>2</sup> = 180,19	< 2.2e-16	X <sup>2</sup> = 44770	< 2.2e-16
out/16	X <sup>2</sup> = 19594	< 2.2e-16	X <sup>2</sup> = 152,91	< 2.2e-16	X <sup>2</sup> = 41030	< 2.2e-16
nov/16	X <sup>2</sup> = 18407	< 2.2e-16	X <sup>2</sup> = 138,33	< 2.2e-16	X <sup>2</sup> = 41290	< 2.2e-16
dez/16	X <sup>2</sup> = 20420	< 2.2e-16	X <sup>2</sup> = 132,09	< 2.2e-16	X <sup>2</sup> = 43614	< 2.2e-16

**Tabela A.III.5** – Teste de autocorrelação Ljung-Box (L-B) dos modelos ETS e SARIMA

Mês	Modelo SARIMA		Modelo ETS	
	Teste L-B	<i>p-value</i>	Teste L-B	<i>p-value</i>
jan/16	Q* = 301,04	< 2.2e-16	Q* = 2089.9	< 2.2e-16
fev/16	Q* = 309,87	< 2.2e-16	Q* = 1032	< 2.2e-16
mar/16	Q* = 324,25	< 2.2e-16	Q* = 2978.2	< 2.2e-16
abr/16	Q* = 343,37	< 2.2e-16	Q* = 2757.6	< 2.2e-16
mai/16	Q* = 367,46	< 2.2e-16	Q* = 5062.6	< 2.2e-16
jun/16	Q* = 388,65	< 2.2e-16	Q* = 3058.2	< 2.2e-16
jul/16	Q* = 416,62	< 2.2e-16	Q* = 5174	< 2.2e-16
ago/16	Q* = 453,02	< 2.2e-16	Q* = 5582.1	< 2.2e-16
set/16	Q* = 483,60	< 2.2e-16	Q* = 6167	< 2.2e-16
out/16	Q* = 452,46	< 2.2e-16	Q* = 3096.8	< 2.2e-16
nov/16	Q* = 470,72	< 2.2e-16	Q* = 4008.9	< 2.2e-16
dez/16	Q* = 487,87	< 2.2e-16	Q* = 3444	< 2.2e-16

**Tabela A.III.6 -** Diagnósticos dos resíduos, das funções de autocorrelação residual, histogramas dos resíduos, gráfico de probabilidade Quantil-Quantil para o modelo ETS e o modelo híbrido STL-ETS.

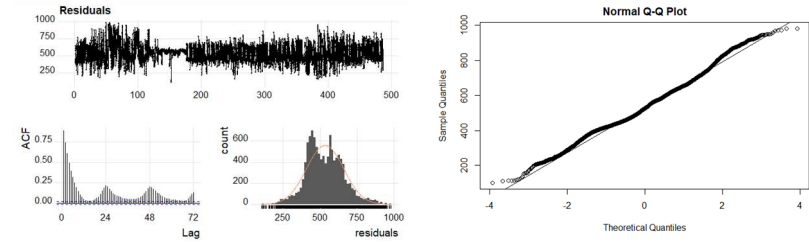
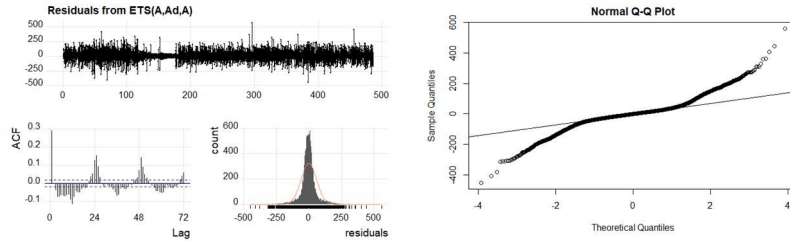


Meses

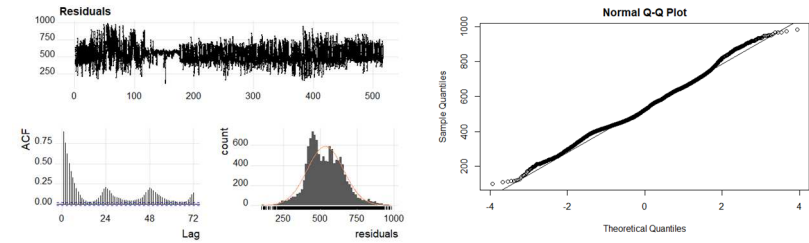
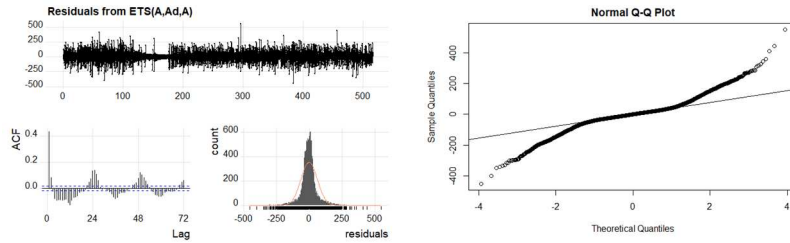
ETS

STL-ETS  
(peso: 0,5 STL e 0,5 ETS)

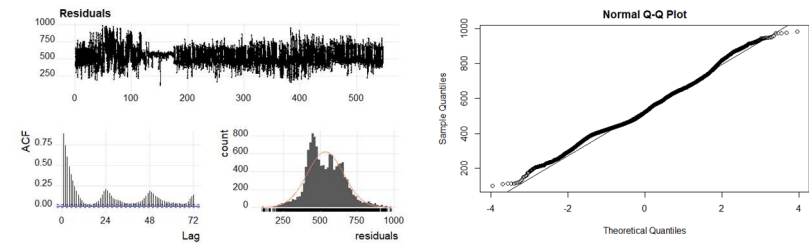
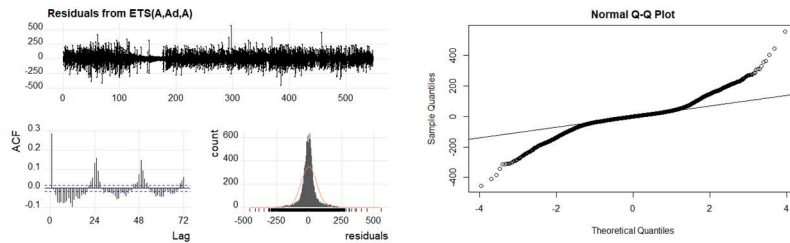
04/2016



05/2016



06/2016

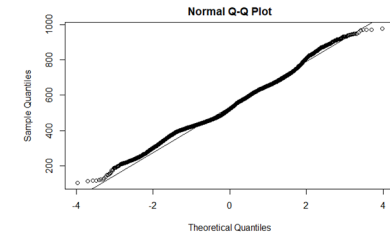
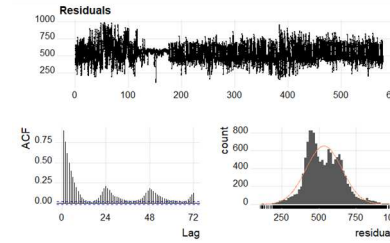
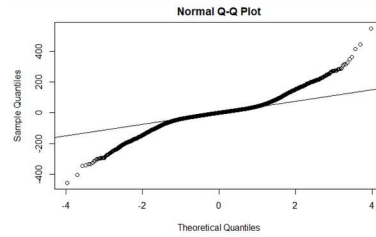
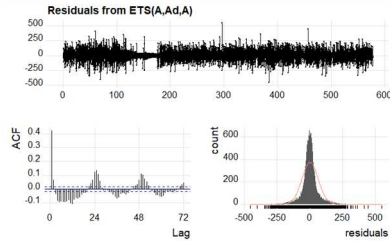


Meses

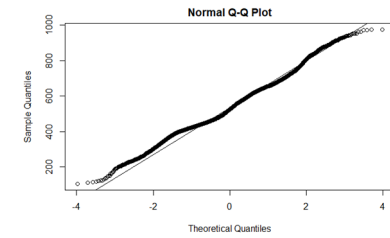
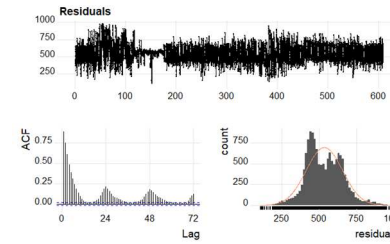
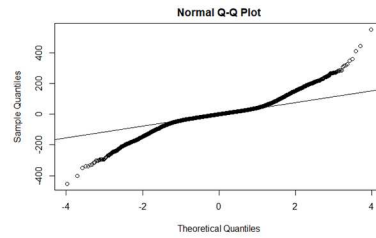
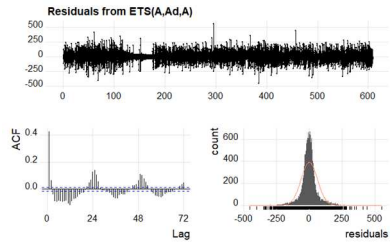
ETS

STL-ETS  
(peso: 0,5 STL e 0,5 ETS)

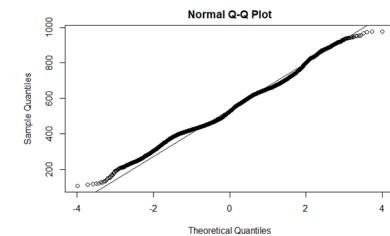
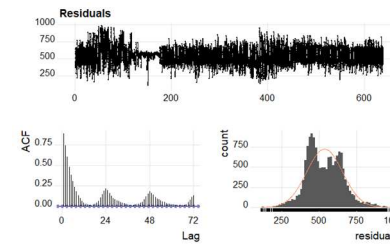
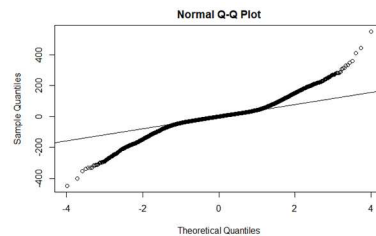
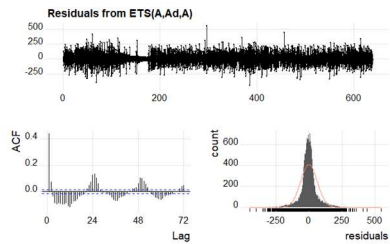
07/2016



08/2016



09/2016

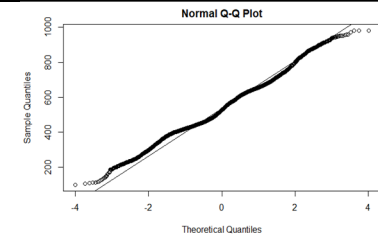
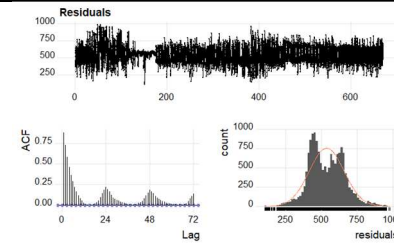
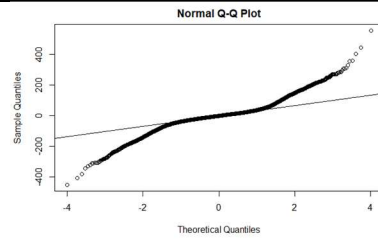
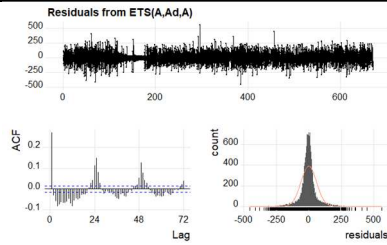


Meses

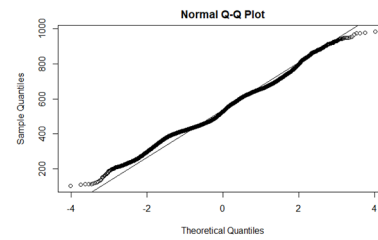
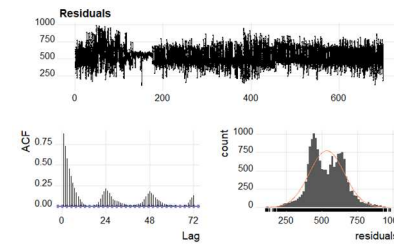
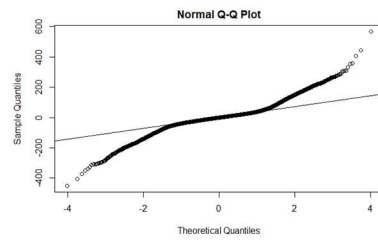
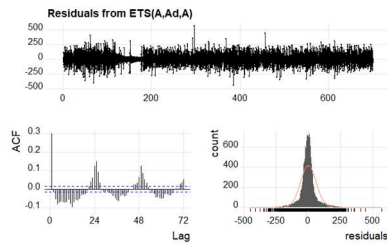
ETS

STL-ETS  
(peso: 0,5 STL e 0,5 ETS)

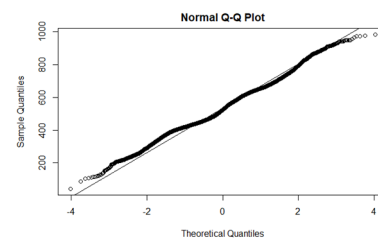
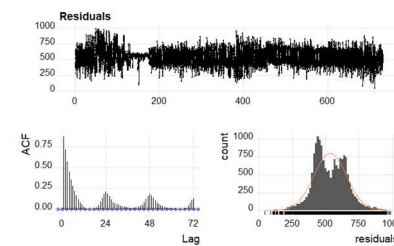
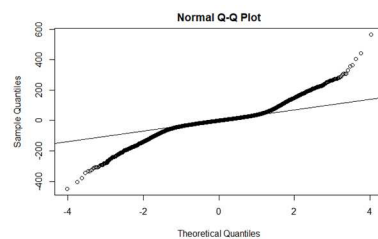
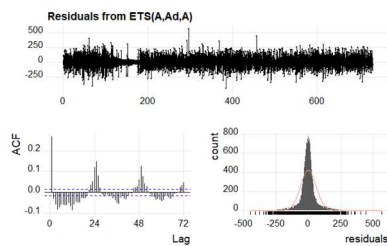
10/2016



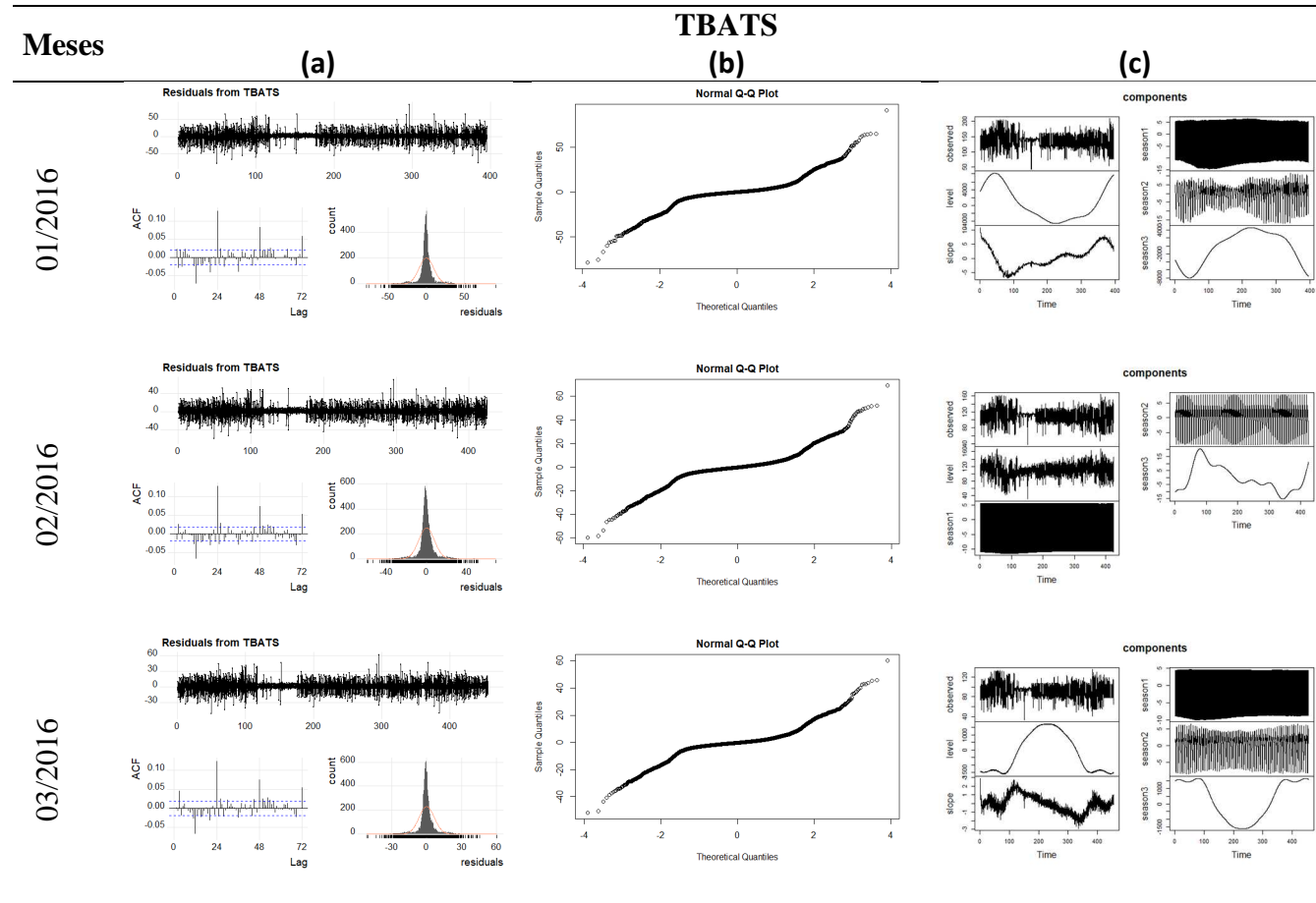
11/2016

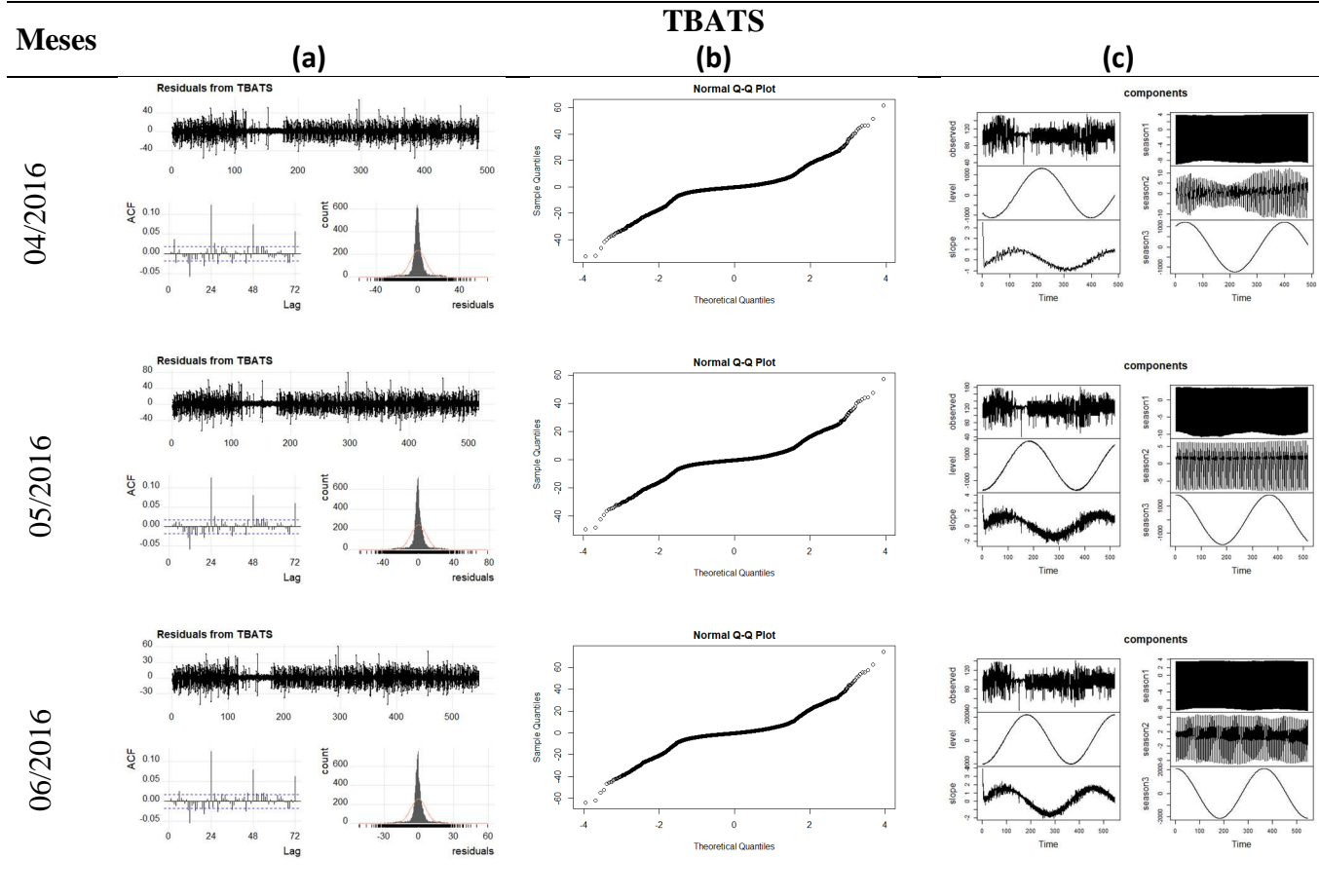


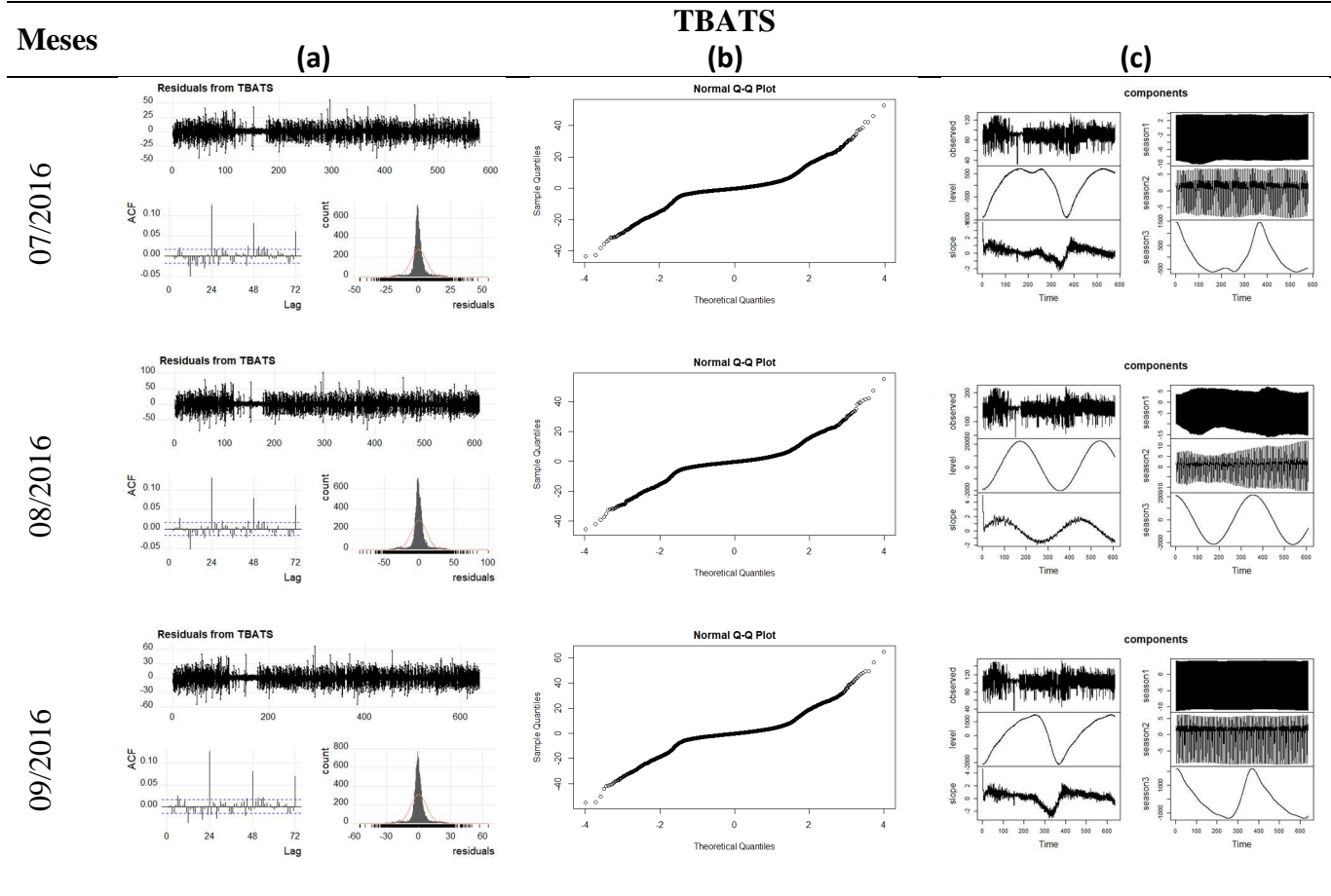
12/2016

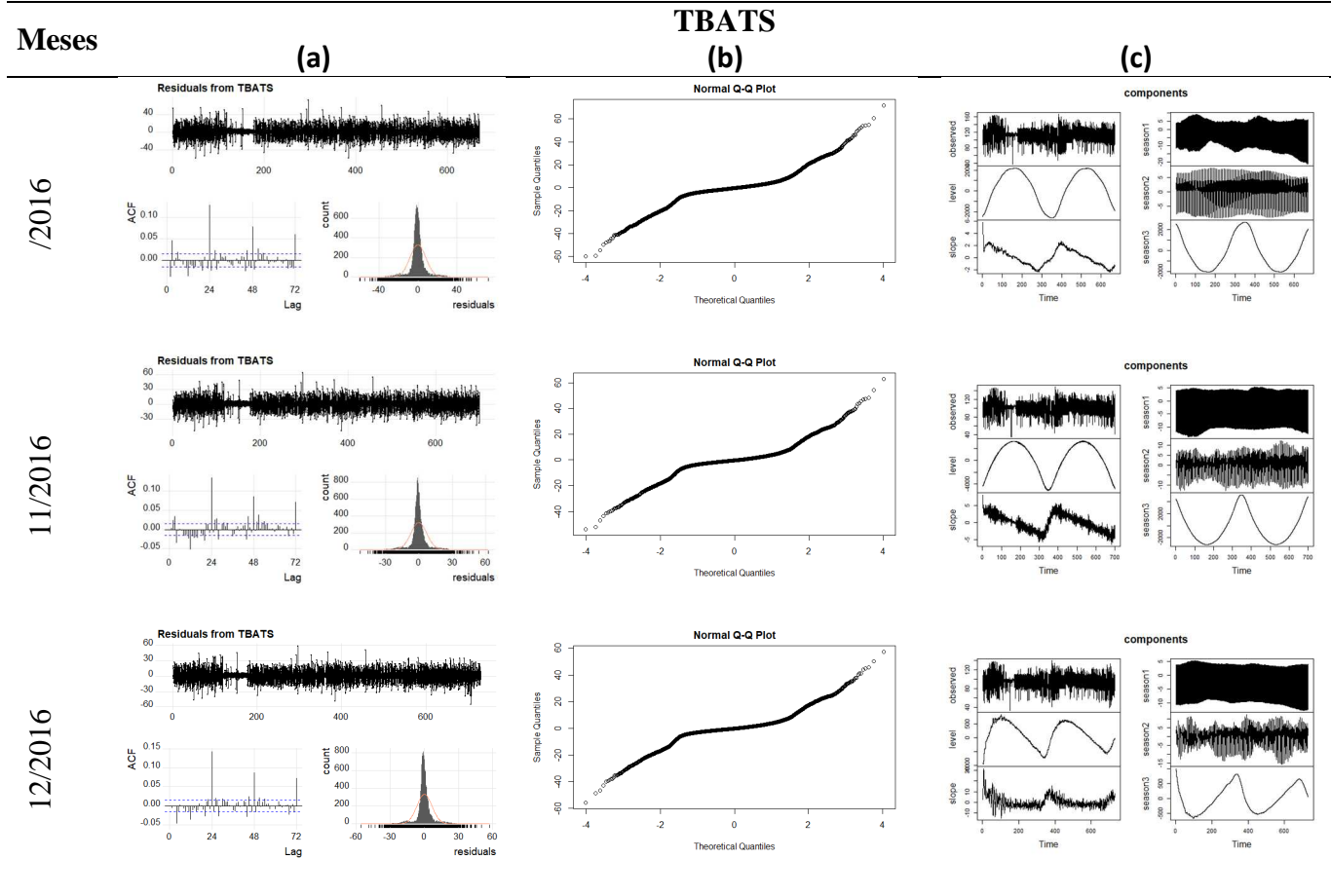


**Tabela A.III.7** – Diagnósticos dos resíduos, funções de autocorrelação residual, histogramas dos resíduos, gráfico de probabilidade Quantil-Quantil e decomposição das múltiplas sazonalidades empregando TBATS.

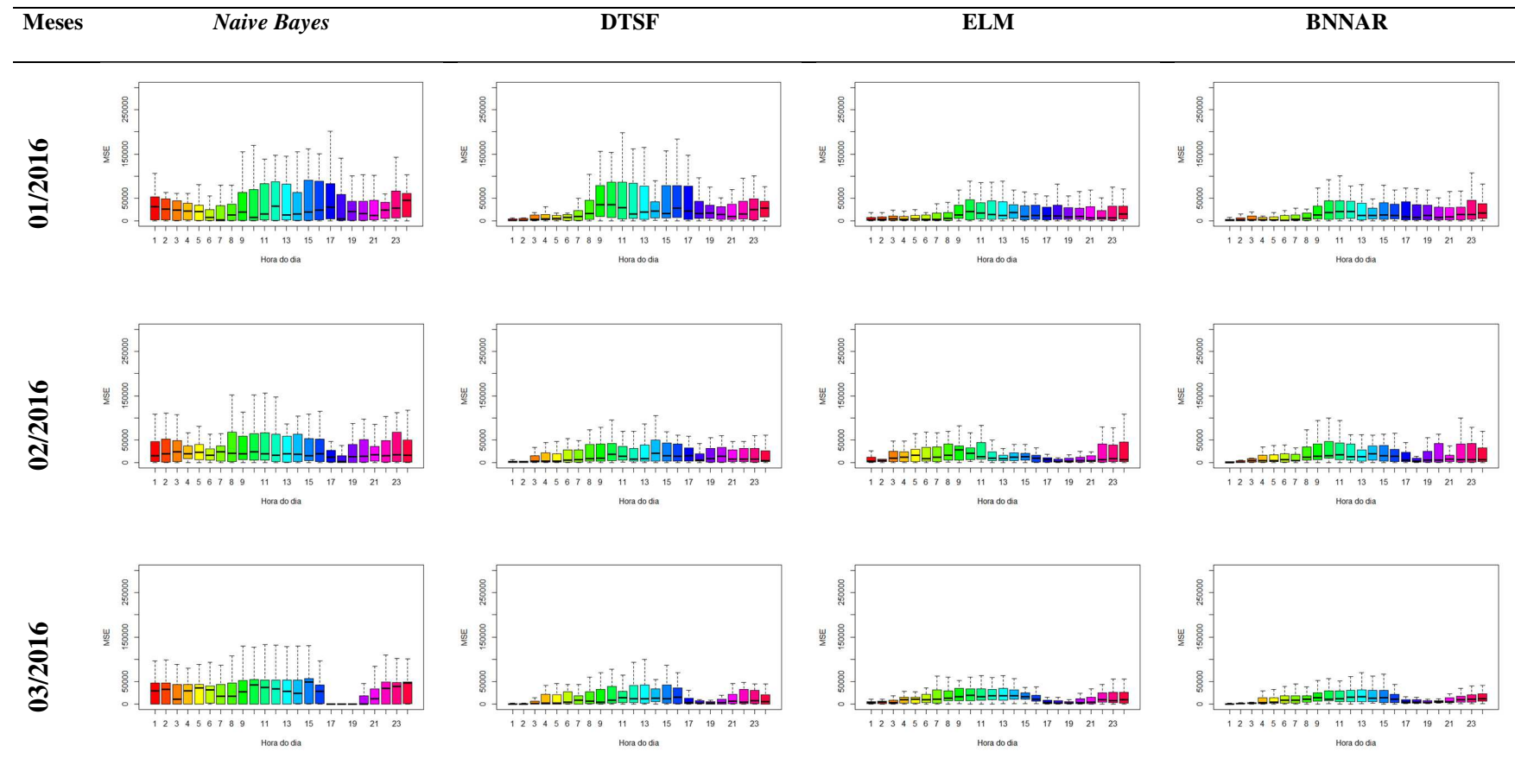






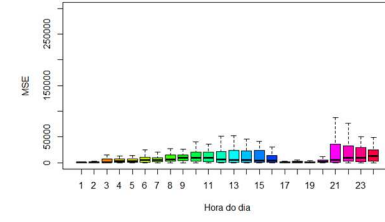
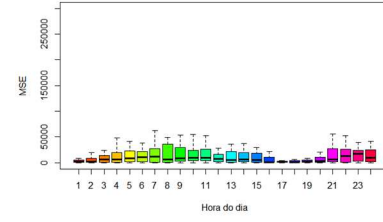
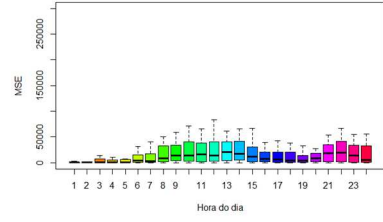
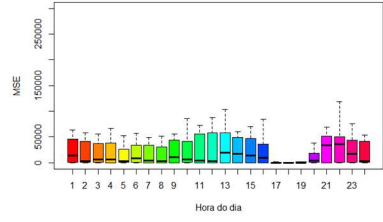


**Tabela A.III.8 - Bloxpot dos erros quadráticos médios (MSE) da matriz de erros das previsões (24 horas para os todos os dias do mês) para os modelos *Naive Bayes*, DTSF, ELM e BNNAR.**

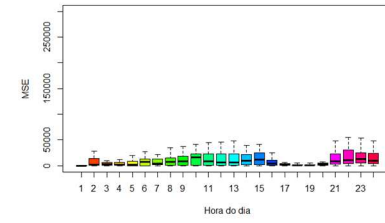
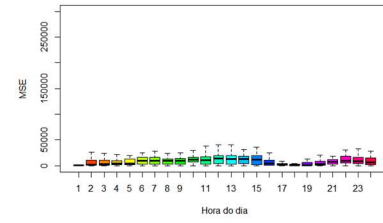
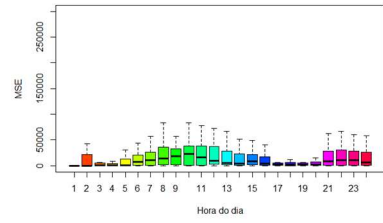
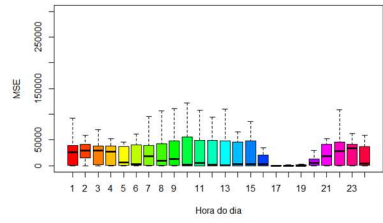


Meses *Naive Bayes* DTSF ELM BNNAR

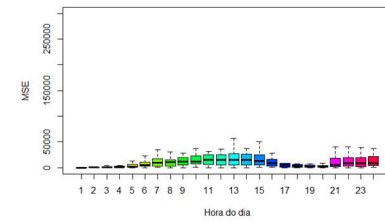
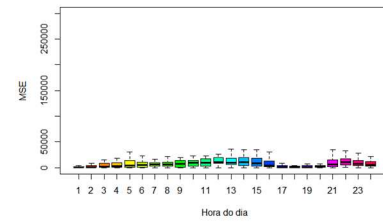
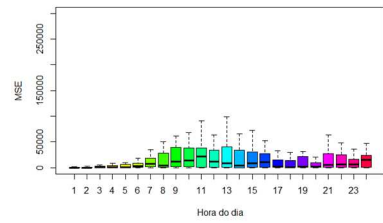
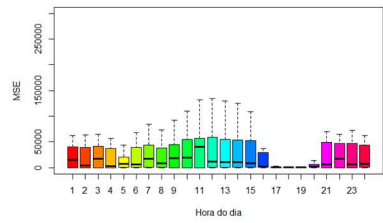
04/2016



05/2016

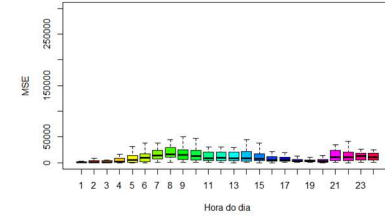
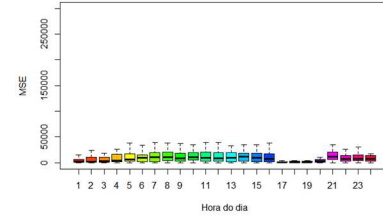
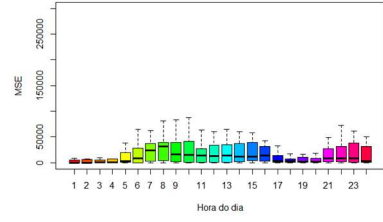
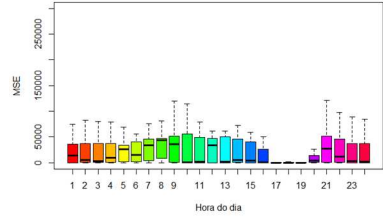


06/2016

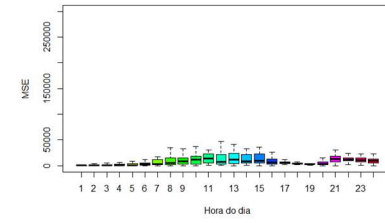
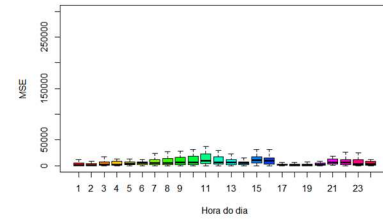
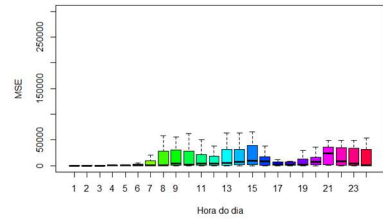
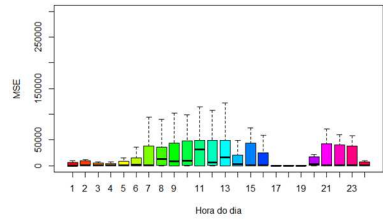


Meses *Naive Bayes* DTSF ELM BNNAR

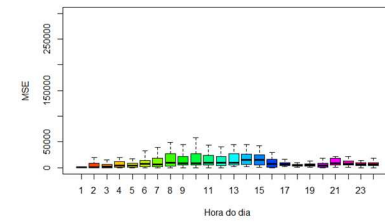
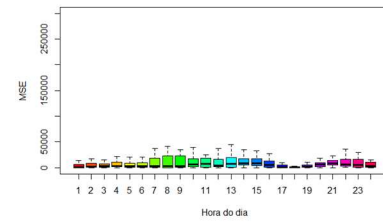
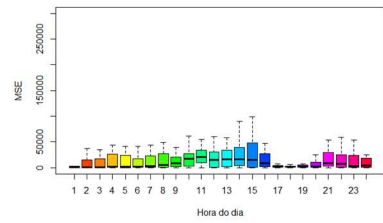
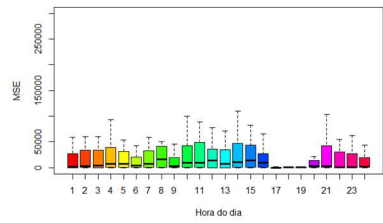
07/2016



08/2016

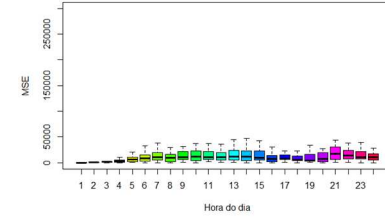
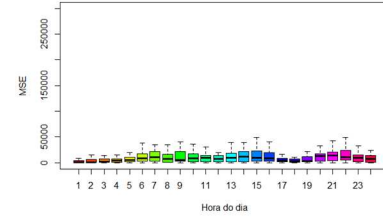
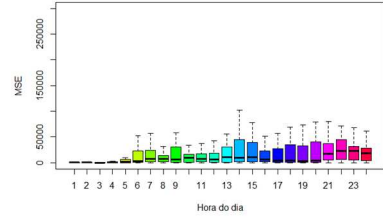
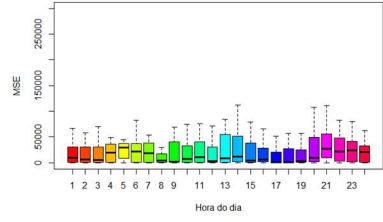


09/2016

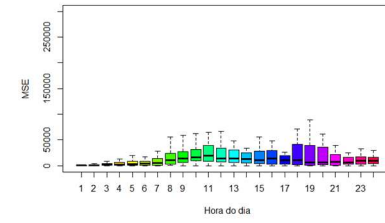
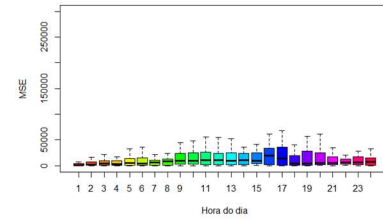
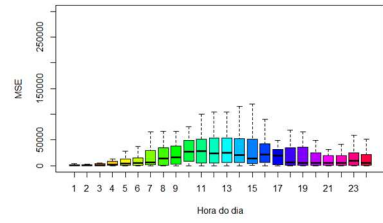
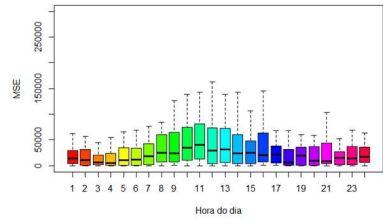


Meses *Naive Bayes* DTSF ELM BNNAR

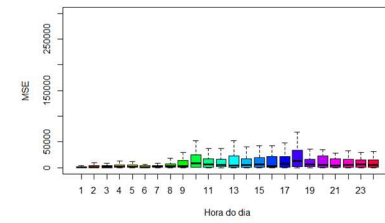
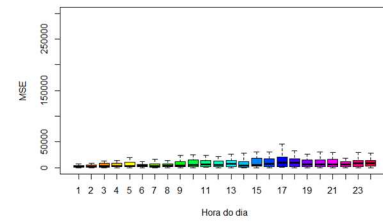
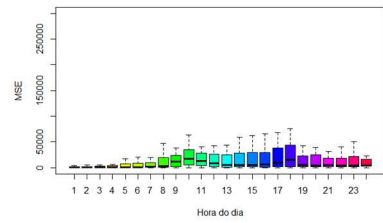
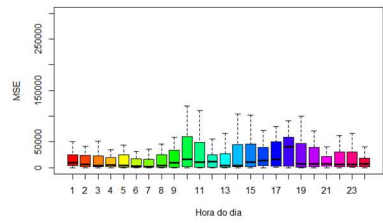
10/2016



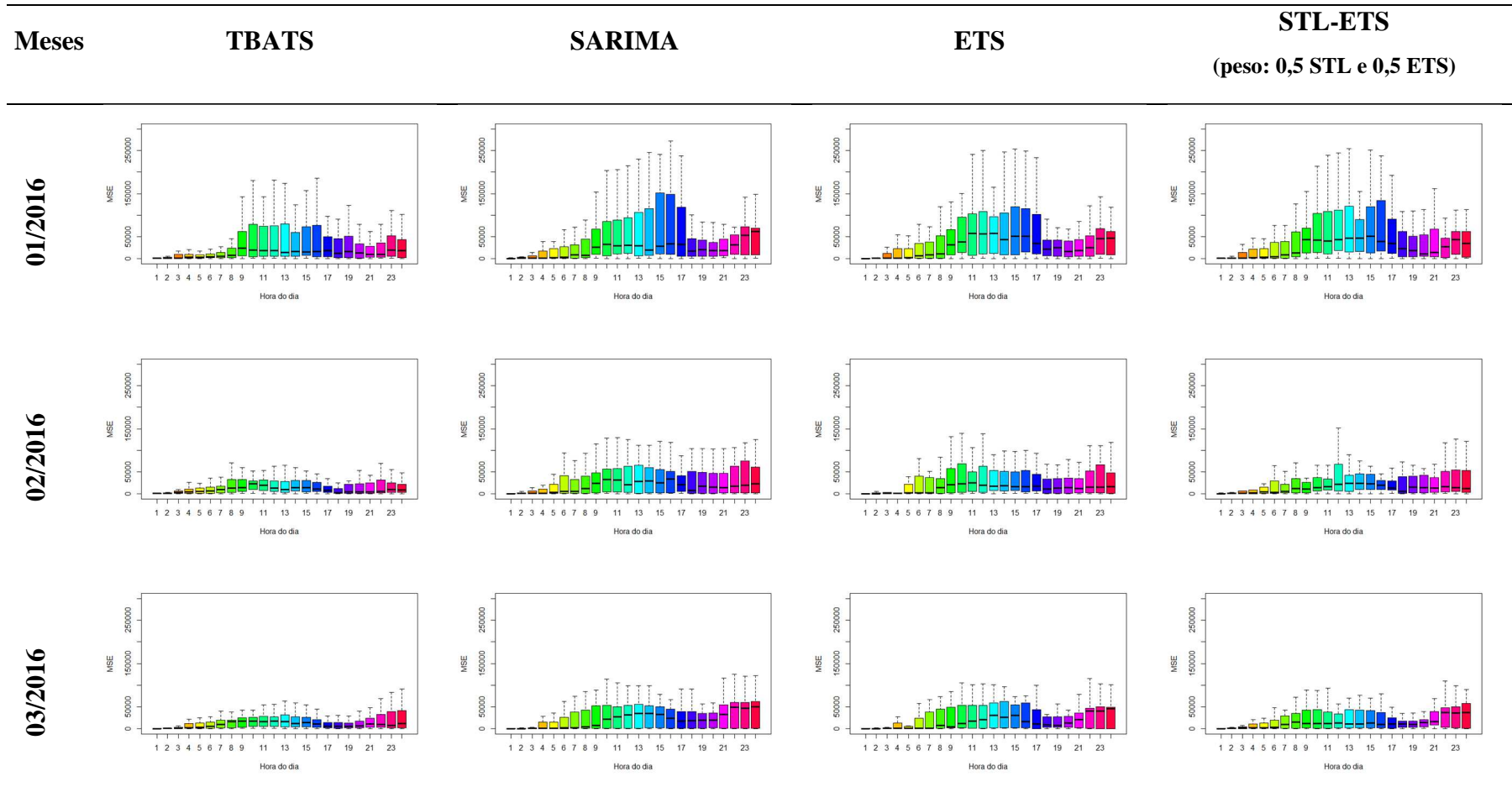
11/2016



12/2016

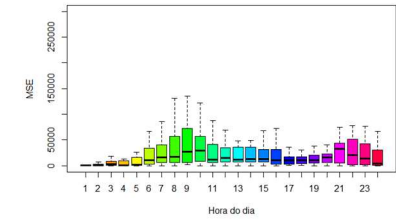
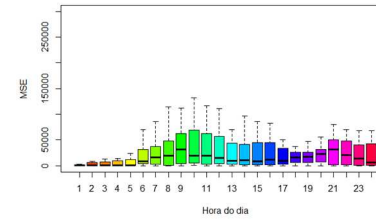
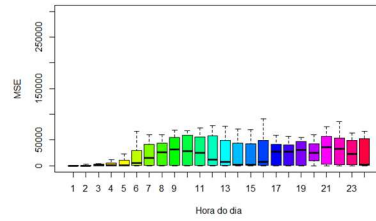
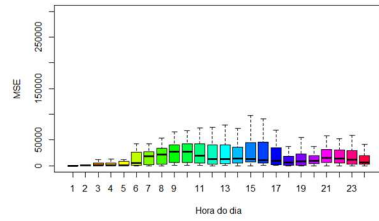


**Tabela A.III.9 - Bloxpot dos erros quadráticos médios (MSE) da matriz de erros das previsões (24 horas para os todos os dias do mês) para os modelos TBATS, SARIMA, ETS e STL-ETS**

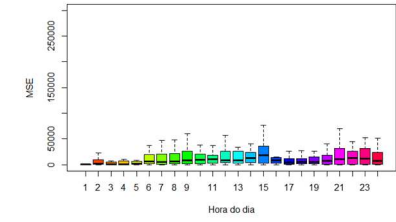
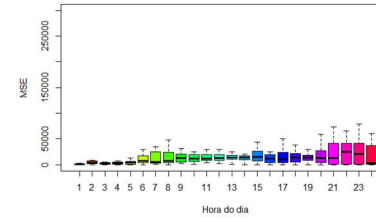
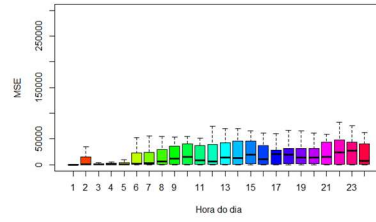
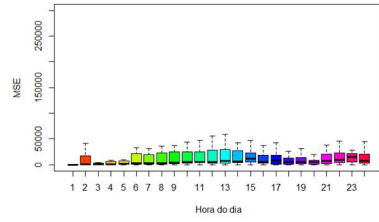


Meses                      **TBATS**                      **SARIMA**                      **ETS**                      **STL-ETS**  
 (peso: 0,5 STL e 0,5 ETS)

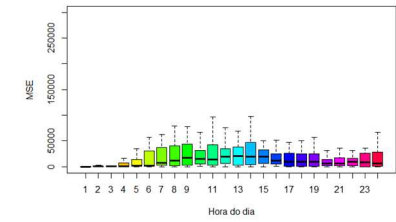
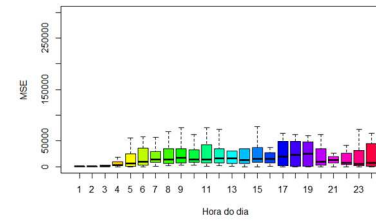
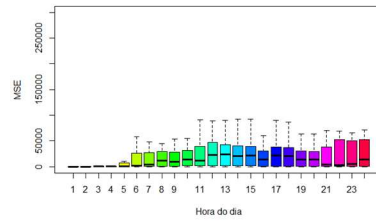
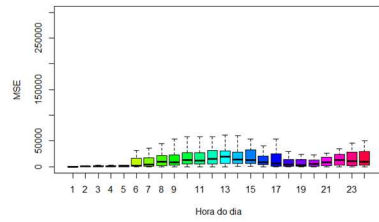
**04/2016**

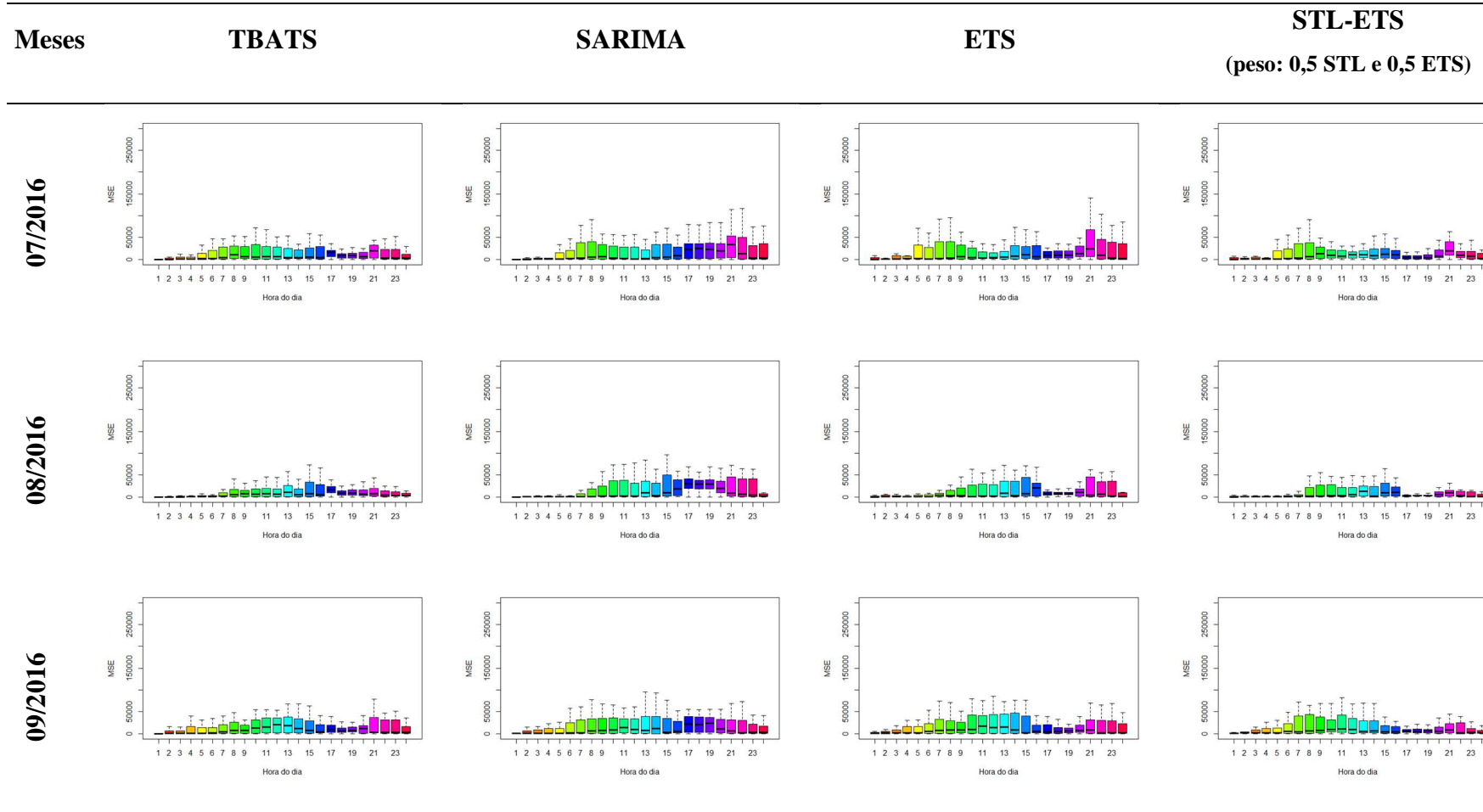


**05/2016**



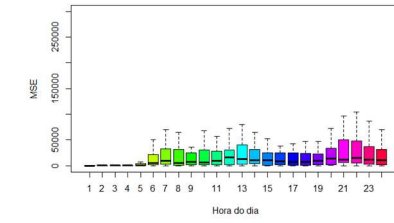
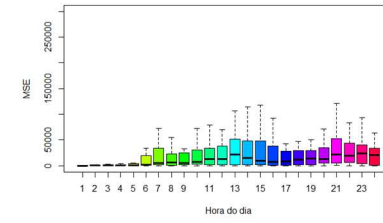
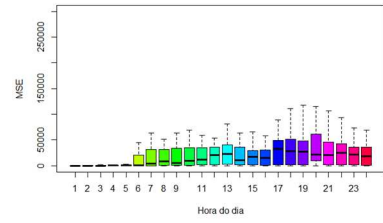
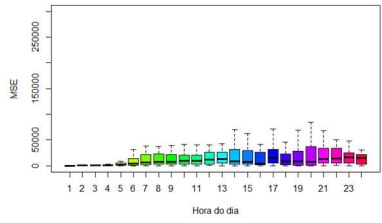
**06/2016**



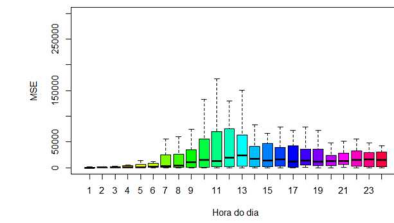
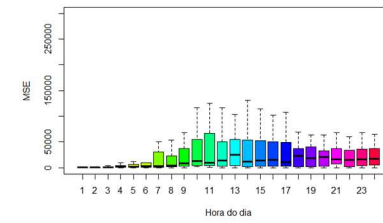
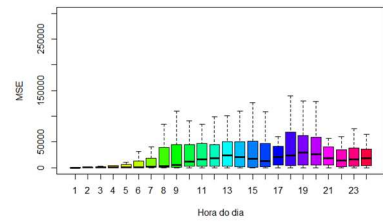
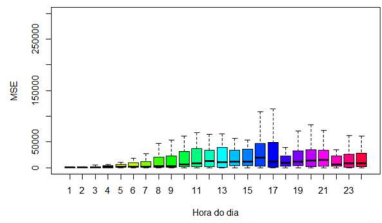


Meses                      **TBATS**                      **SARIMA**                      **ETS**                      **STL-ETS**  
 (peso: 0,5 STL e 0,5 ETS)

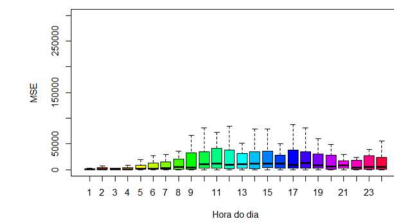
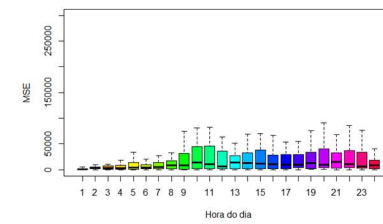
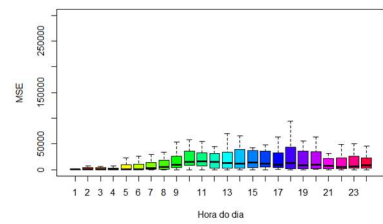
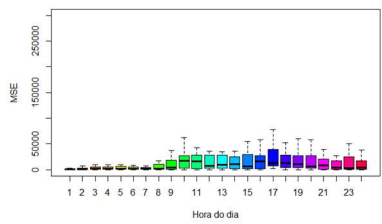
**10/2016**

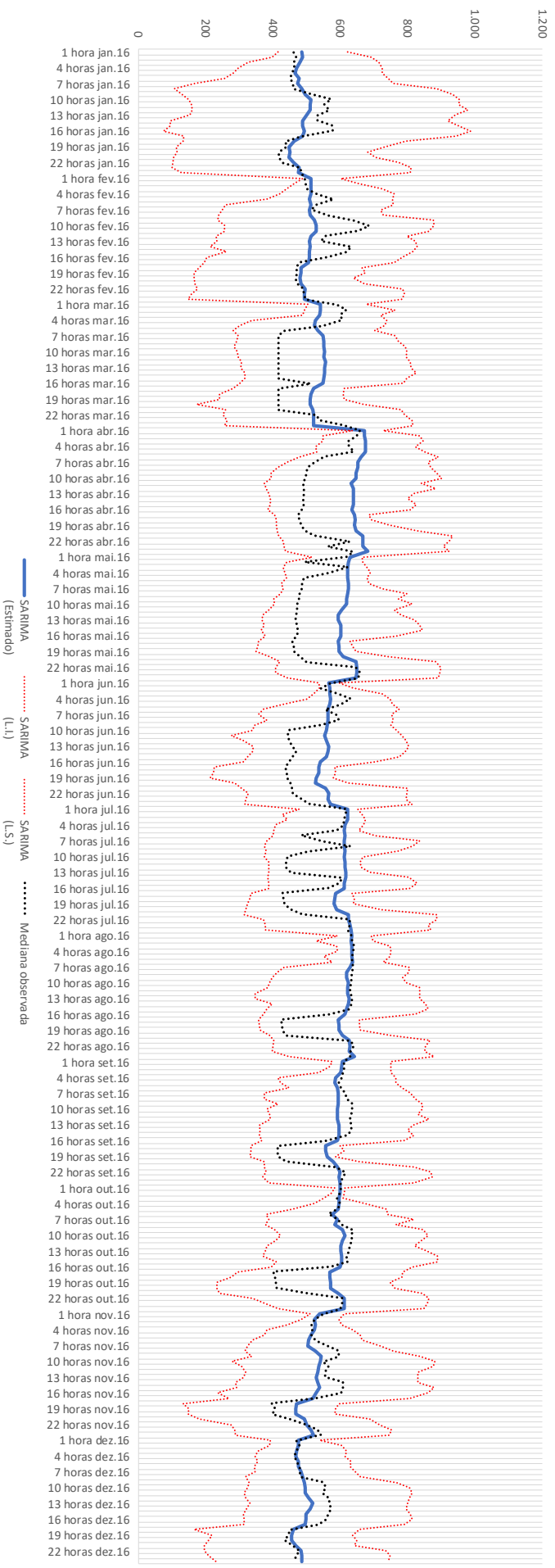


**11/2016**

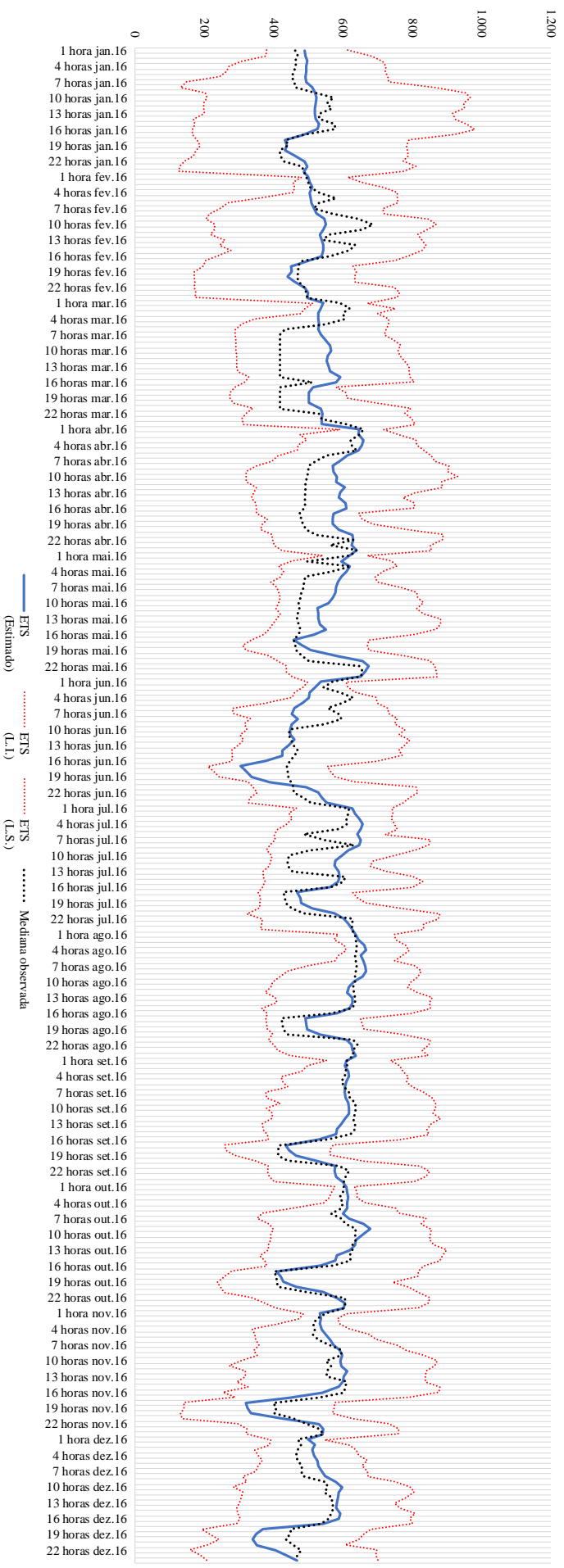


**12/2016**

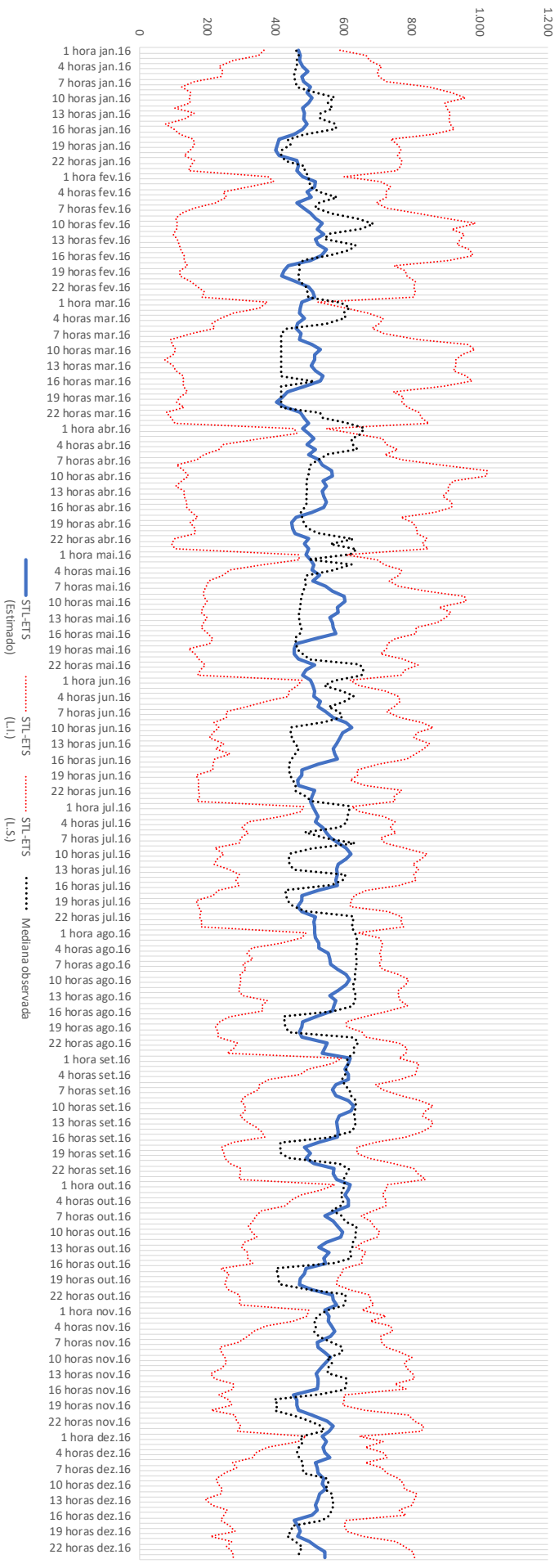




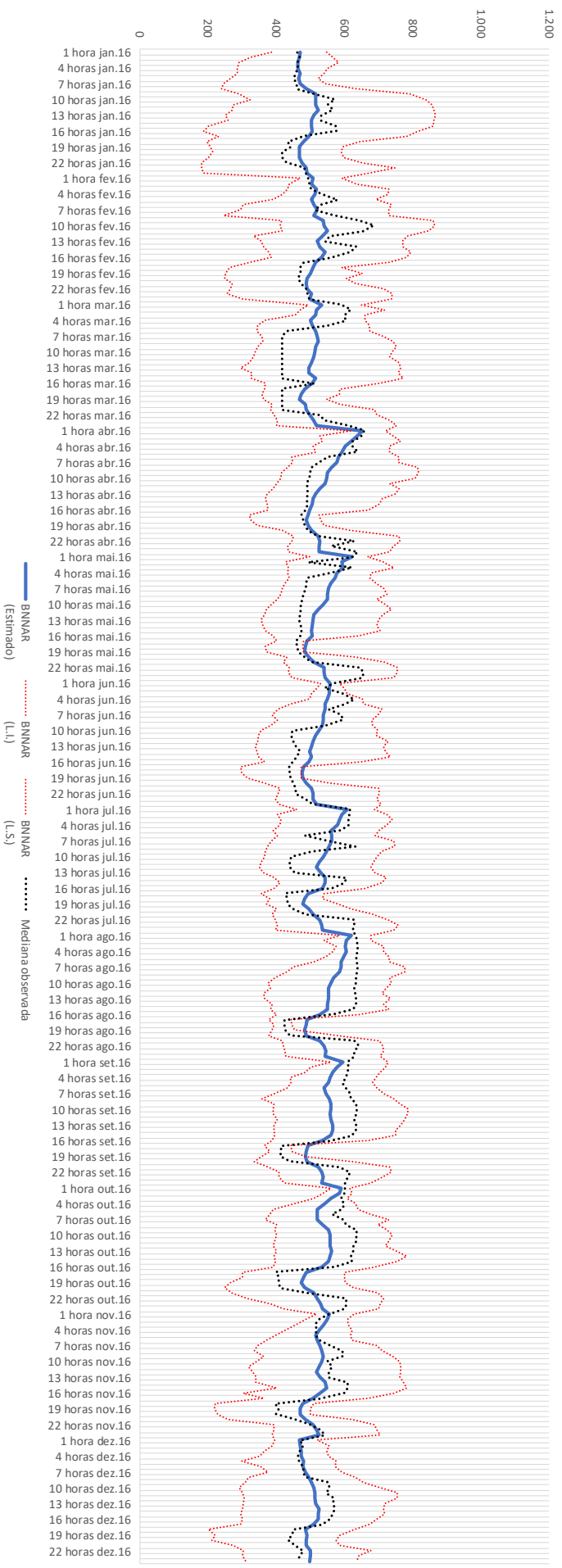
**Figura A.III.1 – Mediana observada e mediana horária por mês da previsão de demanda empregando o modelo SARIMA e o seu respectivo intervalo de confiança empírico.**



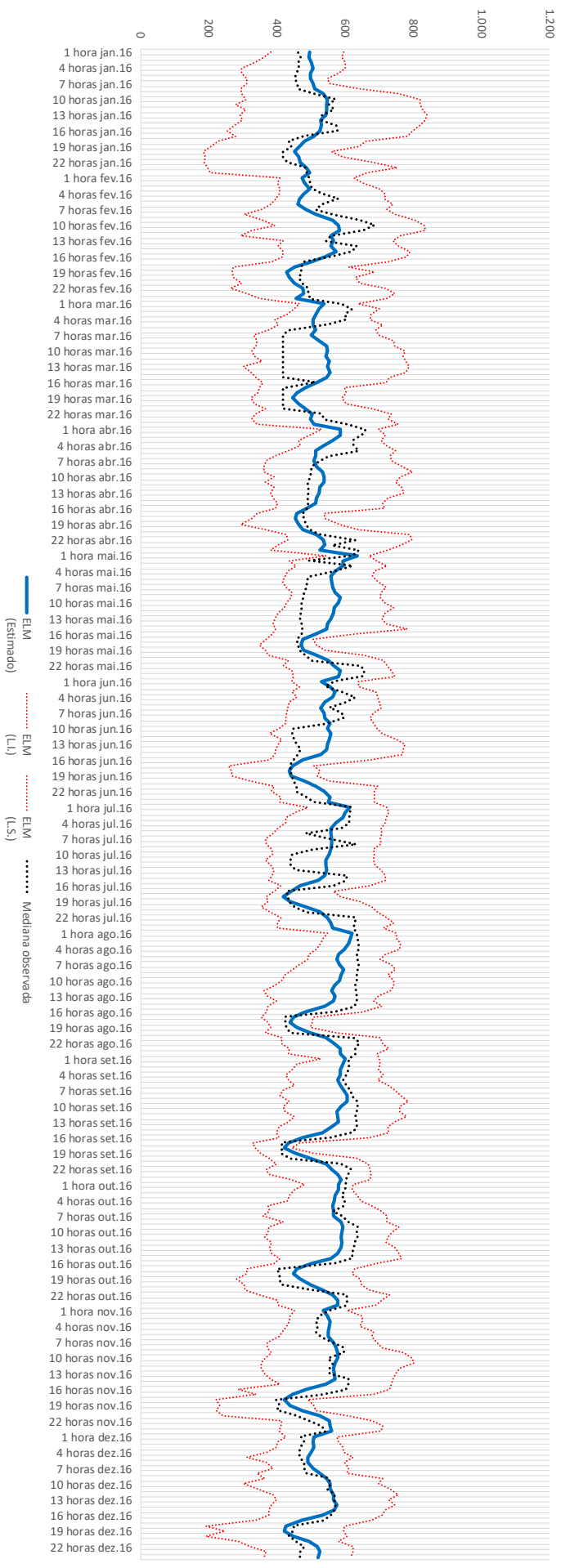
**Figura A.III.2 – Mediana observada e mediana horária por mês da previsão de demanda empregando o modelo ETS e o seu respectivo intervalo de confiança empírico.**



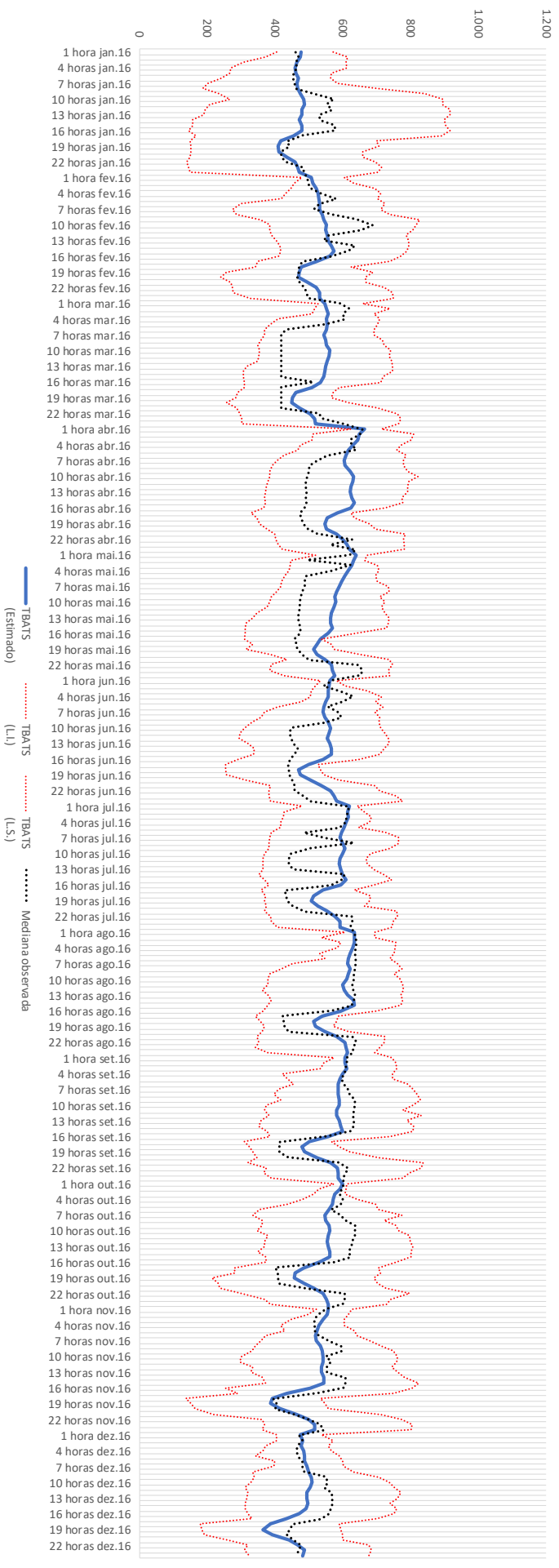
**Figura A.III.3 – Mediana observada e mediana horária por mês da previsão de demanda empregando o modelo STL-ETS e o seu respectivo intervalo de confiança empírico**



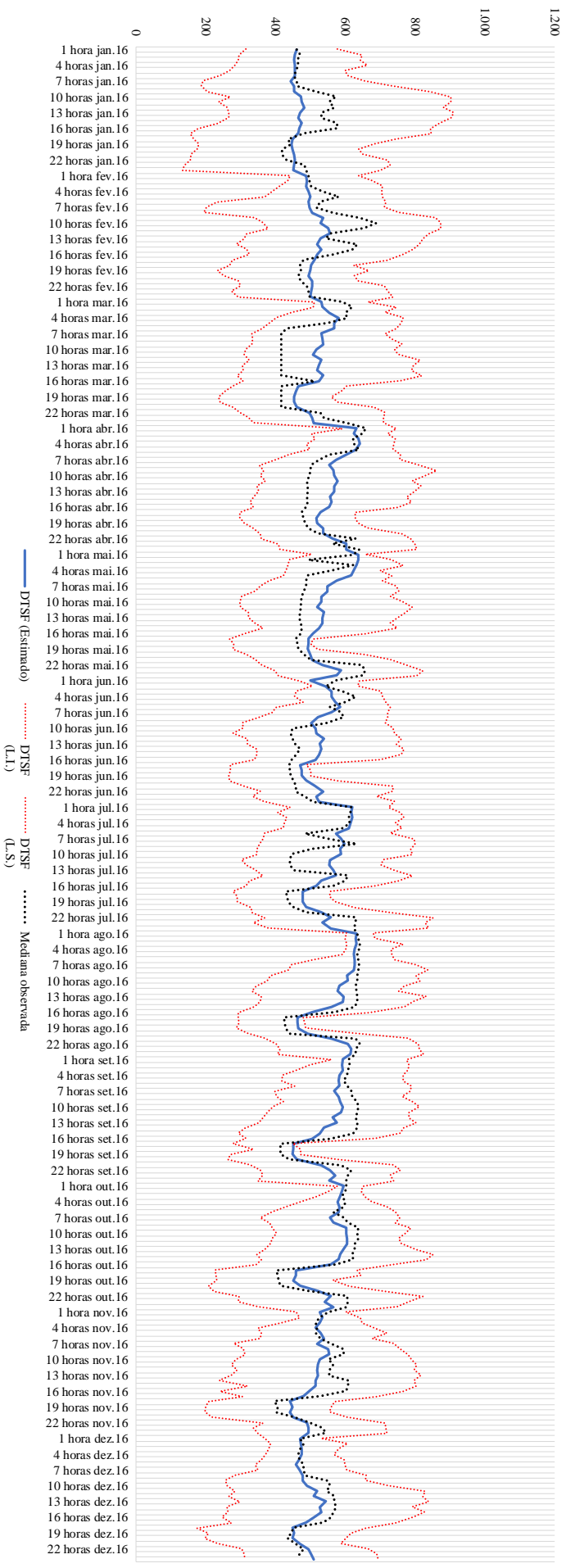
**Figura A.III.4 – Mediana observada e mediana horária por mês da previsão de demanda empregando o modelo BNNAR e o seu respectivo intervalo de confiança empírico.**



**Figura A.III.5 – Mediana observada e mediana horária por mês da previsão de demanda empregando o modelo ELM e o seu respectivo intervalo de confiança empírico.**



**Figura A.III.6 – Mediana observada e mediana horária por mês da previsão de demanda empregando o modelo TBATS e o seu respectivo intervalo de confiança empírico.**



**Figura A.III.7 – Mediana observada e mediana horária por mês da previsão de demanda empregando o modelo DTSF e o seu respectivo intervalo de confiança empírico.**