Pontifícia Universidade Católica
do Rio de Janeiro

Cesar David Revelo Apraez

Abordagem Híbrida Neuro-Evolucionária para Ponderação Dinâmica de Previsores

Dissertação de Mestrado

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-Graduação em Engenharia Elétrica da PUC-Rio.

Orientadora: Profa. Marley Maria Bernardes Rebuzzi Vellasco
Co-orientador: Prof. Luis Martí Orosa

Rio de Janeiro
Novembro de 2016

Pontifícia Universidade Católica
do Rio de Janeiro

Cesar David Revelo Apraz

Abordagem Híbrida Neuro-Evolucionária para Ponderação Dinâmica de Previsores

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-Graduação em Engenharia Elétrica da PUC-Rio. Aprovada pela Comissão Examinadora abaixo assinada.

Profa. Marley Maria Bernardes Rebuzzi Vellasco
Orientadora
Departamento de Engenharia Elétrica – PUC-Rio

Prof. Luis Martí Orosa
Co-Orientador
UFF

Prof. Helio José Correa Barbosa
LNCC

Profa. Karla Tereza Figueiredo Leite
UERJ

Prof. Márcio da Silveira Carvalho
Coordenador Setorial do Centro
Técnico Científico

Rio de Janeiro, 7 de novembro de 2016

Cesar David Revelo Apraez

Graduou-se em Engenharia Eletrônica pela Universidade de Nariño (Pasto, Colômbia) em 2012, obtendo menção meritória no desenvolvimento do trabalho final de curso. Suas principais áreas de pesquisa incluem: técnicas de agrupamento e classificação, aprendizagem supervisionada e não supervisionada, agregação de classificadores e ensembles, meta-heurísticas para problemas de otimização multiobjetivo, tratamento de dados com técnicas de data mining, métodos Bayesianos e modelos de inferência e classificação baseados em redes neurais.

Ficha catalográfica

Abordagem Híbrida Neuro-Evolucionária para Ponderação Dinâmica de Previsores / Cesar David Revelo Apraez; orientadora: Marley Maria Bernardes Rebuzzi Vellasco; co-orientador: Luis Martí Orosa. – 2016.

146 f.: il. color.; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Elétrica.

Inclui bibliografia.

1. Engenharia Elétrica – Tese. 2. Sistemas Neuro-Evolucionários; 3. Otimização Multiobjetivo. 4. Combinação de Previsores/Previsões. 5. Redes Neurais Artificiais. 6. Séries Temporais. I. Rebuzzi Vellasco, Marley Maria Bernardes. II. Martí Orosa, Luis. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Elétrica. IV. Título.

CDD: 621.3

Agradecimentos

Aos meus orientadores Professores Marley Vellasco e Luis Martí pelo apoio, simpatia de sempre, e incentivo para a realização deste trabalho, sem o seu suporte e contribuição, ele não aconteceria.

À CAPES e à PUC-Rio, pelos auxílios concedidos, sem os quais este trabalho não poderia ter sido realizado.

Agradeço aos meus pais, Gladys e Ignacio pelo apoio incondicional em todos os momentos da vida, construindo a base que alicerçou de forma sólida minha estrutura pessoal, permitindo que nos momentos de dúvida me mantivesse com foco no propósito de elaboração desta dissertação.

No decorrer desta dissertação tive a oportunidade de compartilhar de momentos, que mais do que crescimento acadêmico e profissional, me proporcionaram crescimento pessoal. Meus amigos José, Oscar, Jorge e Daniel pelos momentos de discussões filosóficas, diversão e apoio.

Por último, e mais importante, agradeço a Deus, pela possibilidade de poder realizar agradecimentos, um sinal de que Ele colocou no meu caminho pessoas que me possibilitaram crescimento, conhecimento, força e divertimento, agregando a esta formação muito mais do que uma titulação.

Resumo

Revelo Apraez, Cesar David; Vellasco, Marley Maria Bernardes Rebuzzi; Orosa, Luis Martí. **Abordagem Híbrida Neuro-Evolucionária para Ponderação Dinâmica de Previsores**. Rio de Janeiro, 2016. 146p. Dissertação de Mestrado – Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

Estudos empíricos na área de séries temporais indicam que combinar modelos preditivos, originados a partir de diferentes técnicas de modelagem, levam a previsões consensuais superiores, em termos de acurácia, às previsões individuais dos modelos envolvidos na combinação. No presente trabalho é apresentada uma metodologia de combinação convexa de modelos estatísticos de previsão, cujo sucesso depende da forma como os pesos de combinação de cada modelo são estimados. Uma Rede Neural Artificial Perceptron Multi-camada (Multilayer Perceptron - MLP) é utilizada para gerar dinamicamente vetores de pesos ao longo do horizonte de previsão, sendo estes dependentes da contribuição individual de cada previsor observada nos dados históricos da série. O ajuste dos parâmetros da rede MLP é efetuado através de um algoritmo de treinamento híbrido, que integra técnicas de busca global, baseadas em computação evolucionária, junto com o algoritmo de busca local backpropagation, de modo a otimizar de forma simultânea tanto os pesos quanto a arquitetura da rede, visando, assim, a gerar de forma automática um modelo de ponderação dinâmica de previsores de alto desempenho. O modelo proposto, batizado de Neural Expert Weighting - Genetic Algorithm (NEW-GA), foi avaliado em diversos experimentos comparativos com outros modelos de ponderação de previsores, assim como também com os modelos individuais envolvidos na combinação, contemplando 15 séries temporais divididas em dois estudos de casos: séries de derivados de petróleo e séries da versão reduzida da competição NN3, uma competição entre metodologias de previsão, com maior ênfase nos modelos baseados em Redes Neurais. Os resultados demonstraram o potencial do NEW-GA em fornecer modelos acurados de previsão de séries temporais.

Palavras-chave

Sistemas Neuro-Evolucionários; Otimização Multiobjetivo; Combinação de Previsores/Previsões; Redes Neurais Artificiais; Séries Temporais; Previsão Múltiplos Passos a Frente.

Abstract

Revelo Apraez, Cesar David; Vellasco, Marley Maria Bernardes Rebuzzi (Advisor); Martí Orosa, Luis (Co-advisor). **A Hybrid Neuro-Evolutionary Approach for Dynamic Weighted Aggregation of Time Series Forecasters**. Rio de Janeiro, 2016. 146p. MSc. Dissertation – Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

Empirical studies on time series indicate that the combination of forecasting models, generated from different modeling techniques, leads to higher consen+sus forecasts, in terms of accuracy, than the forecasts of individual models involved in the combination scheme. In this work, we present a methodology for convex combination of statistical forecasting models, whose success depends on how the combination weights of each model are estimated. An Artificial Neural Network Multilayer Perceptron (MLP) is used to generate dynamically weighting vectors over the forecast horizon, being dependent on the individual contribution of each forecaster observed over historical data series. The MLP network parameters are adjusted via a hybrid training algorithm that integrates global search techniques, based on evolutionary computation, along with the local search algorithm backpropagation, in order to optimize simultaneously both weights and network architecture. This approach aims to automatically generate a dynamic weighted forecast aggregation model with high performance. The proposed model, called Neural Expert Weighting - Genetic Algorithm (NEW-GA), was com- pared with other forecaster combination models, as well as with the individual models involved in the combination scheme, comprising 15 time series divided into two case studies: Petroleum Products and the reduced set of NN3 forecasting competition, a competition between forecasting methodologies, with greater emphasis on models based on neural networks. The results obtained demonstrated the potential of NEW-GA in providing accurate models for time series forecasting.

Keywords

Neuro-Evolutionary Models; Multiobjective Optimization; Forecast/ Forecasting Combination; Artificial Neural Networks; Time Series; Multistep Ahead Forecasting.

Sumário

Lista de figuras

Lista de tabelas

Lista de símbolos e abreviações

ANP	Agência Nacional de Petróleo, Gás Natural e Biocombustíveis
ARIMA	Autoregressive Integrated Moving Average
BJ	Box & Jenkins
BJ-	Previsor limiar BJ inferior
BJ+	Previsor limiar BJ superior
DIESEL	Vendas de óleo diesel
GASOLINA	Vendas de gasolina convencional
GLP	Vendas de gás liquefeito de petróleo
HW	Holt-Winters
HW-	Previsor limiar HW inferior
HW+	Previsor limiar HW superior
MHG	Metaheurística Genética
MLP	Multilayer Perceptron
MOEA	Multiobjective Genetic Algorithm
MQR	Mínimos Quadrados Restritos
MSE	Mean Squared Error
NEW	Neural Expert Weighting
NN3	Competição NN3
NSGA-II	Nondominated Sorting Genetic Algorithm II
QAV	Vendas de querosene de aviação
SMAPE	Symmetric Mean Absolute Percentage Error
SNE	Sistema Neuro-Evolucionário

Muitos anos depois, diante do pelotão de fuzilamento, o Coronel Aureliano Buendía havia de recordar aquela tarde remota em que seu pai o levou para conhecer o gelo. Macondo era então uma aldeia de vinte casas de barro e taquara, construídas à margem de um rio de águas diáfanas que se precipitavam por um leito de pedras polidas, brancas e enormes como ovos pré-históricos. O mundo era tão recente que muitas coisas careciam de nome e para mencioná-las se precisava apontar com o dedo.

Gabriel García Márquez, *Cem Anos de Solidão.*

1
Introdução

Uma *série temporal* é definida como uma sequência de observações, ordenadas em intervalos regulares de tempo, que podem apresentar maior ou menor correlação temporal entre seus valores [1]. Esta sequência é representada como um conjunto de valores discretos $x_1, x_2, ..., x_T$, onde, T indica o tamanho da série. A *Análise e Previsão de Séries Temporais*, por sua parte, é uma área cujo objetivo consiste em identificar padrões de comportamento na série, a partir dos quais é possível construir um modelo de ajuste para ser utilizado na estimação de valores futuros da série. Esta área é de constante e elevado interesse na comunidade científica, pois é capaz de oferecer suporte à tomada de decisão em diversos âmbitos. No setor financeiro por exemplo, prever um valor futuro de ações é de extremo interesse tanto para investidores em bolsas de valores quanto para pesquisadores acadêmicos que buscam entender o comportamento do mercado de ações. Outro exemplo relevante é observado no setor elétrico, onde as previsões de carga são indispensáveis tanto na operação eficiente e segura do sistema, quanto na tomada de decisão sobre a expansão e planejamento da operação. Da mesma forma, no setor industrial as previsões de demanda fundamentam o planejamento de curto e longo prazo nos departamentos de marketing, logística, produção e finanças.

Diversas técnicas têm sido utilizadas na área de previsão de séries temporais (PST), todas envolvendo uma medida de erro, que varia em maior ou menor medida conforme a natureza (linear ou não linear), e um grau de incerteza da variável estudada. A previsão do preço de ações, assim como a previsão meteorológica, por exemplo, são tipicamente atividades muito mais complexas do que a previsão de demanda energética no setor elétrico. Certamente, técnicas de PST baseadas em modelos estatísticos são as mais utilizadas, principalmente pelo fato de apresentarem maior grau de interpretabilidade, garantido pela simplicidade de análise e cálculo dos modelos matemáticos gerados. No entanto, fatores como a imensa disponibilidade de dados e recursos computacionais cada dia mais eficientes, tem possibilitado o desenvolvimento de novas técnicas oriundas da área de *Inteligência Artificial* (IA) [2]. Uma da suas principais subáreas, a *Inteligência Computacional* (IC) - ou *soft computing* - tem sido a opção preferida dos autores, abarcando principalmente os modelos de

Redes Neurais Artificiais (RNAs) como candidatos naturais na modelagem de séries temporais. Um dos fatores de interesse dos pesquisadores nestes modelos deve-se às caraterísticas das redes neurais que não são encontradas nas técnicas tradicionais de previsão: são modelos não paramétricos e aproximadores universais de funções [3], com capacidade de modelar adaptativamente relações complexas não lineares a partir de dados.

"Os métodos convencionais de previsão fornecem resultados precisos quando os dados estudados apresentam um comportamento linear, porém quando há um grau elevado de não linearidade, os métodos tornam-se pouco eficientes" [4].

A busca por modelos preditivos cada vez mais acurados tem estimulado também o desenvolvimento de abordagens baseadas na *combinação de modelos preditivos*. Diversos resultados empíricos apontam que as previsões melhoram quando previsões individuais originadas a partir de diferentes técnicas são combinadas.

Conforme descrito por Werner [5], previsões confiáveis podem exigir a combinação de várias técnicas de previsão, de modo a captar um maior grau de informação disponível na série. A incorporação de várias técnicas de previsão num mesmo esquema permite a construção dos chamados *sistemas multi previsores* (SMPs), que se bem projetados, levam a previsões consensuais superiores às previsões individuas [6]. Tais sistemas utilizam algum tipo de integração entre previsores, buscando compensar seus erros individuais para melhorar assim o desempenho final da previsão; vale salientar que as vantagens destes sistemas se tornam mais evidentes em casos onde os erros de previsão apresentam baixa correlação [7].

A combinação de modelos preditivos, proposta inicialmente por Bates e Granger ao final da década de 60 [8], é uma técnica bastante difundida e uma alternativa relevante à modelagem na previsão de séries temporais. Nas áreas de aprendizado de máquina e estatística por exemplo, técnicas de *bootstrapping, bagging, stacking*, e *boosting* [9–11] têm demostrado ser abordagens bem sucedidas, baseadas na ideia de que esquemas de combinação (comitês) levam a um melhor desempenho final dos modelos. Pesquisas prévias indicam, também, que mesmo que o melhor previsor possa ser identificado a cada instante, a combinação pode ainda ser uma estratégia atraente, pois oferece ganho potencial em diversificação ou complementariedade entre previsores [12]. Contudo, o sucesso desta abordagem vai depender de quão bem os pesos de combinação são estimados [7].

Estudos baseados em modelos de RNAs mostram-se como uma alternativa bem sucedida na modelagem de SMPs. Nesta linha são incluídos abor-

dagens que utilizam a rede neural como mecanismo de combinação não linear de modelos preditivos [13–16], assim como também modelos que fazem uso da rede neural para inferir os pesos que são utilizados posteriormente na ponderação linear de modelos preditivos [17]. Nessas abordagens, baseando-se nos dados históricos, é gerado um único vetor *estático* de pesos, o qual é aplicado sobre todo o horizonte de previsão [18, 21]. Entretanto, estudos em [19, 20] indicam que a geração *dinâmica* de pesos - abordagens onde os vetores de pesos gerados variam ao longo do horizonte de previsão - podem trazer ganhos no desempenho final da previsão quando comparado às abordagens de geração estática de pesos.

Apesar disso, o emprego de redes neurais com o objetivo específico para **geração dinâmica de pesos** em SMPs é uma área que não foi ainda explorada em profundidade, com exceção do modelo proposto por Valle dos Santos e Vellasco [12], conhecido como *Neural Expert Weighting* (NEW), o qual utiliza redes neurais Perceptron Multi-camada (*Multilayer Perceptron* - MLP) [22] para gerar dinamicamente vetores de pesos que posteriormente são utilizados para ponderar linearmente previsores estatísticos, ao longo do horizonte de previsão. Embora os resultados obtidos pelo modelo tenham sido promissores, uma das principais desvantagens observadas é o elevado custo computacional requerido pela política de treinamento adotada, pois a modelagem do sistema NEW baseia-se em uma busca exaustiva pelos parâmetros e hiper-parâmetros dos modelos de rede neural.

Sabe-se que, em abordagens envolvendo RNAs, não só o ajuste de pesos, mas também a topologia escolhida para o treinamento, é determinante no desempenho do modelo. Caso a topologia escolhida apresente uma pequena quantidade de neurônios e conexões, a rede pode não aprender satisfatoriamente, dada a pequena quantidade de parâmetros ajustáveis. Por outro lado, quando o número de neurônios e conexões é muito grande, a rede pode acabar se adaptando excessivamente aos padrões específicos de treinamento (*overfitting*), gerando uma perda na capacidade de generalização da rede quando submetida a novos padrões.

Apesar de ser fundamental para o sucesso da aplicação, a escolha da topologia é geralmente feita de maneira subjetiva, através de métodos de tentativa e erro. A fim de facilitar este dispendioso processo de tentativas, faz-se necessário um meio automático de escolha da melhor topologia. Esta necessidade motiva o estudo de técnicas de otimização de topologias, pois o problema pode ser formulado como um problema de busca em um espaço de possíveis arquiteturas onde cada ponto do espaço representa uma arquitetura específica [24].

Neste contexto, os Sistemas Híbridos Neuro-Evolucionários [23–25] surgem como uma alternativa viável na modelagem automática de RNAs a partir do uso de Meta-heurísticas prevalentes nos conceitos de seleção natural e recombinação genética, como é o caso dos Algoritmos Genéticos (AG) [26, 27]. Estes modelos combinam a capacidade de generalização e aproximação de funções das RNAs com um método eficiente de busca paralela. O objetivo dos AG em um sistema neuro-evolucionário é melhorar os algoritmos de aprendizado, automatizando, total ou parcialmente, o processo de *configuração da rede* neural, bem como o processo de *treinamento e atualização dos pesos* da mesma [28]. Os AGs são técnicas capazes de encontrar sistematicamente soluções ótimas ou sub-ótimas, em espaços de busca complexos, aplicando uma função objetivo (ou um conjunto de funções, no caso dos Algoritmos Evolutivos Multiobjetivo) adequada para avaliar soluções candidatas e um conjunto de operadores apropriados para percorrer o espaço de busca.

Entretanto, cabe ressaltar que abordagens para otimização de arquiteturas de RNAs usando AG não consideram, geralmente, informação sobre os valores dos pesos das conexões. Os genótipos dos indivíduos (soluções codificadas para serem modificadas através de operadores genéticos) da população, apenas codificam informações sobre a arquitetura das RNAs. Este fator faz com que a avaliação dos fenótipos esteja sujeita a ruídos, já que o desempenho das redes neurais depende, dentre outros fatores, das condições iniciais do treinamento (valores iniciais dos pesos das conexões). Assim, arquiteturas idênticas podem apresentar diferentes medidas de adaptabilidade e, se a avaliação dos genótipos é ruidosa, logo todo o processo evolutivo estará comprometido. Uma solução para resolver este problema é evoluir, simultaneamente, tanto os pesos das conexões como a arquitetura da rede neural [29, 30], fornecendo um mapeamento integral e não ambíguo entre um genótipo e seu fenótipo correspondente, o que permite que a avaliação de adaptabilidade do genótipo seja precisa e direta.

Contudo, devido ao seu mecanismo de busca global, os AG são usualmente ineficientes para encontrar resultados precisos em mínimos locais [24]. Para contornar esta limitação, diversos trabalhos [24, 29, 31] têm proposto sistemas híbridos onde o AG é combinado com outros algoritmos de busca local, tais como o próprio algoritmo *backpropagation* [32] e outros baseados em informações sobre o gradiente decrescente da função de erro [33–35]. Neste tipo de abordagem, a habilidade de AG para encontrar soluções globais pode ser utilizada para localizar inicialmente boas regiões no espaço de busca que poderão, em uma etapa posterior, ser mais precisamente vasculhadas por algum algoritmo de busca local.

No presente trabalho, uma meta-heurística evolutiva é proposta, em conjunto com o algoritmo *backpropagation*, para a otimização simultânea de arquiteturas e pesos em redes neurais *Multilayer Perceptron* aplicadas ao núcleo do modelo NEW. Esta nova abordagem, denominada NEW-GA, permite automatizar o processo de configuração da rede nos modelos NEW e, de acordo com resultados experimentais reportados adiante neste trabalho, é capaz de trazer considerável ganho no desempenho do modelo.

1.1 Objetivos

Como consequência dos argumentos anteriores, este trabalho teve como principal objetivo o desenvolvimento de uma abordagem para a otimização simultânea de arquiteturas e pesos em modelos de redes neurais artificiais do tipo Multilayer Perceptron (MLP), aplicadas ao núcleo do sistema NEW. A rede MLP é configurada automaticamente, através de uma técnica de treinamento híbrida que integra uma meta-heurística evolutiva de otimização multiobjetivo com o algoritmo de busca local *backpropagation*. Com essa nova abordagem, procurou-se gerar um modelo de ponderação mais acurado do que o modelo atual, além de avaliar os benefícios da utilização simultânea de técnicas de busca global e local no ajuste de pesos e configuração automática de redes neurais MLP como parte do modelo *NEW*.

1.2 Contribuições

As principais contribuições do presente trabalho são:

- Desenvolvimento de uma ferramenta computacional que permite a combinação de modelos estatísticos de previsão, a partir de uma abordagem híbrida neuro-evolucionária, baseada em técnicas de inteligência computacional, fornecendo, assim, um mecanismo para gerar automaticamente uma função de ponderação dinâmica de previsores;
- Combinação de técnicas de busca global e local para a configuração automática da rede neural MLP no modelo NEW;
- Comparação dos resultados em dois estudos de casos: 11 séries da *NN3 Forecasting Competition*[1] e séries de vendas de produtos derivados de petróleo, de modo a avaliar o desempenho e compará-lo com resultados reportados em trabalhos anteriores, utilizando o mesmo conjunto de dados.

[1] http:\\www.neural-forecasting-competition.com\NN3\index.htm

1.3 Descrição e Organização da Dissertação

A presente dissertação está organizada em quatro capítulos adicionais:

– Capítulo 2: Apresenta os fundamentos teóricos que servem de apoio para o desenvolvimento realizado nos capítulos subsequentes. O capítulo inclui a descrição dos métodos tradicionais de ponderação em SMPs, para depois apresentar o modelo NEW, detalhando sua estrutura e possibilidades de uso, assim como as possíveis contribuições na modelagem. Por outro lado, a ideia de sistemas híbridos é apresentada e a utilização dos algoritmos genéticos é contextualizada na área de otimização de redes neurais. Além disso, a meta-heurística evolutiva para otimização multi-objetivo *Non-dominated Sorting Genetic Algorithm* (NSGA-II), técnica de busca global na abordagem híbrida adotada, é descrita em maiores detalhes e os principais trabalhos publicados na área são mencionados e discutidos.

– Capítulo 3: Trata da metodologia proposta, apresentando os detalhes de funcionamento do processo desenvolvido para otimizar simultaneamente a arquitetura e os pesos das redes do tipo MLP envolvidas durante a construção de sistemas de ponderação NEW. Assim, são abordadas as seguintes etapas de elaboração: representação das soluções; inicialização e avaliação da população; aplicação do operador de seleção baseado no critério de dominância de Pareto [26]; geração de novas soluções; avaliação do critério de parada; e, finalmente, o mecanismo de escolha da solução ótima a partir de um conjunto de soluções fornecidas pelo algoritmo de treinamento.

– Capítulo 4: Apresenta os estudos de caso para avaliar o ganho em desempenho com respeito ao modelo NEW original, quando os seus parâmetros do núcleo são configurados automaticamente a partir da técnica de otimização proposta no presente trabalho. Estes estudos de caso são divididos em dois grupos de séries: derivados do petróleo (Seção 4.4), e versão reduzida da competição NN3 (Seção 4.5). Para cada grupo de séries o procedimento adotado foi o seguinte: (*i*) com base no conjunto de teste da série, avalia-se o desempenho tanto dos modelos individuais de previsão quanto dos métodos tradicionais de combinação, apresentados na Seção 2.1; (*ii*) fazendo uso da técnica de otimização proposta, ajustam-se a arquitetura e os pesos da rede MLP inserida no núcleo do modelo NEW, mantendo os mesmos hiper-parâmetros utilizados durante a construção do modelo original; (*iii*) uma vez que os parâmetros do núcleo no sistema NEW foram otimizados, avalia-

se o desempenho do modelo proposto no conjunto de teste da série e posteriormente compara-se com os desempenhos obtidos tanto em (i) quanto com o modelo original, aplicando-se testes de hipótese sobre os desempenhos, garantindo assim a validade das conclusões tomadas.

– Capítulo 5: Por fim, apresentam-se as considerações finais e algumas novas direções que podem ser seguidas, tanto para o aprimoramento da técnica de otimização aqui proposta, quanto em termos de comparações inexploradas no trabalho.

2
Fundamentação Teórica

Este capítulo aborda os fundamentos teóricos necessários para a compreensão do modelo proposto na presente dissertação. Na Seção 2.1 são descritos os métodos de ponderação de previsores mais citados ou sugeridos na literatura. A Seção 2.2 apresenta o sistema de ponderação dinâmica de previsores NEW, enquanto que na Seção 2.4 realiza-se uma revisão bibliográfica dos Sistemas Neuro-Evolucionários, com foco principal nas abordagens que utilizam AG como técnica de otimização em RNAs. Por fim, na Seção 2.5.1 é apresentada a meta heurística evolutiva de otimização multiobjetivo NSGA-II, que conjuntamente com o algoritmo *backpropagation* constituem a base da abordagem de treinamento híbrida para RNAs do tipo MLP proposta no presente trabalho.

2.1 Métodos Tradicionais de Ponderação de Previsores

Antes de abordar a descrição dos principais conceitos relacionados aos métodos tradicionais de ponderação, as seguintes definições, que serão referenciadas ao longo da presente dissertação, são apresentadas.

– Série histórica com τ observações (série dentro da amostra):

$$y^{\tau} = [y_1,\ y_2,\ \cdots,\ y_{\tau}].' \tag{2-1}$$

– Série de teste com horizonte máximo H (série fora da amostra):

$$y^{\tau+H|\tau} = [y_{\tau+1},\ y_{\tau+2},\ \cdots,\ y_{\tau+H}].' \tag{2-2}$$

– Vetor de previsões no instante $t+h$ ($h \leq H$), estimado com dados até t, para N previsores:

$$\hat{\mathbf{y}}_{t+h|t} = [\hat{y}_{t+h|t,1},\ \hat{y}_{t+h|t,2},\ \cdots,\ \hat{y}_{t+h|t,N}].' \tag{2-3}$$

– Vetor de pesos de combinação em $t+h$, estimado com dados até t, para N previsores:

$$\hat{\mathbf{w}}_{t+h|t} = [\hat{w}_{t+h|t,1},\ \hat{w}_{t+h|t,2},\ \cdots,\ \hat{w}_{t+h|t,N}].' \tag{2-4}$$

– Combinação convexa dos N previsores em $t+h$, estimada com dados até t:

$$y^{C}_{t+h|t} = \sum_{k=1}^{N} \hat{w}_{t+h|t,k}\hat{y}_{t+h|t,k}\,, \tag{2-5}$$

$$\text{onde } \sum_{k=1}^{N} \hat{w}_{t+h|t,k} = 1 \; e \; \hat{w}_{t+h|t,k} \geq 0\,. \tag{2-6}$$

A equação (2-5) apresenta o esquema mais comum para combinação linear de previsores, onde $\hat{y}_{t+h|t,k}$ corresponde à previsão h passos à frente feita pelo k-ésimo previsor, no instante t, $\hat{w}_{t+h|t,k}$ é o peso associado a esta previsão, enquanto que as restrições impostas em (2-6) tornam a combinação linear convexa. Este esquema de combinação (Figura 2.1) é geralmente desejável por sua simplicidade matemática e facilidade de interpretação, pelo fato de fornecer uma relação biunívoca entre pesos e previsores. Porém, outros motivos fazem com que este esquema seja de grande interesse: (i) garantia de não tendenciosidade na previsão combinada, sempre que os previsores envolvidos forem não tendenciosos e (ii) garantia de convexidade na combinação, o que fornece uma interpretação direta dos pesos, pois estes podem ser vistos como a percentagem de aporte de cada previsor no esquema linear de combinação. As caraterísticas de não tendenciosidade e convexidade são mais exploradas na Seção 2.1.2.

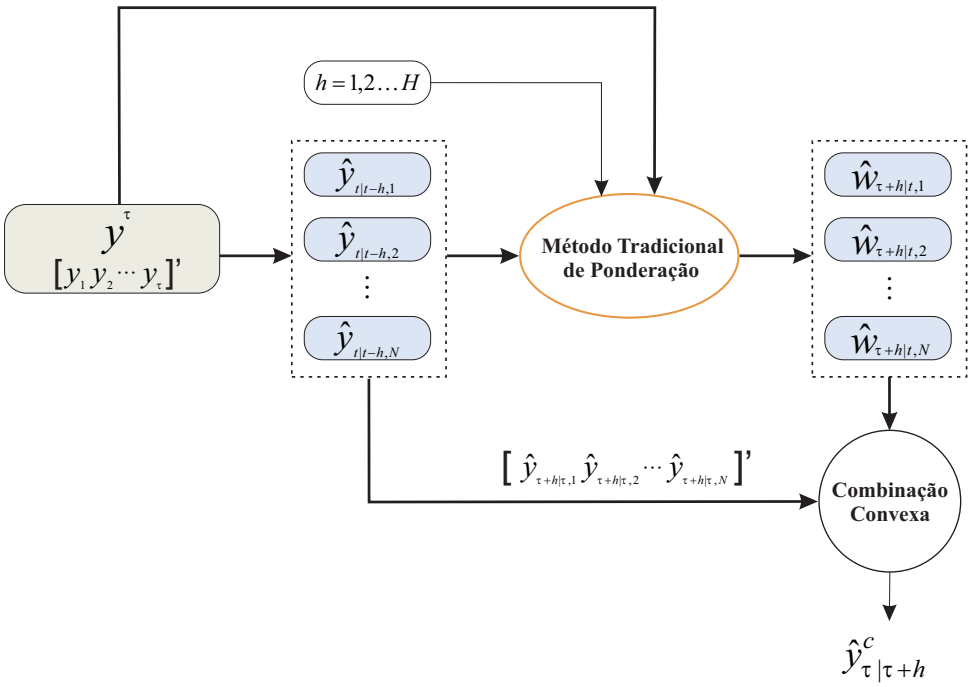


Figura 2.1: Esquema de combinação linear de previsores.

Na combinação linear de previsores existem métodos de geração de pesos consagrados e muito comuns na literatura, sendo todos eles - excluindo-se

o método por média simples - baseados nos desempenhos individuais dos previsores, os quais levam em conta informações disponíveis da série histórica a partir de uma janela de tempo que pode ser fixa ou expansiva. Quando considera-se uma janela fixa de tamanho ν, o cálculo do vetor de pesos no instante t levará em conta apenas as ν informações mais recentes; por outro lado, ao considerar uma janela de tempo expansiva, a estimativa do vetor de pesos levará em conta toda a informação anterior a t.

2.1.1 Média Simples

Estimar a média das previsões é sem dúvida a forma mais simples de combinação linear convexa. Apesar disso, apresenta resultados difíceis de serem superados [7], mesmo quando comparado com outros métodos de combinação linear mais complexos [36–38]. Ao calcular a média dos N previsores disponíveis, o vetor de pesos vai ser sempre o mesmo, independente do horizonte de previsão h:

$$\hat{w}_{t+h|t,k} = 1/N \; \forall \; k = 1, 2, \ldots, N \, . \tag{2-7}$$

2.1.2 Mínimos Quadrados Restritos

A estimação de pesos pelo método de mínimos quadrados restritos (MQR) parte da ideia de obter os pesos minimizando o erro quadrático entre os vetores de previsão e os valores atuais da série ao longo das observações disponíveis dentro da amostra. Nesse sentido, para cada vetor de previsões, determina-se um vetor de ponderação linear, no intervalo [0, 1], o que pode ser descrito como o seguinte problema de minimização:

$$\hat{w}_{t+h|t}(\nu) = \begin{bmatrix} \hat{w}_{t+h|t,1} \\ \hat{w}_{t+h|t,2} \\ \vdots \\ \hat{w}_{t+h|t,N} \end{bmatrix} = \arg\min_{w} \sum_{i=0}^{\nu-1} \left(y_{t-i} - \sum_{k=1}^{N} w_{t-i|t-i-h,k} \; \hat{y}_{t-i|t-i-h,k} \right)^2 , \tag{2-8}$$

$$\text{sujeito a } \sum_{k=1}^{N} w_{t|t-h,k} = 1 \text{ e } w_{t|t-h,k} \geq 0 \, . \tag{2-9}$$

onde N é o número de previsores disponíveis, ν corresponde ao tamanho da janela de tempo utilizada na estimação do vetor de pesos, e h é o número de passos à frente para o qual a previsão é feita a partir do instante de tempo t.

A expressão em (2-8) tem solução exata, utilizando o método dos mínimos quadrados ordinários [39], porém quando as restrições em (2-9) são aplicadas, o problema de minimização torna-se um problema com solução aproximada,

pois uma expressão analítica neste caso é inviável, sendo necessário recorrer a métodos iterativos [40], tais como o método do gradiente conjugado, para poder assim aproximar-se a sua solução.

O método dos mínimos quadrados restritos atende duas restrições desejáveis em esquemas de combinação linear de previsores, (i) terem componentes somando 1, (ii) não terem componentes negativos [7]. A primeira restrição garante não tendenciosidade na previsão combinada, desde que os previsores envolvidos sejam não tendenciosos. Entende-se como previsor não tendencioso, aquele que tem como valor esperado, E, de uma previsão a própria realização da série [41];

$$E(\hat{y}_{t+h|t,k}) \equiv y_{t+h}\,. \tag{2-10}$$

Quando aplica-se o valor esperado na equação (2-5), o efeito da restrição (i) é verificado:

$$E(y^{C}_{t+h|t}) = E\left(\sum_{k=1}^{N} \hat{w}_{t+h|t,k}\hat{y}_{t+h|t,k}\right)\,, \tag{2-11}$$

$$E(y^{C}_{t+h|t}) = \hat{w}_{t+h|t,1}E(y_{t+h|t,1}) + \cdots + \hat{w}_{t+h|t,N}E(y_{t+h|\,.t,N}) \tag{2-12}$$

Logo, considerando previsores individuais não tendenciosos, (2-12) é reduzida para (2-13), a qual indica que a previsão combinada é não tendenciosa, somente quando $\hat{w}_{t+h|t,1} + \hat{w}_{t+h|t,2} + \cdots + \hat{w}_{t+h|t,N} = 1$.

$$E(y^{C}_{t+h|t}) = y_{t+h}(\hat{w}_{t+h|t,1} + \cdots + \hat{w}_{t+h|t,N})\,. \tag{2-13}$$

A segunda restrição (ii), associada também à condição de não tendenciosidade, torna a combinação convexa, fazendo com que a magnitude da previsão combinada esteja dentro da região de valores definida pelas previsões individuais. A Figura 2.2 exibe esta questão.

As combinações convexas, além de fornecer melhor interpretação dos pesos, tendem a ser mais estáveis ou de menor variabilidade do que as não convexas.

A Equação (2-8), também pode ser adaptada, com o objetivo de considerar uma janela de tempo expansiva na estimação dos vetores de ponderação:

$$\hat{w}_{t+h|t} = \begin{bmatrix} \hat{w}_{t+h|t,1} \\ \hat{w}_{t+h|t,2} \\ \vdots \\ \hat{w}_{t+h|t,N} \end{bmatrix} = \arg\min_{w} \sum_{i=h+1}^{t} \left(y_i - \sum_{k=1}^{N} w_{i|i-h,k}\cdot\, \hat{y}_{i|i-h,k}\right)^2\,. \tag{2-14}$$

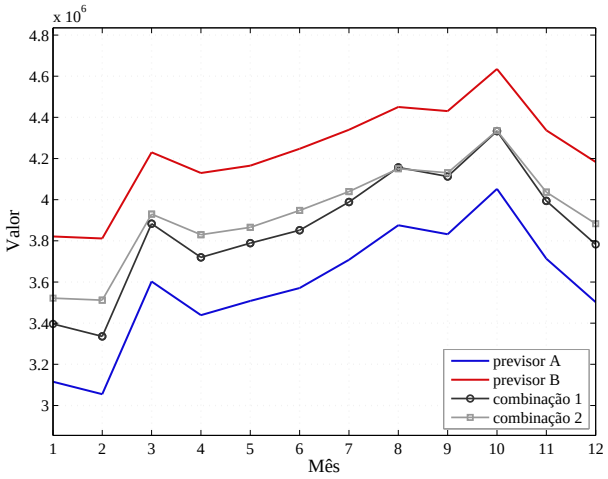


Figura 2.2: Combinações convexas entre os previsores A e B geram previsões limitadas à região por eles definida.

2.1.3 Bates & Granger

Nesta abordagem de ponderação simples, considera-se os pesos de combinação inversamente proporcionais ao erro quadrático médio (MSE), associado a cada *k-ésimo* previsor [8, 42, 43]:

$$\hat{w}_{t+h|t,k}(\nu) = \frac{\left[\nu^{-1}\sum_{i=0}^{\nu-1}(y_{t-i} - \hat{y}_{t-i|t-i-h,k})^2\right]^{-1}}{\sum_{j=1}^{N}\left[\nu^{-1}\sum_{i=0}^{\nu-1}(y_{t-i} - \hat{y}_{t-i|t-i-h,j})^2\right]^{-1}}, \tag{2-15}$$

sendo N o número de previsores disponíveis, ν o tamanho da janela de tempo considerada e h o horizonte de previsão. O denominador de (2-15), garante a convexidade (Seção 2.1.2) na geração do vetor de pesos no instante t, uma vez que o valor de MSE nunca é negativo. De forma similar que na seção anterior, (2-15) pode ser adaptada de modo a considerar uma janela de tempo expansiva na estimação dos vetores de pesos:

$$\hat{w}_{t+h|t,k}(\nu) = \frac{\left[\frac{1}{t-h}\sum_{i=h+1}^{t}(y_i - \hat{y}_{i|i-h,k})^2\right]^{-1}}{\sum_{j=1}^{N}\left[\frac{1}{t-h}\sum_{i=h+1}^{t}(y_i - \hat{y}_{i|i-h,j})^2\right]^{-1}}. \tag{2-16}$$

2.1.4 Método AFTER

Na metodologia *Aggregated Forecast Through Exponential Re-weighting* (AFTER) [20] o peso atual, associado a cada k-ésimo previsor, é obtido de forma recursiva, aplicando-se um fator sobre o peso mais recente:

$$\hat{w}_{t+h|t,k}(\hat{\sigma}) = \frac{\hat{w}_{t-1+h|t-1,k}\,\hat{\sigma}_{t|t-h,k}^{-1/2}\,\exp\left\{-\left[\frac{(y_t-\hat{y}_{t|t-h,k})^2}{2\hat{\sigma}_{t|t-h,k}}\right]\right\}}{\sum_{j=1}^{N}\hat{w}_{t-1+h|t-1,j}\,\hat{\sigma}_{t|t-h,j}^{-1/2}\,\exp\left\{-\left[\frac{(y_t-\hat{y}_{t|t-h,j})^2}{2\hat{\sigma}_{t|t-h,j}}\right]\right\}}\,. \tag{2-17}$$

O escalar $\hat{\sigma}_{t|t-h,k}$ corresponde a um estimador de variância condicional dos erros do *k-ésimo* previsor. Quando condiciona-se o estimador a uma janela fixa de tamanho ν, a seguinte expressão é obtida:

$$\hat{\sigma}_{t|t-h,k}(\nu) = \nu^{-1}\sum_{i=0}^{\nu-1}\varepsilon_{t-i|t-i-h,k}^2\,. \tag{2-18}$$

Onde os erros $\varepsilon_{t|t-h,k}$ são calculados da seguinte forma:

$$\varepsilon_{t|t-h,k} = y_t - \hat{y}_{t|t-h,k}\,. \tag{2-19}$$

Alternativamente, como nos métodos anteriores, é possível também utilizar uma janela de tempo expansiva na estimação dos pesos:

$$\hat{\sigma}_{t|t-h,k} = \frac{1}{t-h}\sum_{i=h+1}^{t}\varepsilon_{i|i-h,k}^2\,. \tag{2-20}$$

No método de ponderação AFTER, a contribuição de cada previsor é ponderada duas vezes (*re-weighted*) [19]: (i) pelo seu desempenho médio, através do termo $\hat{\sigma}_{t|t-h,k}$ e (ii) pelo seu último desempenho através do termo exponencial. Novamente, o denominador na equação (2-17) garante que o vetor de pesos $\hat{\mathbf{w}}_{t+h|t} = [\hat{w}_{t+h|t,1},\ \hat{w}_{t+h|t,2},\ \cdots\ \hat{w}_{t+h|t,N}]'$ gere uma combinação convexa (Seção 2.1.2).

2.2 Ponderação Neural de Expertos

A Ponderação Neural de Expertos (*Neural Expert Weighting* - NEW) é uma nova abordagem proposta em [12] para gerar modelos de ponderação dinâmica em esquemas de combinação linear de previsores, baseada em redes neurais do tipo *Multi-Layer Perceptron* (MLP) [22]. O sistema NEW faz uso da rede neural para ponderar dinamicamente previsores estatísticos ao longo do horizonte de previsão.

Nesta abordagem, o autor sugere o uso de previsores complementares quando sistemas multi–previsores são considerados, isto é, previsores que apresentem baixa correlação entre seus erros individuais de previsão. Neste contexto, o autor argumenta que a busca de previsores complementares pode seguir dois caminhos: (i) *combinar previsores de naturezas diferentes* ou (ii) *combinar previsores que sejam variantes da mesma técnica*. Por exemplo pode-se somar ou subtrair constantes nas previsões, usar hiper–parâmetros diferentes

no algoritmo de previsão ou utilizar alguma técnica de re-amostragem. Assim, a abordagem NEW propõe:

1. Selecionar previsores individuais seguindo os seguintes critérios: (i) terem naturezas diferentes na modelagem entre si, (ii) serem capazes de representar tendência e sazonalidade, dado que são caraterísticas recorrentes na maioria das séries temporais observadas no mundo real e (iii) ter um modelo padrão, bem conhecido e de fácil implementação. Deste modo, os modelos individuais de previsão utilizados foram derivados de duas das metodologias mais citadas na literatura, detalhadas no Anexo A:

   (a) Holt-Winters multiplicativo (HW);

   (b) ARIMA Box & Jenkins (BJ).

2. Utilizar um novo paradigma de combinação convexa, onde cada previsor individual é substituído por dois novos previsores, chamados de **previsores limiares**, cada um representando os **limites do intervalo de confiança** de 95% que abrange o modelo original de previsão. O objetivo é compensar as limitações impostas em esquemas de combinação convexa, pois como foi visto na Seção 2.1.2, o resultado da combinação nestes esquemas estará sempre limitado pela magnitude dos previsores individuais. Estes previsores limiares no sistema NEW são batizados com o nome do previsor original seguido dos prefixos "+" (limite superior) e "-" (limite inferior). A Figura 2.3 ilustra melhor esta proposta. O limite de confiança de 95% é definido como sendo o intervalo de ± 2 desvios-padrão a partir do previsor original; no sistema NEW considera-se, por simplificação, que o desvio-padrão é constante e vale $\sqrt{MSE}$ - raiz quadrada do erro quadrático médio de previsão, tomado dentro da amostra[1].

2.2.1 Estrutura do Sistema NEW

O sistema NEW utiliza um modelo de rede MLP para gerar, em cada ponto do tempo, um vetor de pesos convexos que ponderam linearmente as previsões disponíveis naquele ponto [12]. Este sistema é representado por:

$$\hat{w}_{\tau+h|\tau} = G(\hat{y}_{\tau+h|\tau}, h)' \,, \tag{2-21}$$

$$\text{sujeito a } h \leq H, \sum_{k=1}^{N} \hat{w}_{\tau+h|\tau,k} = 1 \text{ e } \hat{w}_{\tau+h|\tau,k} \geq 0 \,, \tag{2-22}$$

[1]Na abordagem NEW, ao subtrair uma constante da previsão original, deve-se cuidar para que não ocorram previsões com valores negativos.

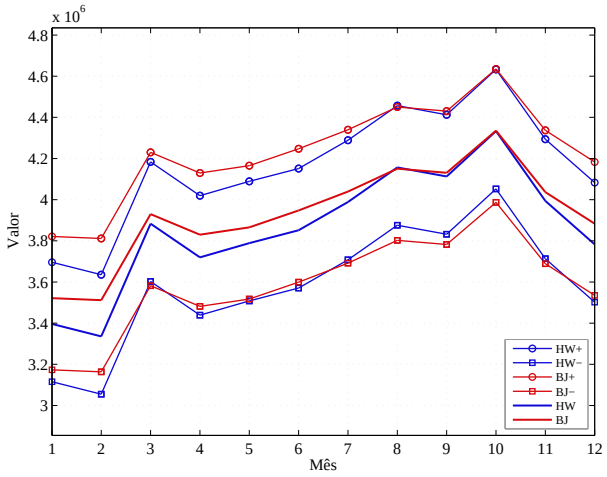


Figura 2.3: Previsores limiares HW e BJ para uma serie em particular. HW+, BJ+ e HW-, BJ- correspondem respetivamente aos limites superiores e inferiores dos previsores originais (HW e BJ).

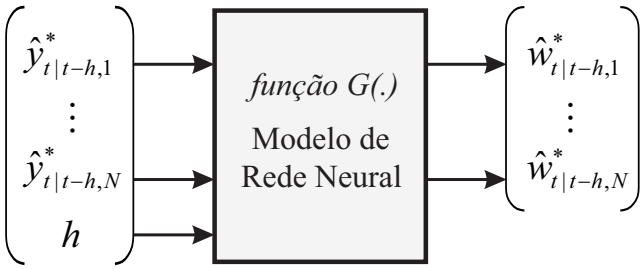


Figura 2.4: Esquema do modelo de treinamento NEW.

Onde $\boldsymbol{G}$ representa a rede neural propriamente treinada, e $\hat{y}_{\tau+h|\tau}$ corresponde ao vetor de previsões estimado para h passos a frente a partir do instante de tempo τ. O processo de treinamento da rede, por sua parte, é levado a cabo a partir do conjunto de vetores de previsão estimados dentro da amostra e seus correspondentes vetores de pesos ideais (Figura 2.4).

Para cada vetor de previsões $\hat{y}^*_{t|t-h}$ *dentro da amostra* é associado um vetor de pesos $\hat{w}^*_{t|t-h}$, considerado ideal, sendo este vetor calculado a partir de uma adaptação do método MQR (Seção 2.1.2), de modo a garantir a combinação convexa entre os modelos individuais de previsão. As equações (2-23) e (2-24) mostram como o método MQR é adaptado para funcionar como gerador de vetores de pesos de treinamento para o sistema NEW. Cabe ressaltar que esta adaptação leva em conta uma janela de tempo que pode ser de tamanho fixo ν ou expansivo, conforme foi apresentado na Seção 2.1.2.

$$\hat{w}^*_{t|t-h}(\nu) = \begin{bmatrix} \hat{w}_{t|t-h,1} \\ \hat{w}_{t|t-h,2} \\ \vdots \\ \hat{w}_{t|t-h,N} \end{bmatrix} = \arg\min_{w*} \sum_{i=0}^{\nu-1} \left(y_{t-i} - \sum_{k=1}^{N} w^*_{t-i|t-i-h,k} \cdot \hat{y}^*_{t-i|t-i-h,k} \right)^2 , \tag{2-23}$$

$$\text{sujeito a } \sum_{k=1}^{N} w^*_{t|t-h,k} = 1 \; e \; w^*_{t|t-h,k} \geq 0 \, . \tag{2-24}$$

Observa-se que a origem das previsões utilizadas no cálculo do vetor de pesos é sempre a mesma $(t-h)$ e, além disso, inclui-se também a desfasagem zero no cálculo do vetor, ou seja, a estimação do vetor de pesos no instante t leva em conta valores da série original menores ou iguais a t (e não apenas menores que t, como acontece nos métodos tradicionais, expostos na Seção 2.1), de modo que os índices nas equações (2-8) e (2-9) são alterados de $t+h \mid t$ para $t \mid t-h$. Esta adaptação é feita intencionalmente, como uma tentativa de trazer uma melhora em relação aos métodos tradicionais.

Assim, a construção dos pares de treinamento NEW (equação (2-25)), leva em conta duas grandezas vetoriais, com tantas dimensões quanto forem o número de previsores individuais, embora o vetor de entradas inclua adicionalmente o número de passos à frente h, como variável explicativa para o modelo neural.

$$\langle \{ \hat{y}^*_{t|t-h}, h \} \mid \hat{w}^*_{t|t-h} \rangle \, ; \; h \leq H < t < \tau \, . \tag{2-25}$$

De uma forma geral, e levando em conta que o sistema NEW considera modelos de redes MLP com apenas uma única camada escondida, cada componente do vetor de ponderação $\hat{w}^*_{t|t-h} = \left[\hat{w}^*_{t|t-h,1} \; \hat{w}^*_{t|t-h,2} \; \cdots \; \hat{w}^*_{t|t-h,N} \right]'$ à saída da rede, pode ser definido matematicamente pela equação:

$$\hat{w}^*_{t|t-h,k} = logs(z_k) = \frac{1}{1+e^{-z_k}} \, , \tag{2-26}$$

$$\text{onde } z_k = \sum_{i=1}^{p} \beta_{k,i} \tanh(u_i) + \beta_{k,0} \, , \tag{2-27}$$

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \, , \tag{2-28}$$

$$u_i = \sum_{j=1}^{N} \beta_{i,j} \cdot \hat{y}^*_{t|t-h,i} + \beta_{i,N+1} \cdot h + \beta_{i,0} \, . \tag{2-29}$$

O limite superior p no somatório de (2-27), referido na literatura como o número de *neurônios na camada escondida*, é um hiperparâmetro, que junto

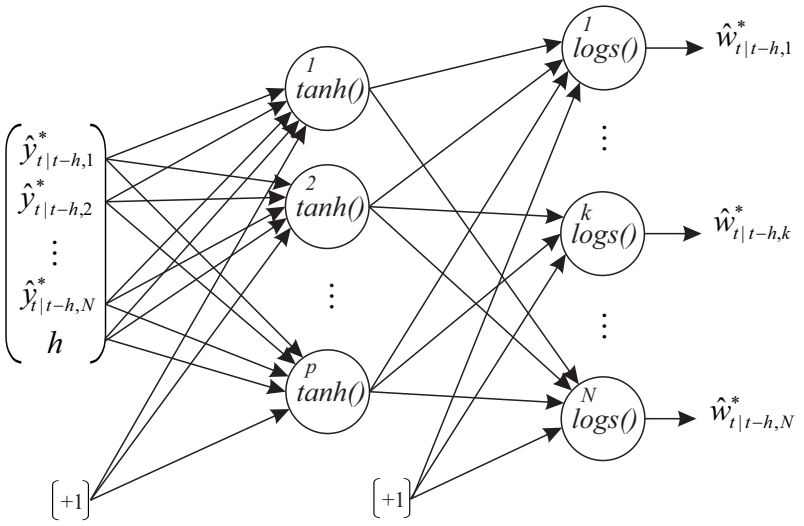


Figura 2.5: Representação gráfica do modelo de rede MLP utilizado no sistema NEW. Os círculos de 1 até p correspondem aos neurônios da camada escondida com função de ativação tangente hiperbólica, enquanto que os círculos de 1 até N indicam os neurônios de saída com função de ativação sigmóide logística.

com os parâmetros internos βs (equações (2-27) e (2-29)), chamados de *pesos sinápticos da rede*, são ajustados a partir da política de treinamento adotada pelo sistema NEW, denominada *holdout* repetido [44], a qual será abordada mais na frente. A função tangente hiperbólica em (2-27) apresenta propriedades que validam o teorema da aproximação universal [45], teorema que suporta o uso das redes neurais como modelos robustos de regressão não linear [12]. A função sigmoide logística em (2-26), por sua parte, garante as saídas da rede no intervalo [0,1], o qual é desejável, pois uma das caraterísticas no sistema NEW é a geração de vetores de ponderação *convexos.*

A formulação dada por (2-26) – (2-29), pode ser representada graficamente de acordo com a Figura 2.5; cada conexão (arco) representa um peso sináptico $\beta_{i,j}$, onde os índices i,j definem a conexão desde o neurônio j na primeira camada até o neurônio i na segunda camada. As conexões associadas com os índices $j=0$ representam os pesos das entradas fixas [+1], conhecidas como *bias* (viés).

2.2.2 Política de Treinamento no Sistema NEW

Na condição de modelos não lineares complexos, todas as redes MLP têm seus parâmetros internos ajustados por otimização aproximada, isto é, sem solução exata. Nesta linha, um dos algoritmos mais utilizados na prática é o algoritmo baseado no método do gradiente decrescente, conhecido como *backpropagation* [32]. No entanto, como descrito na Seção 2.3.2, o algoritmo

é sensível ao problema de convergência em mínimos locais, sendo dependente dos valores iniciais nos pesos sinápticos da rede.

Como foi mencionado na subseção anterior, a política de treinamento conhecida como *holdout* repetido permite o ajuste dos parâmetros e hiper-parâmetros nos modelos MLP dentro do sistema NEW. Esta política basicamente separa uma porcão do conjunto de vetores de treinamento (equação (2-25)) para formar um novo conjunto chamado de validação, de modo que o conjunto restante, denominado de conjunto de estimação, é utilizado para treinar a rede neural diversas vezes - cada vez com uma configuração diferente (número de neurônios na camada escondida e pesos sinápticos iniciais diferentes) - a partir do algoritmo *backpropagation*. Assim, o modelo de rede que apresentar melhor desempenho no conjunto de validação é selecionado. No sistema NEW, como nos modelos envolvendo séries temporais, é desejável selecionar as observações mais recentes do conjunto de treinamento para conformar o conjunto de validação, de forma a garantir a dependência sequencial no processo de aprendizado.

No processo de modelagem da rede neural MLP dentro do sistema NEW, duas questões relevantes podem ser observadas. A primeira delas refere-se ao elevado custo computacional requerido pela política de treinamento adotada, pois esta considera o treinamento de um número considerável de replicações para cada uma das arquiteturas de rede disponíveis (uma vez que o desempenho da rede depende da inicialização dos pesos sinápticos), o que, portanto, torna ineficiente o processo, pois uma *busca exaustiva pelos parâmetros e hiper-parâmetros da rede* deve-se levar a cabo antes de poder selecionar o melhor modelo. Assim, uma das direções seguidas no presente trabalho é avaliar uma abordagem capaz de oferecer uma busca automática e paralela pelos parâmetros e hiper-parâmetros no processo de modelagem da rede MLP como parte do sistema NEW.

A segunda questão observada no processo de modelagem da rede MLP refere-se à existência de duas formas de avaliar o desempenho dos modelos sobre o conjunto de validação. *(i)* A primeira delas consiste em comparar os vetores de pesos de combinação inferidos pela rede ($\hat{w}_{t+h|t}$) com os vetores de pesos de combinação ideais ($\hat{w}^*_{t+h|t}$), a partir de uma métrica de erro chamada de *Erro de Treinamento* (E_T). *(ii)* A segunda forma de avaliar o desempenho dos modelos tem uma interpretação mais direta, pois consiste em comparar as observações originais da série (y_{t+h}) com respeito às previsões combinadas ($y^C_{t+h|t}$), obtidas a partir da *combinação convexa* entre as previsões individuais disponíveis ($\hat{y}_{t+h|t}$), quando são ponderadas e combinadas linearmente a partir dos vetores de pesos inferidos pela rede ($\hat{w}_{t+h|t}$). Esta segunda forma de avaliar

o desempenho utiliza uma métrica de erro chamada de *Erro de Combinação* (E_C).

Neste sentido, o sistema NEW propõe utilizar uma função de erro multi-critério, a partir da soma das duas métricas de erro previamente mencionadas, de modo que o desempenho de cada modelo possa ser avaliado conjuntamente sobre o conjunto de validação. Embora a função de erro proposta considere o desempenho da rede em *(i)* e *(ii)*, o treinamento de cada modelo, levado a cabo a partir do algoritmo *backpropagation,* é guiado apenas pela minimização do *Erro de Treinamento*, sem levar em conta o comportamento do *Erro de Combinação* durante o ajuste dos parâmetros da rede, pois este último é avaliado apenas quando a etapa de treinamento é concluída, isto é, quando o *Erro de Treinamento* é mínimo. Neste sentido, uma segunda direção seguida no presente trabalho consiste em avaliar uma abordagem de treinamento que leve em conta a avaliação *simultânea* das duas métricas de erro *durante o ajuste dos parâmetros da rede*, pois na abordagem atual é assumido que a minimização do *Erro de Treinamento* implica a minimização do *Erro de Combinação*, fato que pode trazer perda no desempenho final do modelo, na medida que as duas métricas de erro apresentem comportamento *conflitante* entre elas.

2.3 Redes Neurais

As Redes Neurais Artificiais (RNA) têm sido utilizadas com sucesso para resolver problemas diferentes e de caraterísticas gerais, tendo como área de aplicação uma gama bastante extensa. Entre as aplicações usuais das RNAs encontra-se: reconhecimento e classificação de padrões, controle, agrupamento, aproximação de funções, análise de imagens, mineração de dados e, o foco do presente trabalho, previsão de séries temporais [22, 46].

As RNAs constituem uma ferramenta de grande importância na atualidade, por sua capacidade de "aprender" padrões através de treinamento, o que torna seu uso importante no desenvolvimento da Inteligência Artificial. A principal ventagem de uma rede neural é a sua habilidade de aproximar relações funcionais, particularmente quando as relações não são bem definidas e/ou são não lineares.

Uma RNA é um modelo computacional, inspirado biologicamente no funcionamento do cérebro, formada por unidades de processamento (denominados neurônios) e conexões entre esses elementos, com pesos ligados a essas conexões, formando assim uma estrutura neural, sobre a qual são implementados algoritmos de aprendizado que ajustam os parâmetros da estrutura levando em conta o tipo de conhecimento a ser codificado na rede. RNAs são chamadas de modelos conexionistas devido à grande importância das conexões entre os

neurônios para o processamento da rede. É importante ressaltar que os pesos das conexões são os responsáveis pelo "conhecimento" codificado nas RNAs.

As RNAs possuem algumas caraterísticas que se tornam alvos de intensas pesquisas, tais como:

- **Aprendizado e Adaptação**: uma das propriedades mais importantes de uma RNA é a capacidade de aprender por intermédio de exemplos e realizar inferências sobre o que aprendeu, melhorando gradativamente o seu desempenho [46];
- **Generalização**: uma RNA é capaz de generalizar o seu conhecimento a partir de exemplos anteriores e com isso lidar com informações nunca antes vistas no conjunto de treinamento;
- **Processamento Paralelo**: caraterística de processamento intrínseca das RNAs, herdada da sua inspiração biológica no cérebro;
- **Robustez**: capacidade das RNAs em prover uma saída coerente mesmo sendo alimentadas com dados ruidosos ou incompletos.

Existe uma variedade bastante razoável de modelos de RNAs que executam vários tipos de tarefas. No presente trabalho, dado que é uma extensão do modelo NEW original, será focada uma estrutura particular das RNAs, a rede Perceptron Multi-camadas (*Multilayer Perceptron* - MLP), um modelo de rede amplamente empregado e aceito pela comunidade acadêmica na utilização em aplicações de previsão [47–49].

2.3.1 Perceptron Multicamadas

A arquitetura da rede MLP (Figura 2.6) é composta por neurônios conectados, seguindo uma formação em camadas. Essas conexões são responsáveis por propagar as entradas da RNA. As unidades de entrada, que compõem essa camada, têm o objeto de difundir o sinal inicial sem nenhuma modificação para a segunda camada. Os dados são apresentados à rede pela camada de entrada, são processados pelas camadas subsequentes e por último a rede gera uma saída para a informação que lhe foi apresentada. É importante salientar que se cada camada de neurônios é somente ligada à camada subsequente (com exceção da camada de saída, que só recebe ligações) e a informação trafegar em um único sentido, a rede é chamada de *feedforward* (Figura 2.6). Caso existam conexões de retorno entre as camadas, a rede é chamada de recorrente, Figura 2.7.

Redes MLP apresentam um poder computacional maior do que aquele apresentado pelas redes perceptron que possuem uma única camada [22,46,50].

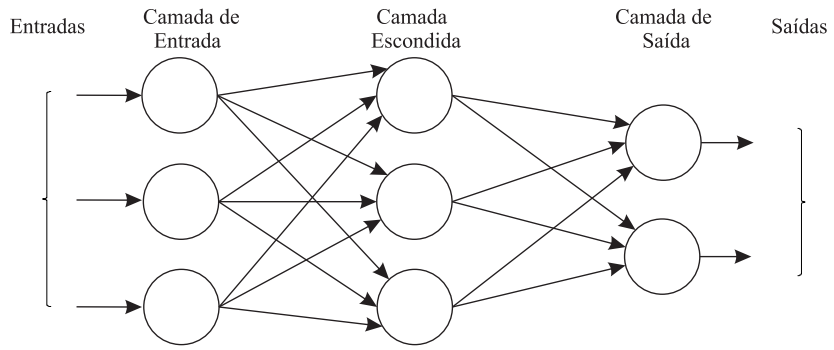


Figura 2.6: Arquitetura de uma rede MLP *feedforward*.

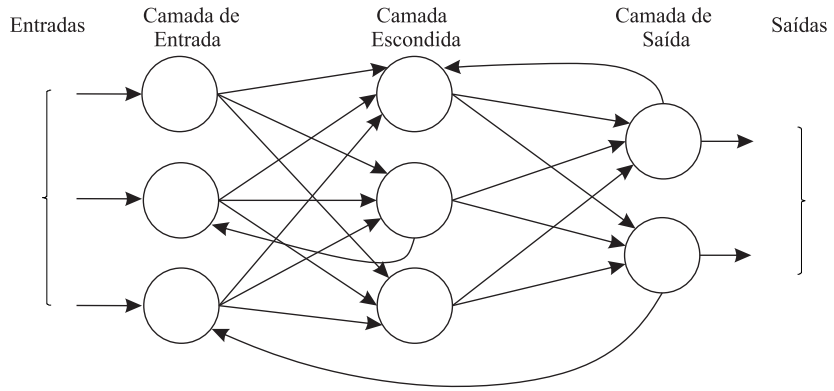


Figura 2.7: Arquitetura de uma rede MLP recorrente.

Tal poder computacional é conseguido com a adição de uma ou mais camadas intermediárias entre a entrada e a camada de saída. Em [51], foi provado que são necessárias, no máximo, duas camadas intermediárias, com um número suficiente de neurônios por camada que é definido de forma empírica, para se aproximar qualquer função. Também foi provado que apenas uma camada intermediária é suficiente para aproximar qualquer função contínua.

Desta forma, a definição de uma rede MLP apresenta as seguintes caraterísticas:

- **Definição da Estrutura**: determinação do número de camadas intermediárias e das quantidades de neurônios em cada uma dessas camadas;
- **Tipo de Conexão**: determinação do tipo de conexões entre as camadas da rede (*feedforeward* ou recorrente);
- **Função de Ativação**: determinação das funções de ativação dos neurônios. Os neurônios da uma mesma camada geralmente utilizam o mesmo tipo de função de ativação, que pode ser diferente para cada camada, dependendo do problema em particular.

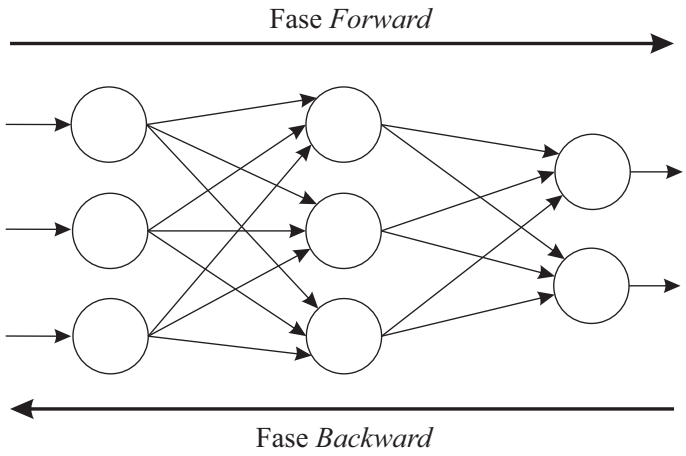


Figura 2.8: Fases do Algoritmo *Backpropagation*.

2.3.2 Algoritmo de Treinamento *Backpropagation*

Um dos algoritmos mais conhecidos e difundidos para o processo de treinamento de redes MLP é o algoritmo *backpropagation,* sendo este um algoritmo supervisionado de propósito geral que utiliza pares de entrada e saída para, através de um mecanismo de correção de erros, ajustar os pesos da rede. O treinamento é executado em duas fases que percorrem a rede em sentidos opostos, conforme ilustrado na Figura 2.8.

Na fase *forward* um dado padrão é apresentado à rede, fazendo com que as ativações para este padrão sejam propagadas pela rede até a última camada, onde é gerada uma saída correspondente daquele padrão de entrada. Na fase *backward* a saída gerada pela rede é confrontada com a saída desejada para o referido padrão de entrada, sendo esta informação utilizada para atualizar os pesos das conexões entre os neurônios.

O algoritmo *backpropagation* é poderoso, mas bastante dispendioso em termos de processamento computacional para treinamento. Isto se deve ao fato do *backpropagation* usar o método do gradiente decrescente de uma função para minimizar o erro global da rede [22]. Devido à utilização do método baseado no gradiente, o *backpropagation* pode apresentar o problema denominado mínimo local, que ocorre quando o processo de aprendizado pára quando o erro se localiza numa região de mínimo local ao invés de mínimo global [50].

2.4 Sistemas Neuro-Evolucionários

Atualmente, tem merecido crescente atenção, na área de Inteligência Artificial, o desenvolvimento de *sistemas híbridos,* que resultam da combinação de duas ou mais técnicas distintas para resolver um dado problema. A motivação para tais sistemas está no fato de que as diversas técnicas existentes

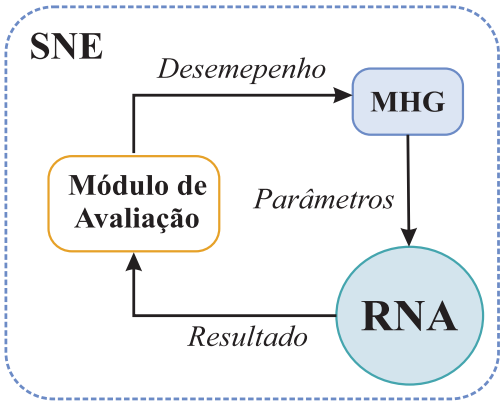


Figura 2.9: Diagrama genérico de um SNE.

de Inteligência Artificial podem ser adequadas para determinados casos, mas podem apresentar deficiências significativas para resolver outros tipos de problemas. Estas limitações estimulam o estudo dos sistemas híbridos, os quais procuram combinar as caraterísticas favoráveis de duas ou mais técnicas, com o intuito de superar as limitações que cada uma apresenta individualmente na resolução do problema de interesse.

Entre estes sistemas híbridos, especial ênfase tem sido dedicada à combinação de Meta-Heurísticas Genéticas (MHGs) e Redes Neurais Artificiais (RNAs), constituindo os chamados Sistemas Neuro-Evolucionários (SNEs). Define-se neste contexto por MHGs o conjunto de Algoritmos Evolucionários usados para a parametrização de uma RNA, como, por exemplo: *Algoritmos Genéticos (AG)* [27], *Programação Genética (PG)* [52], *Algoritmos Evolutivos Multiobjetivos (MOEA)* [26, 53, 54], entre outros.

O usual na área dos SNE é a RNA usufruir da MHG como mecanismo de treinamento dos pesos sinápticos em uma topologia fixa ou a busca de arquiteturas ótimas. O esquema básico e mais frequente de um SNE é mostrada na Figura 2.9.

Nas próximas subseções apresentam-se comentários sobre a aplicação de algoritmos genéticos para otimização de redes neurais artificiais. Vale ressaltar que o objetivo não é descrever detalhadamente os métodos de otimização, pois tais explicações já existem em grande quantidade na literatura. A intenção é apresentar uma visão geral de como esta técnica já foi empregada para otimizar redes neurais, a fim de contextualizar o presente trabalho e apontar direções para a elaboração da abordagem de treinamento híbrida, aplicável ao núcleo do sistema NEW, aqui proposta.

2.4.1 Algoritmos Genéticos: Visão Geral

Algoritmos Genéticos (AG) são técnicas de busca e otimização prevalentes nos conceitos da seleção natural e recombinação genética. Foram principalmente inseridos no mundo computacional por John Holland na década de 1970 [27], e mais tarde popularizados por David Goldberg, a partir de 1989 [26]. Holland estudou a evolução natural considerando esta como um processo *robusto,* simples e poderoso, que poderia ser adaptado para obtenção de soluções computacionais eficientes em problemas de otimização. O conceito de robustez relaciona-se ao fato de os AGs, independentemente da escolha dos parâmetros iniciais, em geral, produzirem soluções de qualidade [26, 55].

O primeiro aspecto a ser considerado num AG é a representação dos parâmetros do problema, ou seja, a codificação das possíveis soluções do problema em estruturas que podem ser manipuladas pelos algoritmos genéticos. Uma solução possível do problema, antes da codificação, recebe o nome de *fenótipo.* Cada fenótipo é codificado em uma estrutura, que recebe o nome de *indivíduo, cromossomo* ou *genótipo.* AGs trabalham com um conjunto de indivíduos simultaneamente, e este conjunto recebe o nome de *população.* Cada indivíduo da população é associado a uma *aptidão,* que representa a capacidade da solução candidata de resolver o problema de interesse [27].

O funcionamento de um algoritmo genético envolve uma sequência de iterações, que também são chamadas de *gerações.* A cada geração, a população passa pelos processos de *seleção* (escolha dos indivíduos a serem reproduzidos) e *reprodução* (combinação e/ou modificação dos indivíduos selecionados, produzindo os indivíduos da próxima geração). Um dos métodos de seleção mais utilizados é o da *seleção por torneio* [56]. Neste método, um subconjunto da população de tamanho k é sorteado e os melhores indivíduos desse grupo são selecionados para decidir qual irá reproduzir. Uma importante propriedade deste método é que não leva em consideração o ranqueamento que o individuo ocupa na população, permitindo assim, um processo de seleção com menores tendências [54]. A reprodução, por sua parte, é feita por meio de *operadores genéticos,* que procuram manter as caraterísticas dos indivíduos selecionados nos novos indivíduos a serem gerados. Os operadores genéticos principais são o *cruzamento* e a *mutação.* O cruzamento é responsável pela combinação de caraterísticas dos *pais* (indivíduos originais), a fim de gerar *filhos* (indivíduos criados a partir da reprodução), sendo aplicado com uma determinada taxa, chamada de *taxa de cruzamento.* A mutação, por sua vez, procura manter a diversidade genética na população, fazendo modificações arbitrárias em uma ou mais partes de indivíduos escolhidos aleatoriamente. A taxa com que este operador é aplicado recebe o nome de *taxa de mutação.* Uma representação geral

Algoritmo 1 Pseudocódigo de um AG típico.

```
Passos Iniciais:
Entrada: Parâmetros típicos [58]
Saída: População final de indivíduos
Inicialização da população com soluções candidatas aleatórias
Avalia cada indivíduo da população
repita
    Seleciona pais
    Recombina pares de pais
    Muta descendentes resultantes
    Avalia novos indivíduos
    Seleciona indivíduos para a nova geração
até Condição de parada satisfeita;
```

de um AG típico pode ser vista no pseudocódigo do Algoritmo 1, baseando-se em [54].

A utilização dos operadores de seleção e reprodução nos AGs equilibra dois objetivos aparentemente conflitantes: o aproveitamento das melhores soluções e a exploração do espaço de busca (*Exploiting and Exploring* [57]). O processo de busca é, portanto, multidimensional, preservando soluções candidatas e provocando a troca de informação entre as soluções exploradas [59,60]. Finalmente, o AG pára quando um determinado número de gerações é alcançado, quando a melhor solução é encontrada (quando esta é conhecida), quando há perda relevante e irreparável da diversidade dos indivíduos da última população, ou quando nas últimas k gerações não há melhora da aptidão média ou máxima.

O fato de existirem mais trabalhos que se utilizam de algoritmos genéticos para otimizar redes neurais certamente está associado às inspirações biológicas de ambos os métodos [61,62]. Na maioria dos trabalhos publicados observa-se basicamente duas abordagens. Na primeira, os AGs são utilizados para ajustar todos os parâmetros da RNA, incluindo os valores dos pesos. Assim, o AG é o responsável por conduzir o processo de treinamento da RNA. Na segunda abordagem, o AG interage com a RNA criando várias redes que serão possíveis candidatas a melhores modelos para solução do problema, mas o treinamento é realizado por um algoritmo de treinamento convencional, como por exemplo, o *backpropagation.*

2.4.2 AG Para Otimização de Pesos em RNAs

O treinamento de uma rede neural artificial pode ser formulado como a minimização de uma função de erro, como, por exemplo, o erro médio quadrático entre saídas da rede e saídas desejadas de todos os padrões de trei-

namento, através de um ajuste iterativo de pesos. Como foi mencionado na Subseção 2.2.2, um dos algoritmos de aprendizado mais utilizados para o treinamento de redes MLP, é o *backpropagation* [32]. Este algoritmo, enquadrado entre os métodos de *gradiente decrescente,* utiliza informações sobre a derivada da função de erro durante o processo de treinamento. Apesar de existirem muitas aplicações eficientes do algoritmo *backpropagation* para o treinamento de redes MLP, tal algoritmo apresenta, em muitos casos, o grave problema da *convergência local,* ou seja, o estancamento em mínimos locais da função de erro. Neste sentido, várias abordagens têm sido propostas para contornar este problema, como, é o caso do conhecido *termo de momentum* [63].

Algoritmos baseados no gradiente decrescente são geralmente considerados como métodos *locais,* pois são concebidos para se aproximar iterativamente ao ponto mínimo, utilizando informações sobre o gradiente da função de erro, que são informações locais. Tais informações servem para determinar a direção e magnitude do ajuste de pesos mais adequado para caminhar em direção ao mínimo.

Técnicas de treinamento, baseadas em AGs, são chamadas de *globais,* pois são concebidas para realizar uma busca mais geral no espaço, procurando o ponto de mínimo seguindo um processo que leva em conta aspectos globais da superfície de erro. Estas técnicas têm sido amplamente utilizadas para melhorar o processo de treinamento em RNAs, contornando o problema da convergência em mínimos locais, pois estes métodos de otimização global não precisam de informações sobre o gradiente da função do erro [24].

As abordagens dedicadas ao estudo da aplicabilidade de AGs para a otimização dos pesos das conexões de redes neurais podem ser classificadas seguindo o tipo de representação usado na codificação de pesos: as que fazem uso da *representação binária* e as que utilizam *representação real*.

Na representação binária, cada peso de uma conexão é representado por uma sequência de bits (0s ou 1s), cujo comprimento pode ser fixo ou não. Assim, a representação de uma rede neural é construída, a partir da concatenação de todos os seus pesos em um único cromossomo. Tal representação tem com vantagem a simplicidade, principalmente no que se refere à aplicação direta dos operadores clássicos de cruzamento e mutação, não havendo a necessidade de elaboração de operadores muito elaborados para lidar com este tipo de cromossomo.

A principal desvantagem da representação binária é o balanço que é necessário fazer entre a precisão da representação e o tamanho final do cromossomo. Se a codificação utiliza poucos bits para representar os pesos, o treinamento pode não ter sucesso, pois os valores reais dos pesos podem

não ser representados com precisão suficiente pelos valores discretizados. Por outro lado, se muitos bits forem usados para representar os pesos, o tamanho excessivo dos cromossomos pode tornar o processo evolutivo bastante ineficiente [24].

Na representação real, ou *direta*, cada peso é representado diretamente pelo seu valor real, de modo que cada cromossomo é formado por um vetor de números reais. Dessa forma, operadores tradicionais de cruzamento e mutação não são mais aplicáveis. Montana e Davis [64] definiram uma grande variedade de operadores genéticos que incorporam diversas heurísticas a respeito do treinamento de redes neurais. A ideia por trás destes operadores é preservar o comportamento de extração de caraterísticas que são formados ao redor dos neurônios durante a evolução dos pesos.

Independentemente da representação utilizada, sempre é desejável que os pesos codificados de um mesmo neurônio fiquem juntos no cromossomo, uma vez que estes se comportam como extratores e detetores de caraterísticas, a separação de seus pesos de entrada pode levar à perda deste comportamento quando aplicados operadores de cruzamento [24].

Dentro das abordagens utilizadas para otimização de pesos, uma técnica que merece destaque, além de ser uma das principais diretrizes no presente trabalho, é o *treinamento híbrido,* a qual surge para contribuir com o processo de busca por soluções mais precisas, localizadas em mínimos locais. Neste tipo de abordagem híbrida, a rede neural é treinada com um algoritmo genético e, em seguida, é submetida a um treinamento com um método de gradiente decrescente, encarregado de realizar um ajuste fino dos pesos. Neste contexto, o algoritmo genético tem a função de localizar boas regiões no espaço de busca, enquanto que o método do gradiente fica com o papel de identificar o ponto de mínimo destas regiões *"busca local"*. Abordagens de treinamento híbrido têm sido utilizadas com sucesso em diversas áreas de aplicação.

2.4.3 AG Para Otimização de Arquiteturas em RNAs MLP

A escolha da topologia em abordagens envolvendo RNAs é um fator crítico no desempenho do modelo. Uma quantidade pequena de neurônios e conexões na rede pode diminuir a capacidade de aprendizado, devido à quantidade insuficiente de parâmetros ajustáveis. Por outro lado, uma rede neural com quantidade elevada de neurônios e conexões pode apresentar dificuldades para generalização quando forem apresentados padrões ainda não vistos.

A escolha da arquitetura normalmente é feita através de uma sequência de tentativas com diversas topologias. Portanto, torna-se necessário um método

automático para a definição da topologia da rede neural, visando evitar este processo ineficiente de tentativas e erros.

Como foi mencionado no contexto do capítulo anterior, a definição da arquitetura de uma rede neural pode ser formulada como um problema de otimização, em que cada ponto no espaço de busca representa uma arquitetura específica da rede. Neste contexto, algoritmos genéticos, como técnica de otimização global, têm sido utilizados com relativo sucesso para contornar este problema.

Nestas abordagens, a codificação das soluções é crucial no processo evolutivo. O método deve ser capaz de excluir redes inválidas, além de verificar se os operadores de reprodução, quando aplicados aos indivíduos selecionados, geram topologias válidas.

A questão mais relevante na especificação do esquema de representação de arquiteturas é a quantidade de informação sobre a arquitetura que se deve codificar em um cromossomo. Um primeiro tipo de codificação, chamada de *indireta* ou de *alto nível,* especifica apenas os parâmetros mais importantes da arquitetura, como o número de camadas escondidas e neurônios ocultos em cada camada, de modo que os parâmetros internos da rede são determinados a partir de algum algoritmo de treinamento externo [25]. Já na codificação *direta* ou de *baixo nível*, todos os detalhes da arquitetura são codificados. Neste caso, informações sobre todas as conexões de cada neurônio são embutidas na construção dos cromossomos. Como esperado, cada uma destas abordagens possui vantagens e desvantagens, que podem ser mais ou menos relevantes de acordo com a natureza do problema abordado.

Na representação direta, cada conexão da rede é especificada individualmente. Dessa forma, cada arquitetura com n neurônios pode ser representada por uma matriz de dimensões n x n, onde cada uma das n linhas simboliza um nó da topologia, bem como cada uma das n colunas. Assim, o elemento na posição *(i,j)* da matriz, representa a conexão que parte do neurônio i e entra ao neurônio j. Dessa forma, a arquitetura da rede é codificada no cromossomo a partir da concatenação das linhas da matriz.

A fim de diminuir a quantidade de bits utilizados na representação direta do cromossomo, podem ser aplicadas restrições na concatenação, dependendo de algum conhecimento prévio. Por exemplo, em uma arquitetura *feedforward,* como é o caso dos modelos utilizados no sistema NEW, sabe-se que os elementos da diagonal principal da matriz, bem como os elementos abaixo da diagonal principal, são todos nulos, pois neste tipo de arquiteturas não existem conexões que partem para neurônios de camadas anteriores, nem conexões de um neurônio para ele próprio (Figura 2.5). Então, a concatenação das linhas

da matriz é feita utilizando apenas os elementos acima da diagonal principal, o que traz uma redução considerável do número de bits utilizados na codificação do cromossomo [24]. Na Figura 2.10, apresenta-se um exemplo deste tipo de codificação.

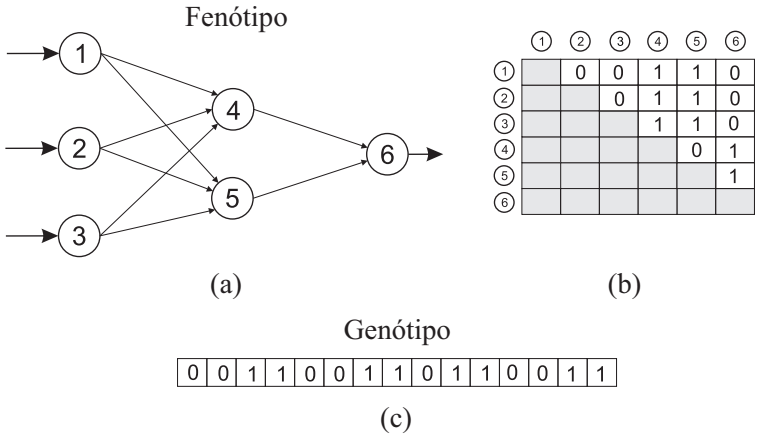


Figura 2.10: (a) Rede neural *feedforward*; (b) Matriz de conexões entre neurônios; (c) Vetor cromossomo, representado pelos valores binários da matriz triangular superior.

A representação direta têm como vantagem a facilidade de implementação e de conversão entre genótipo e fenótipo, mas a desvantagem é que esta representação pode tornar o espaço de busca excessivamente amplo, implicando a necessidade de um maior número de iterações por parte do algoritmo. Por este motivo, geralmente, *a máxima topologia é definida previamente pelo usuário*, limitando o crescimento das redes e permitindo uma maior exploração no espaço de busca definido.

É valido lembrar também que a representação indireta pode reduzir consideravelmente o tamanho dos cromossomos, pois este tipo de representação é capaz de gerar uma especificação mais compacta das topologias, mas, em alguns casos, pode trazer perda na capacidade de generalização da rede. Outro aspecto importante que deve-se mencionar é que quando o método otimiza apenas a arquitetura, a avaliação de uma topologia contém ruído, já que um genótipo sem informação sobre os pesos da rede é aproximado por um fenótipo contendo uma rede treinada [29]. Dependendo dos pesos iniciais escolhidos e dos parâmetros de treinamento, a avaliação de um mesmo genótipo pode gerar resultados diferentes. Para evitar este problema, a avaliação de cada arquitetura pode ser feita através de várias inicializações de pesos, para que seja computada a média dos resultados obtidos, mas o procedimento leva a um aumento dramático no tempo de execução.

2.4.4 Otimização Simultânea de Pesos e Arquiteturas em RNAs

Conforme mencionado anteriormente, quando é otimizada somente a arquitetura, a avaliação de uma dada topologia está sujeita a ruídos, pois uma rede neural treinada previamente é utilizada para calcular o custo da solução, sendo que a solução representa uma topologia de rede sem informação acerca dos pesos.

Uma possível abordagem para reduzir os efeitos deste ruído, e assim evitar comprometer o processo evolutivo, é a *otimização simultânea de arquiteturas e pesos*. Dessa forma, uma rede é especificada não só pela sua topologia, mas também pelo seu conjunto completo de pesos, fornecendo um mapeamento integral e não ambíguo entre um genótipo e seu fenótipo correspondente, o que permite que a avaliação de cada indivíduo seja precisa e direta.

As considerações feitas nas subseções anteriores a respeito da codificação, cruzamento e mutação de indivíduos, são também válidas na presente abordagem. Uma das questões mais relevantes na evolução conjunta de pesos e arquiteturas é a escolha dos operadores genéticos. Sabemos que as redes neurais são estruturas que armazenam o conhecimento aprendido de forma distribuída, através de seus pesos. E, sob este ponto de vista, recombinar partes de uma rede neural com partes de outra implica a descaracterização da funcionalidade de ambas. Assim, muitas abordagens evitam a utilização do cruzamento genético e têm adotado apenas a mutação como operador para gerar novas populações de indivíduos.

Em Yao e Liu et al. [29] é proposto um sistema automático, denominado EPNet, capaz de evoluir simultaneamente os pesos das conexões e a arquitetura de uma RNA. Nesta abordagem, novos indivíduos são gerados utilizando uma série de operadores de mutação que modificam os valores dos pesos e a forma da arquitetura da rede. Os operadores são utilizados para aumentar ou diminuir o número de neurônios e das conexões escondidas na rede. Por outro lado, um operador de treinamento híbrido, que utiliza um algoritmo de retro-propagação modificado (MBP), tem a função de modificar os valores dos pesos sinápticos na rede.

Outra abordagem interessante é encontrada em [52]. Neste caso, a codificação utilizada (Figura 2.11) apresenta duas partes: a primeira contém a representação indireta do padrão de conectividade utilizando um vetor de bits de ativação, enquanto que a segunda contém a representação direta dos pesos a partir de um vetor de números reais. A primeira parte influi na avaliação da segunda, de modo que, se uma dada conexão é considerada inexistente (bit em 0), seu peso codificado, apesar de permanecer representado no cromossomo, não é utilizado na avaliação da sua aptidão.

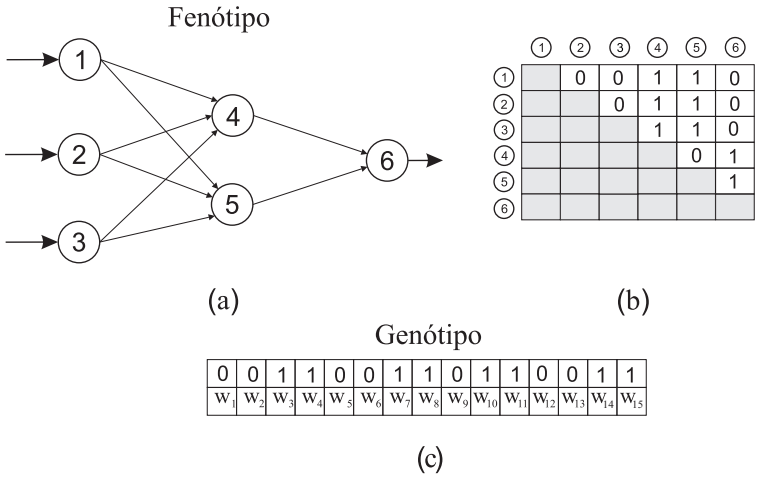


Figura 2.11: (a) Rede neural *feedforward*; (b) Matriz de conexões entre neurônios; (c) Cromossomo, contendo os vetores de conectividade e de pesos.

Abordagens que levam em conta apenas o operador de mutação nem sempre são uma regra geral. Por exemplo, em [65], os operadores de cruzamento agem a partir do seguinte procedimento: se a conexão existir nos dois pais, esta é transmitida para os filhos; caso a conexão exista em apenas um dos pais, é passada para o indivíduo filho com uma dada probabilidade especificada pelo usuário. Uma certa fração do peso da conexão é transmitida para o filho, sendo esta taxa também definida pelo usuário.

Contudo, cada abordagem possui vantagens e desvantagens, as quais podem ser relevantes dependendo da natureza do problema a ser resolvido.

Nas seções anteriores foram expostas algumas abordagens que servem de base para compor o *framework* proposto na presente dissertação. Com esse objetivo foi apresentado uma visão geral da teoria de AGs e de como estes algoritmos têm sido utilizados na otimização de RNAs para o ajuste de pesos, a definição da topologia e inclusive o ajuste simultâneo de pesos e arquiteturas da rede, a partir de técnicas híbridas que utilizam adicionalmente algoritmos de busca local mais eficientes, para o ajuste refinado dos pesos das conexões.

Antes de apresentar formalmente o modelo NEW-GA, torna-se necessário lembrar que, no sistema NEW, a avaliação de cada rede MLP tem associadas duas métricas de desempenho sobre o conjunto de validação (Subseção 2.2.2), sendo estas métricas diretamente influentes na capacidade final de generalização da rede. Este fato leva a considerar uma abordagem de otimização, na qual o grau de aptidão de cada indivíduo, considere as duas métricas de desempenho associadas a cada rede.

Na próxima seção apresenta-se, de forma breve, uma extensão dos AGs

aplicados a problemas de otimização multiobjetivo, pois estes tornam-se uma alternativa atraente para a abordagem proposta, uma vez que o processo de otimização pode ser guiado pela minimização simultânea das duas métricas de erro associadas a cada modelo de rede MLP.

2.5 Algoritmos Evolutivos para Otimização Multiobjetivo

Muitos problemas de tomada de decisão no mundo real possuem a necessidade de otimização simultânea de múltiplos objetivos. Para esses problemas, levar a cabo o processo de otimização, observando um único objetivo, torna-se uma abordagem insuficiente para encontrar soluções satisfatórias visto que grande parte desses problemas apresentam uma coleção de objetivos a serem otimizados. Esses objetivos nem sempre são harmônicos, podendo haver conflitos entre eles, e consequentemente, a melhoria de um implica na deterioração do outro.

Em tais problemas de otimização a qualidade ou aptidão da solução é definida com base na sua adequação em relação aos diversos objetivos possivelmente conflitantes [53, 54]. Na prática, os métodos de solução tradicional buscam reduzir esses problemas a outros com apenas um objetivo. Neste contexto, uma classe de métodos bastante utilizados é dos *métodos baseados em pesos,* onde é criada uma função objetivo, somando cada objetivo, previamente multiplicado por um determinado peso.

Esses métodos, no entanto, são dependentes da escolha adequada de pesos, o que, em muitos casos, implica um conhecimento prévio dos intervalos correspondentes aos pesos mais adequados. Neste sentido, métodos que tentam encontrar soluções que apresentam um compromisso com os vários objetivos sem a necessidade de utilizar pesos passaram a ser explorados [54]. Nesses problemas, no caso geral, não existe somente uma solução única para o problema, mas sim um conjunto de soluções ótimas, denominado *conjunto Pareto ótimo* [53]. A Figura 2.12 mostra vários exemplos de conjuntos Pareto ótimos mapeados no espaço de objetivos, conforme várias combinações de maximização/minimização de duas funções f_1 e f_2 são consideradas. A curva em negrito indica a localização do conjunto Pareto ótimo. Observa-se também que é possível ter mapeamentos de conjuntos formados por uma região contínua ou pela união de regiões descontínuas.

Schafer [66], implementou um dos primeiros AGs para otimização multiobjetivo chamado VEGA (*Vector Evaluated Genetic Algorithm*), sendo este uma extensão do GENESIS, programa para incluir funções multi-critério. Um dos problemas do algoritmo proposto por Schaffer é que as soluções obtidas, em geral, possuem baixa diversidade.

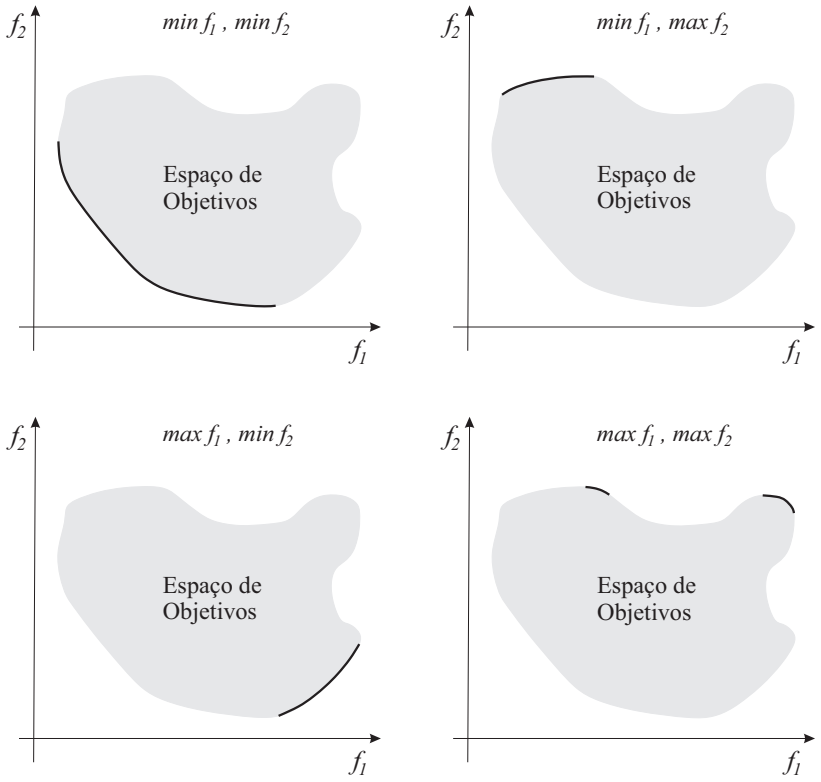


Figura 2.12: Vários exemplos de conjuntos Pareto ótimos.

Goldberg [26] por sua parte, criou um procedimento que ordena as soluções baseado no conceito de *dominância,* sendo este uma medida de aptidão proporcional ao número de soluções que esta domina.

O conceito de dominância é empregado para comparar duas soluções factíveis do problema (definição 1).

Definição 1 *Dadas duas soluções x e y, diz-se que a solução x domina a solução y, denotado como $x \prec y$, se $\forall i \in \{1, \ldots, m\} : f_i(x) \leq f_i(y)$ e $\exists i \in \{1, \ldots, m\} : f_i(x) \neq f_i(y)$.*

Na definição 1, o valor de m faz referência ao número de funções objetivo contempladas no problema específico de otimização (neste caso minimização). Assim, no caso de duas funções objetivo, existe um conjunto de soluções que possuem vantagens com respeito a um dos objetivos mas que não são melhores com respeito a outro e vice-versa. Ou seja, existe um conjunto de soluções ótimas que são *não dominadas* entre si. A fim de ilustrar melhor este procedimento, a Figura 2.13 mostra um conjunto de vetores objetivos (soluções do problema mapeadas no espaço de objetivos), onde os pontos A e B dominam C, os pontos E e F são dominados por C e os pontos D e G não apresentam relação de dominância com respeito a C.

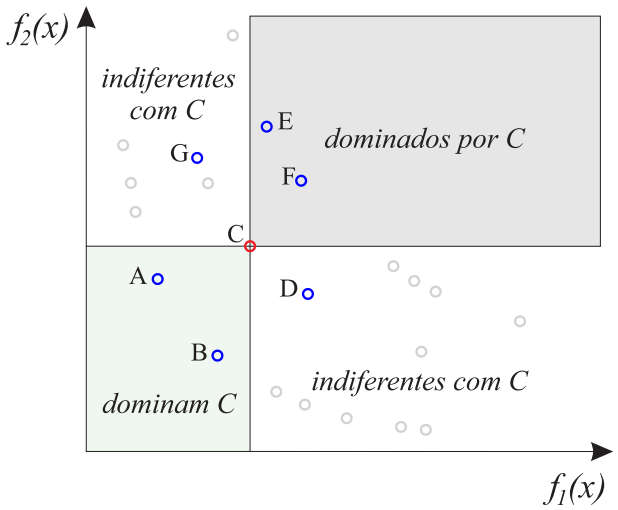


Figura 2.13: Ilustração dos conceitos de dominância em um problema de minimização com dois objetivos.

Em problemas de otimização multiobjetivo, o conjunto de soluções não-dominadas é chamado de conjunto Pareto ótimo, enquanto que a *fronteira de Pareto* corresponde ao conjunto de valores das funções objetivo quando cada solução formando parte do conjunto Pareto ótimo é avaliada (Figura 2.14).

Na Figura 2.14 é ilustrado o espaço de decisão (onde serão aplicados os diferentes operadores genéticos sobre as soluções previamente codificadas), e o espaço de objetivos (onde o grau da aptidão de cada solução é definida, a partir do conceito de dominância) de um problema de minimização com dois objetivos. Observa-se, também, que a imagem de X^*, denominada *Espaço Objetivo Factível*, é denotada por $Y^* = \{(f_1(x),\ f_2(x),\ ...,\ f_n(x),\ x \in X^*\}$, onde o valor de n corresponde ao número de funções objetivo consideradas no problema.

A principal diferença entre os AGs tradicionais e os AGs para otimização multiobjetivo é o operador de seleção, dado que a comparação entre duas soluções deve realizar-se de acordo com o conceito de dominância.

Em [67] é proposta uma abordagem denominada Algoritmo Genético de Classificação por Não Dominância (*Non-Dominated Sorting Genetic Algorithm* - NSGA), a qual está baseada na classificação dos indivíduos em diferentes grupos, denominados fronteiras. Esse processo de agrupamento é realizado com base na *não–dominância* dos indivíduos.

No ano 2000 o NSGA foi estendido surgindo o NSGA-II [68]. O objetivo do NSGA-II é diminuir a complexidade computacional no processo de classificação por *não dominância*, além de introduzir o elitismo e eliminar a

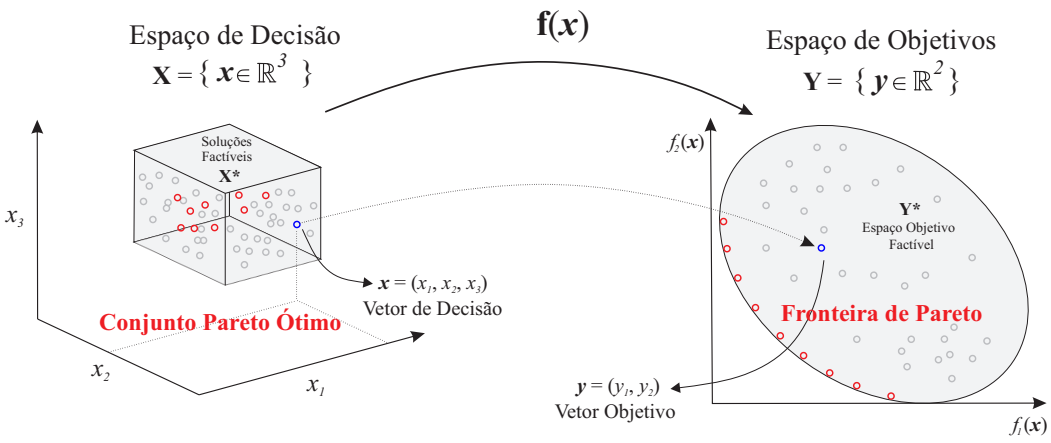


Figura 2.14: Espaço de decisão e espaço objetivo com os respectivos *conjunto Pareto-ótimo* e *Fronteira de Pareto*, para um problema de minimização com dois objetivos.

subjetividade na atribuição do parâmetro de compartilhamento, aplicando um processo conhecido como Distância de Agrupamento (*Crowding Distance Assignment*), garantindo assim a diversidade da população.

2.5.1 Algoritmo Genético por não Dominância II (*Non-Dominated Sorting Genetic Algorithm* - NSGA-II)

O NSGA-II, da mesma forma que seu predecessor o NSGA, implementa o conceito de dominância, classificando os indivíduos da população em fronteiras conforme o grau de dominância. Os melhores indivíduos de cada geração ficam localizados na primeira fronteira (não dominados) ao passo que os piores indivíduos na última fronteira [68].

Um das caraterísticas que diferençam o NSGA-II de um AG simples é a forma como o operador de seleção é aplicado, sendo este subdividido em dois processos: Classificação Rápida por Não Dominância (*Fast Non-Dominated Sorting*) e a Distância de Agrupamento (*Crowding Distance*). Os demais operadores são aplicados de maneira tradicional [68].

O processo de Classificação Rápida por Não Dominância, exemplificado na Figura 2.15, é feito em 2 partes: inicialmente todos os indivíduos da população (novos e antigos) são classificados entre si para determinação do grau de dominância e consequentemente classificados. Ao término da primeira parte, os indivíduos que possuírem o grau de dominância igual a zero (não dominados), são inseridos na primeira fronteira (Ranqueamento 1) [68].

A segunda parte do processo irá tratar os indivíduos cujo grau de

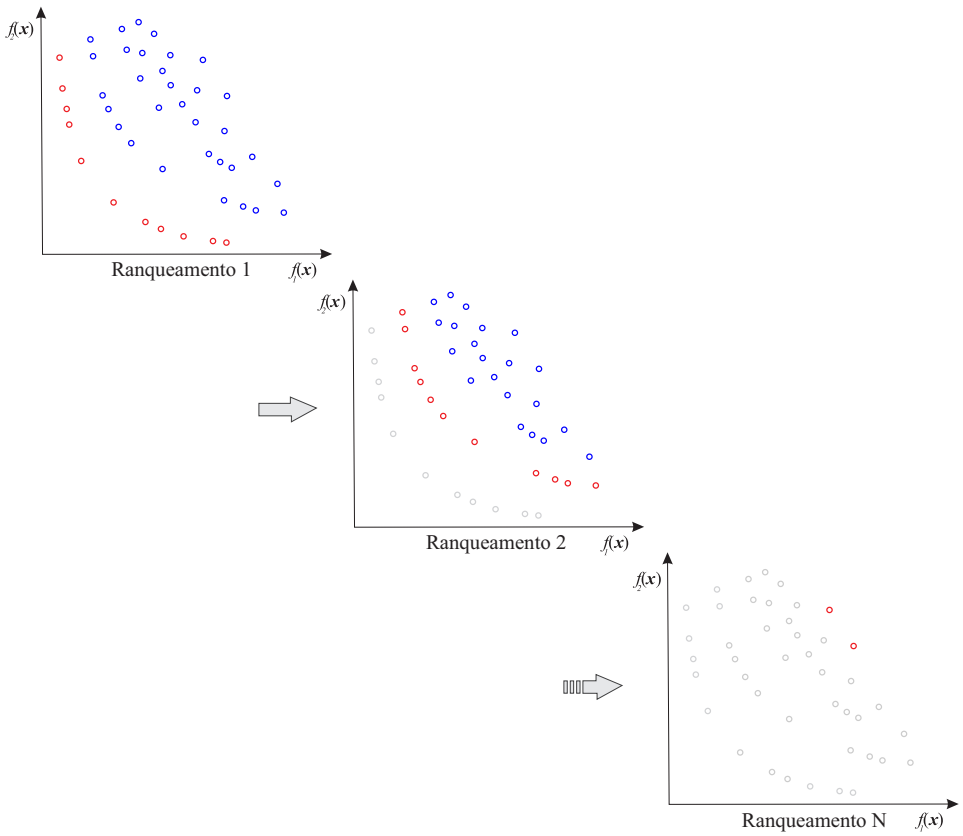


Figura 2.15: Ordenamento das soluções pelo critério de não dominância em um problema de minimização com dois objetivos.

dominância é diferente de zero. Excluem-se da população os indivíduos de dominância zero e recalcula–se a dominância dos demais indivíduos. Após o novo cálculo os indivíduos de dominância zero são inseridos na próxima fronteira (Ranqueamento 2), assim por diante para todas as fronteiras. O processo é repetido até que a população esteja vazia [68].

O processo de busca por soluções Pareto ótimas tende a convergir para uma mesma região do espaço de busca. Em AG multiobjetivo uma caraterística desejada é que as soluções encontradas estejam espalhadas nesta região. Para tal fim, é aplicado o segundo processo do operador de seleção, a Distância de Agrupamento.

A Distância de Agrupamento é uma abordagem baseada na comparação de aglomerado, proposta para substituir a abordagem da função de compartilhamento no primeira versão do NSGA. Outra função da Distância de Agrupamento é estabelecer uma ordem parcial das soluções dentro de uma mesma fronteira [68].

Com o fim de compreender melhor a abordagem de Distância de Agrupamento, é necessário definir a métrica para estimação de densidade e o operador

de comparação.

A estimação de densidade de soluções que cercam uma solução particular na população é obtida através do cálculo da distância média entre a solução anterior e a posterior da solução particular, ao longo de cada um dos objetivos. Esta métrica serve como uma estimativa do perímetro do cuboide formado usando os vizinhos mais próximos da solução, como os vértices, conforme a Figura 2.16.

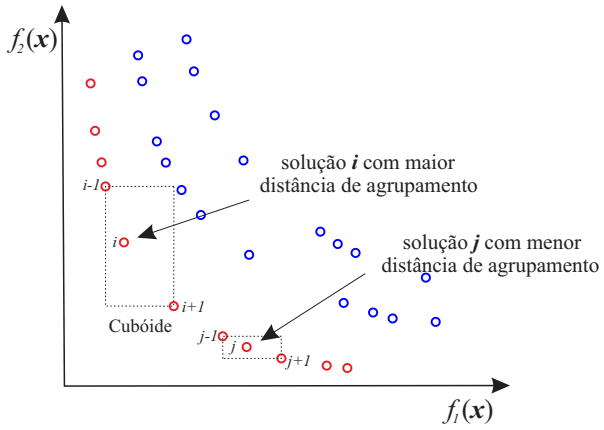


Figura 2.16: Cálculo da Distância de Agrupamento em um problema de minimização com dois objetivos. Pontos marcados em círculos vermelhos são soluções da fronteira não dominada.

Desta forma, o operador de comparação ($\prec_n$) tem o objetivo de orientar o processo de seleção nas várias fases do algoritmo em direção a uma fronteira de Pareto ótima uniformemente espalhada. Supondo que cada indivíduo na população tem dois atributos:

1. Ranqueamento de não dominância (i_{ranq});
2. Distância de Agrupamento (i_{distance}).

Uma ordem parcial $\prec_n$ é definida por:

$$i \prec_{\text{n}} j \text{ se } (i_{\text{ranq}} < j_{\text{ranq}}) \tag{2-30}$$

$$\text{ou } (i_{\text{ranq}} = j_{\text{ranq}}) \text{ e } (i_{\text{distance}} > j_{\text{distance}})$$

Deste modo, para duas soluções localizadas em diferentes fronteiras não dominadas, este modelo dá preferência à escolha da solução com menor

ranqueamento, caso contrário, é escolhida a solução localizada em uma região de menor aglomeração [68].

Uma vez finalizado o processo de classificação, os indivíduos são selecionados para compor a próxima geração a partir dos pertencentes à primeira fronteira, até completar a nova população. Se uma fronteira não puder ser totalmente inserida na população seguinte, o algoritmo utiliza como critério a distância de agrupamento para escolher quais indivíduos serão selecionados. A Figura 2.17 exemplifica o processo descrito no NSGA-II.

Assim, a partir da nova população P_{t+1}, é criada uma população de descendentes Q_{t+1} por meio de operadores genéticos de seleção, recombinação e mutação. O processo é então repetido até que o critério de parada seja satisfeito.

Com base nos fundamentos teóricos expostos, o próximo capítulo apresenta formalmente o modelo NEW-GA, sendo este uma nova abordagem para otimização simultânea de arquiteturas e pesos em modelos de redes MLP, aplicadas ao núcleo do sistema NEW, o qual faz uso de uma técnica de treinamento híbrida que integra a meta-heurística evolutiva de otimização multiobjetivo, *NSGA-II,* junto com o algoritmo de busca local baseado no gradiente *backpropagation.*

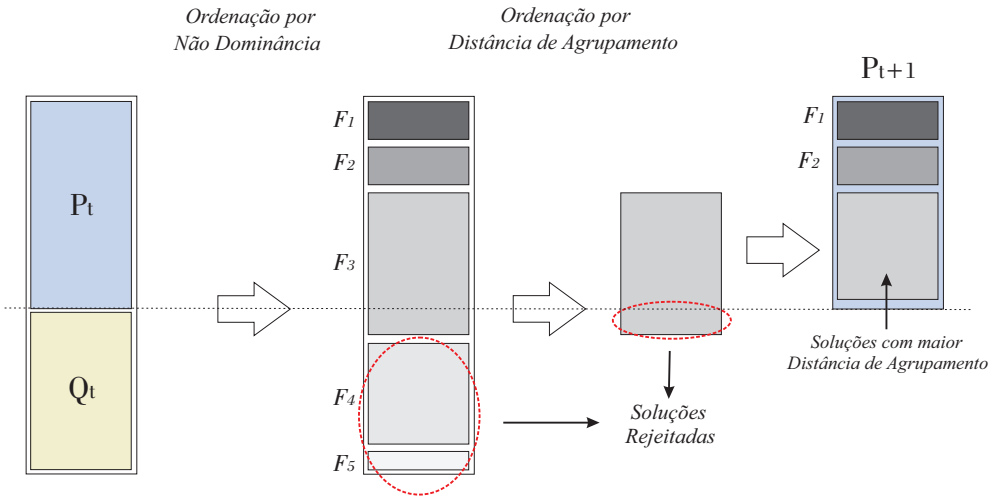


Figura 2.17: Procedimento do NSGA-II, baseado em [53].

3
Modelo NEW-GA

Como foi mencionado anteriormente, este trabalho propõe uma metodologia para a otimização simultânea de arquiteturas e pesos em redes MLP aplicadas ao núcleo do sistema NEW [12]. Segundo esta abordagem, cada ponto ou solução factível do espaço de busca, codifica tanto a estrutura topológica da rede quanto os seus valores de pesos, possuindo cada solução uma medida de aptidão estimada a partir do conceito de dominância abordado no capítulo anterior.

O objetivo deste capítulo é detalhar a metodologia proposta, que combina caraterísticas do algoritmo *NSGA-II*, junto com o algoritmo de busca local *backpropagation* no processo de otimização da rede. São descritas as etapas de construção, desde a codificação das soluções em estruturas que podem ser manipuladas pelos operadores genéticos, até o critério para selecionar uma única solução a partir do conjunto de soluções não dominadas (*Fronteira de Pareto*), fornecido na etapa final do algoritmo de otimização aqui proposto. A Seção 3.1 apresenta as definições e notações utilizadas nas seções subsequentes. A Seção 3.2, por sua vez, descreve a arquitetura do modelo *NEW-GA*, enquanto que na Seção 3.3 é apresentado um resumo do presente capítulo.

3.1 Definições e Notações

No modelo *NEW-GA*, os conjuntos de dados utilizados durante o processo de otimização da rede MLP são criados da mesma forma que no modelo *NEW* (Seção 2.2): a partir de uma série temporal e da definição de uma janela de tempo fixa ou expansiva, são geradas as *entradas* com os previsores *Holt-Winters* e *Box & Jankins,* assim como as *saídas* ou vetores de ponderação, a partir do método *MQR.*

Para exemplificar melhor a formação destes conjuntos (Figura 3.1), considera-se a série original $y^{\tau+H} = [y_1,\ y_2,\ \cdots,\ y_\tau,\ \cdots,\ y_{\tau+H}]'$, com $\tau + H$ observações e composta pelos trechos individuais chamados de:

1. *Série de treinamento* ou *série histórica* $y^\tau = [y_1,\ y_2,\ \cdots,\ y_\tau]'$, contendo as primeiras τ observações da série original e;

2. *Série de teste* $y^{\tau+H|\tau} = [y_{\tau+1},\ y_{\tau+2},\ \cdots,\ y_{\tau+H}]'$, formada pelas H observações mais recentes da série original.

Assim, para cada observação no instante de tempo t da *série histórica,* estima-se um vetor de N previsões $\hat{Y}_{t|t-h} = [\hat{y}_{t|t-h,1},\ \hat{y}_{t|t-h,2},\ \cdots,\ \hat{y}_{t|t-h,N}]$, a partir dos previsores *Holt-Winters* e *Box & Jankins*, calculado com informações até o instante $t-h$ da série histórica, onde $h \leq H < t \leq \tau$. Analogamente, para cada vetor de previsões, estima-se um vetor de N pesos convexos $\hat{W}_{t|t-h} = [\hat{w}_{t|t-h,1},\ \hat{w}_{t|t-h,2},\ \cdots,\ \hat{w}_{t|t-h,N}]$, a partir do método *MQR*, os quais ponderam linearmente as previsões disponíveis naquele instante de tempo t da *série histórica*.

Levando em conta as definições anteriores, a equação (3-1), da mesma forma que no modelo *NEW* original, representa os conjuntos de entradas e saídas utilizados durante o processo de otimização da rede MLP.

$$\left\langle \left\{ \hat{Y}_{t|t-h}, h \right\} \mid \hat{W}_{t|t-h} \right\rangle \tag{3-1}$$

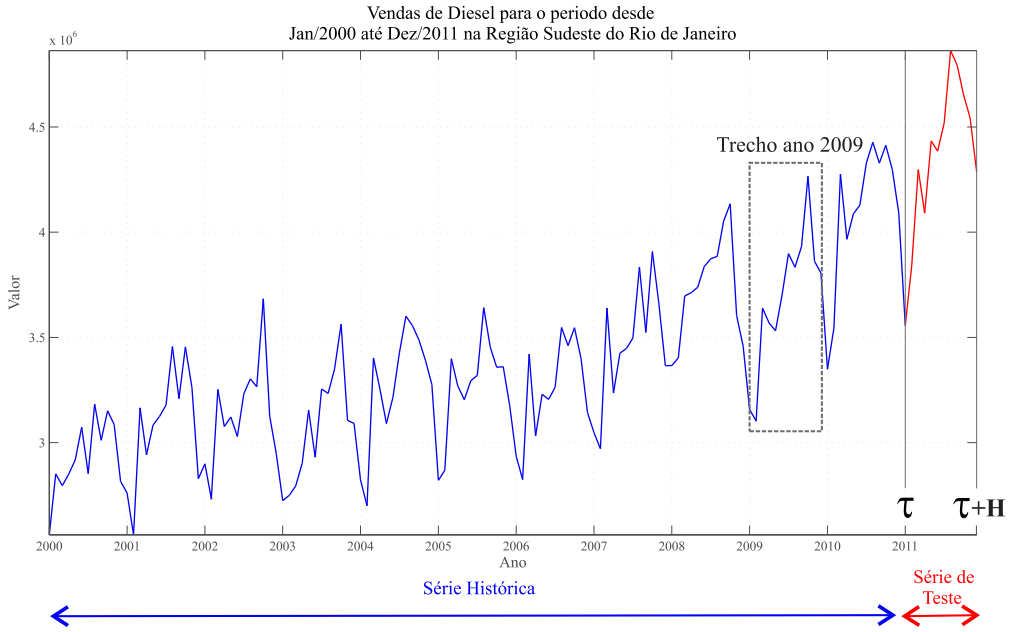
Onde, o vetor de entradas, obtido a partir dos previsores *Holt-Winters* e *Box & Jankins*, e o vetor de saídas, estimado pelo método *MQR*, são dados pelas equações (3-2) e (3-3) respetivamente.

$$\left\{ \hat{Y}_{t|t-h}, h \right\} = \begin{bmatrix} \hat{y}_{t|t-h,1} \\ \hat{y}_{t|t-h,2} \\ \vdots \\ \hat{y}_{t|t-h,N} \\ h \end{bmatrix} \tag{3-2}$$

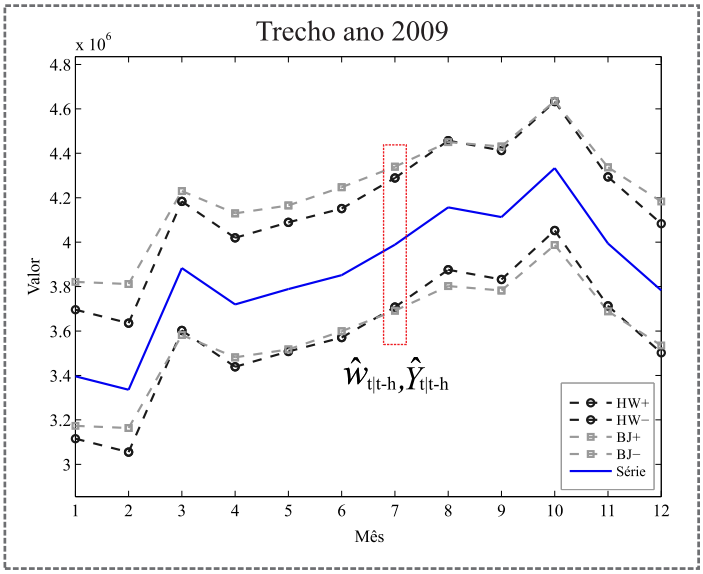
$$\hat{W}_{t|t-h}(\nu) = \begin{bmatrix} \hat{w}_{t|t-h,1} \\ \hat{w}_{t|t-h,2} \\ \vdots \\ \hat{w}_{t|t-h,N} \end{bmatrix} \tag{3-3}$$

$$\text{sujeito a } h \leq H < t \leq \tau\,. \tag{3-4}$$

A equação (3-3) está sujeita também às restrições de convexidade dadas por (2-9), abordadas na Seção 2.1.2 do Capítulo 2.



3.1(a): Série original



3.1(b): Previsões limiares *HW* e *BJ* para um trecho especifico da *série histórica*

Figura 3.1: Associação dos vetores de previsão e de ponderação para cada observação da série histórica. Neste casso, cada observação (mês) da série tem associado um vetor de 4 previsões e um vetor de 4 pesos.

Estes conjuntos de entradas e saídas são, por sua vez, divididos em 2 conjuntos chamados de *Treinamento e Validação*, os quais estão associados a diferentes trechos da *série histórica* (Figura 3.2), sendo os mesmos utilizados

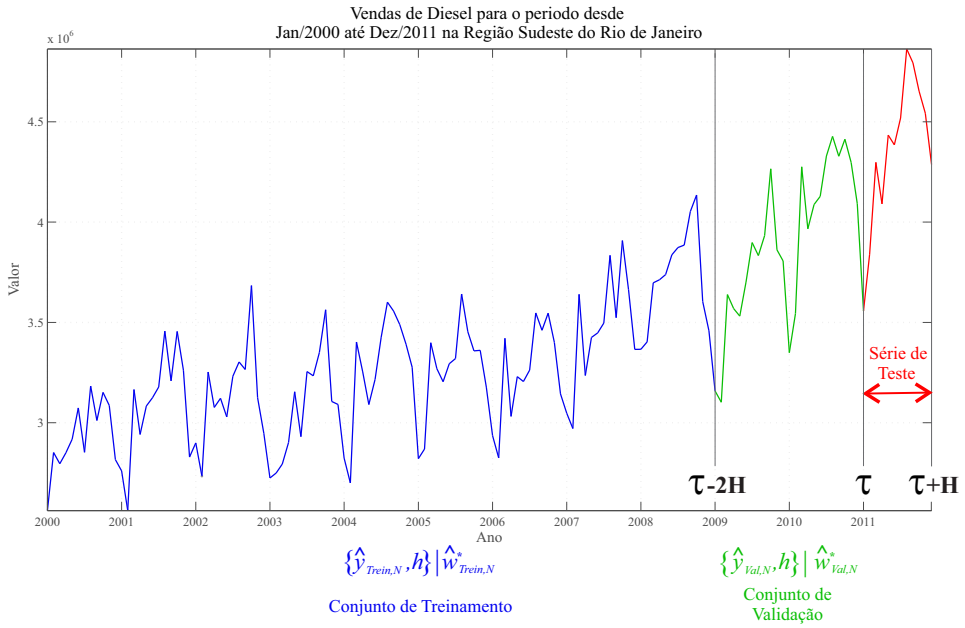


Figura 3.2: Associação dos conjuntos de *Treinamento* e *Validação* para os diferentes trechos da série histórica.

de forma alternada, durante distintas etapas no processo de otimização da rede.

A seguir apresenta-se as equações associadas a cada conjunto, baseadas na conformação de conjuntos proposta em [12].

– Conjunto de Treinamento:

$$\left\langle \left\{ \hat{Y}_{t|t-h}, h \right\} \mid \hat{W}_{t|t-h} \right\rangle \rightarrow \{\hat{y}_{\text{Trein,N}}, h\} \mid \hat{W^*}_{\text{Trein,N}} \tag{3-5}$$

– Conjunto de Validação:

$$\left\langle \left\{ \hat{Y}_{t+h|t}, h \right\} \mid \hat{W}_{t+h|t} \right\rangle \rightarrow \{\hat{y}_{\text{Val,N}}, h\} \mid \hat{W^*}_{\text{Val,N}} \tag{3-6}$$

Destaca-se que, no modelo proposto, os vetores de ponderação reais, estimados a partir do método *MQR*, são representados pela letra $\hat{W}^*$. $\hat{W^*}_{\text{Trein,N}}$ representa o conjunto real utilizado durante o ajuste parcial dos pesos sinápticos da rede a partir da aplicação de um dos operadores genéticos propostos, chamado de *operador de busca local* (Seção 3.2). $\hat{W^*}_{\text{Val,N}}$, por sua vez, representa o conjunto real utilizado na avaliação de uma das métricas de desempenho que direciona o processo evolutivo, a qual leva em conta a estimação dos erros com relação à resposta estimada pela rede, ou seja, o erro da rede na inferência dos vetores de ponderação, os quais são denotados por $\hat{W}_{\text{Val,N}}$.

É importante assinalar que esta segunda métrica de desempenho se encarrega de direcionar o processo de otimização, a partir dos vetores de ponderação $\hat{W}_{\text{Val,N}}$ gerados pela rede, realizando a combinação dos correspondentes vetores

de previsões $\hat{y}_{\text{Val,N}}$ e avaliando o erro médio entre o vetor de previsão combinado y^{C}_{Val} obtido, e o trecho da série histórica correspondente ao conjunto de validação y_{Val} (trecho verde na Figura 3.2). Deste modo, cada rede MLP, ou indivíduo da população, terá duas métricas de desempenho sobre o conjunto de validação, as quais correspondem às funções objetivo a serem minimizadas durante o processo de otimização multiobjetivo proposto. Os valores de $\hat{y}$ nos conjuntos de treinamento e validação são normalizados conjuntamente entre -1 e 1, a fim de evitar um comportamento de saturação nas funções de ativação dos neurônios da rede MLP.

Por último, é importante mencionar que para cada observação da *série de teste* estima-se também um vetor de previsões a partir dos modelos *Holt-Winters* e *Box & Jankins*, sendo o conjunto destes vetores representado pela seguinte equação:

$$\hat{Y}_{\tau+h|\tau} \rightarrow \hat{y}_{\text{Teste,N}} \tag{3-7}$$

Durante a etapa de avaliação final do modelo, cada vetor de previsão, associado à *série de teste*, é ponderado pelos vetores de pesos estimados a partir do modelo representado pela equação (3-8), obtendo, desta forma, um vetor de previsão combinada y^{C}_{Teste} que é utilizado para estimar o erro médio com relação às observações reais da *série de teste,* sendo esta métrica utilizada para comparar o desempenho do modelo proposto com respeito ao modelo *NEW* original, assim como também com os modelos individuais de previsão e os modelos tradicionais de ponderação abordados na Seção 2.1

$$\hat{W}_{\tau+h|\tau} = G\left(\hat{Y}_{\tau+h|\tau}, h\right) \tag{3-8}$$

$$\text{sujeito a } h \leq H\,,$$

$$\sum_{k=1}^{N} \hat{W}_{\tau+h|\tau,k} = 1 \; e \; \hat{W}_{\tau+h|\tau,k} \geq 0\,. \tag{3-9}$$

No modelo da equação (3-8), G representa a rede neural MLP propriamente ajustada e otimizada pelo modelo proposto, onde h corresponde ao horizonte de previsão, garantindo a combinação convexa da equação (3-9).

3.2 Arquitetura Básica

Este tópico apresenta o modelo NEW-GA. Cada uma das seções subsequentes tratará de uma etapa de modelagem: representação e formação das soluções; inicialização e avaliação da população; descrição dos operadores genéticos (seleção, cruzamento, mutação e busca local); avaliação do critério de parada; e, finalmente, o mecanismo de escolha da solução ótima a partir do conjunto de soluções Pareto ótimo fornecido pelo algoritmo. A Figura 3.3 ex-

põe de forma macro os passos para o ajuste do modelo NEW-GA, enquanto que a seguir apresenta-se o pseudocódigo do modelo.

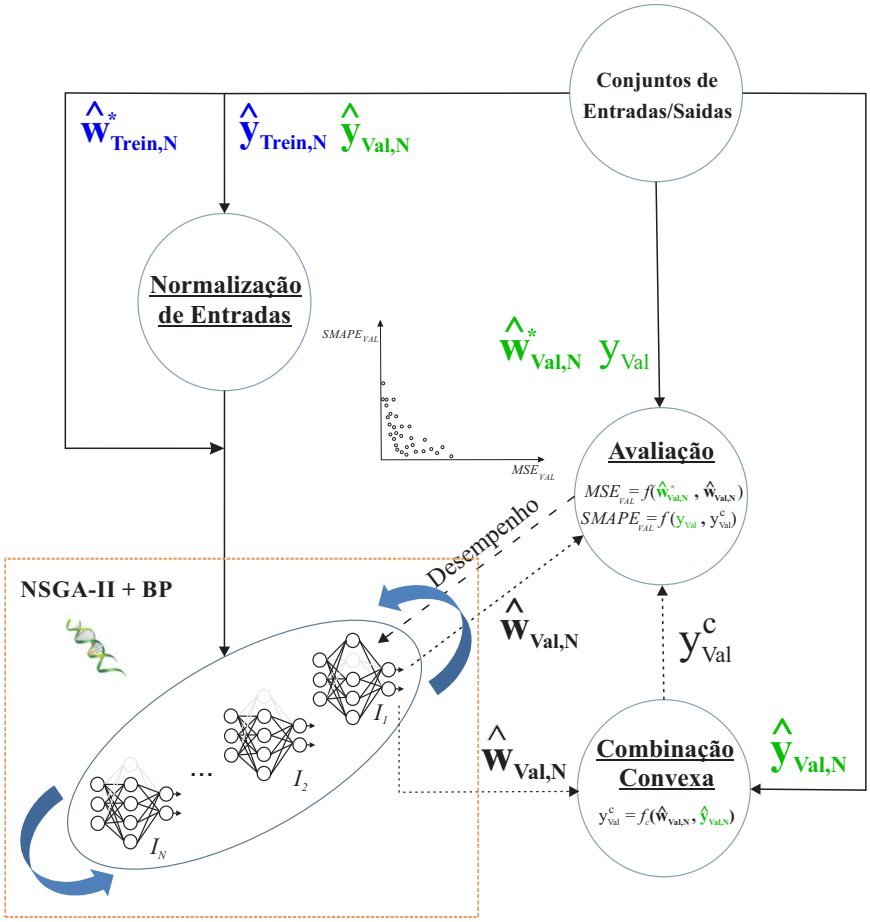


Figura 3.3: Diagrama genérico das etapas de elaboração do modelo NEW-GA.

Como o modelo NEW-GA utiliza uma técnica de treinamento híbrida, o seu mecanismo está baseado na mistura entre um algoritmo de busca global (NSGA-II), encarregado de localizar boas regiões no espaço de busca, e um algoritmo de busca local, baseado no gradiente (Backpropagation), que fica com o papel de identificar o ponto de mínimo destas regiões. Neste sentido, o operador genético, chamado de *operador de busca local* é aplicado a cada certo número de gerações com o objetivo de percorrer localmente o espaço de busca.

Algoritmo 2 Pseudocódigo do Modelo NEW-GA.

1: Passos Iniciais: - Adequação dos conjuntos de entrada/saída - Definição dos parâmetros internos do algoritmo
2: Inicialização da população P_0 e cálculo das funções objetivo de cada individuo
3: Ordenamento da população pelo *Critério de Não Dominância*
4: Geração de Q_0 a partir de P_0 por meio dos operadores genéticos de seleção, cruzamento, mutação e busca local
5: $t \rightarrow 0$
6: **repita**
7: Cálculo das funções objetivo para cada indivíduo de Q_t
8: Combinação de P_t e Q_t para formar a população R_t
9: Ordenamento da população R_t pelo *Critério de Não Dominância*
10: Criação da população P_{t+1} a partir de R_t levando em conta o *Ranqueamento de Não Dominância* e a *Distância de Agrupamento* associada a cada indivíduo
11: Geração de Q_{t+1} a partir de P_{t+1} por meio dos operadores genéticos de seleção, cruzamento, mutação e busca local
12: $t \rightarrow t + 1$
13: **até** Critério de parada satisfeito;
14: Retornar a *Fronteira Pareto ótimo*
15: Seleção da solução ótima a partir do conjunto de soluções *Pareto ótimo*

3.2.1 Representação e Formação das Soluções

No modelo proposto, as topologias MLP possuem uma única camada escondida, contendo todas as conexões possíveis entre camadas adjacentes sem haver conexões entre camadas não-adjacentes nem conexões recorrentes pois o tipo de redes utilizadas apresentam uma arquitetura do tipo *feedforward*.

Para a codificação da rede, o usuário precisa definir o *número máximo de neurônios ocultos* (N_{max}) das redes a evoluir. O número de neurônios das camadas de entrada e de saída são previamente estabelecidos em função das dimensões dos próprios conjuntos de treinamento (dimensão dos vetores de previsão e dos vetores de ponderação respectivamente). Em consequência, define-se o *número máximo de conexões* de todas as redes neurais candidatas (N_{Cmax}) como:

$$N_{Cmax} = (N_1 + 1) * N_{max} + (N_{max} + 1) * N_3 \quad (3\text{-}10)$$

Onde, N_1 corresponde ao número de neurônios (dependente da dimensão

do vetor de previsores) da camada de entrada, N_3 é o número de neurônios (dependente da dimensão do vetor de pesos de ponderação) da camada de saída, e N_{max} é o número máximo de neurônios (definido pelo projetista da rede) na única camada intermediária. A equação (3-10) também inclui a conexão do *bias*, aplicado externamente sobre cada neurônio, tanto na camada oculta quanto na camada de saída, o qual esta representado por b_k na Figura 3.4. O bias b_k tem o efeito de aumentar ou diminuir a entrada da função de ativação do neurônio, dependendo se ele fosse positivo ou negativo, respetivamente.

O esquema de codificação proposto, inspirado em [69], é uma mistura do esquema binário e real. Cada solução S é codificada por uma dupla de vetores, formada pelas C conexões e pelos P pesos sinápticos da rede. A informação contida no vetor de conexões fornece a existência ou não de uma conexão entre dois neurônios, enquanto que o valor do peso sináptico associado a essa conexão é dado pelo vetor de pesos. A existência da conexão é representada por um bit de conectividade possuindo o valor 1 caso a conexão exista e o valor 0 caso não exista. Já os pesos são representados por números reais ($\mathbb{R}$).

$$S = (C, P) \tag{3-11}$$

$$C = (c_1, c_2, ..., c_{N_{Cmax}}), \quad c_i \in \{0, 1\}, \quad i = 1, 2, ..., N_{Cmax} \tag{3-12}$$

$$P = (p_1, p_2, ..., p_{N_{Cmax}}), \quad p_i \in \mathbb{R}, \quad i = 1, 2, ..., N_{Cmax} \tag{3-13}$$

A Figura 3.4 esclarece a forma como uma arquitetura NEW-GA é representada num cromossomo S com valores binários e reais. O vetor binário C, que representa o padrão de conectividade da rede, é composto pelos valores do triângulo superior da matriz que contém, como número de linhas e colunas, o número total de neurônios na rede ($N_1 + N_2 + N_3$) mais os bias das camadas escondida e de saída. Os valores 0 e 1, nesta parte da matriz, indicam a ausência e a presença da conexão entre os neurônios da linha e da coluna, respetivamente. Ainda na Figura 3.4, mostra-se um exemplo didático para a codificação de uma rede neural, apresentada na parte esquerda da figura, composta por 10 neurônios no total, sendo 3 na camada de entrada, 3 na camada de saída e 4 neurônios ocultos, dos quais apenas 2 encontram-se ativados.

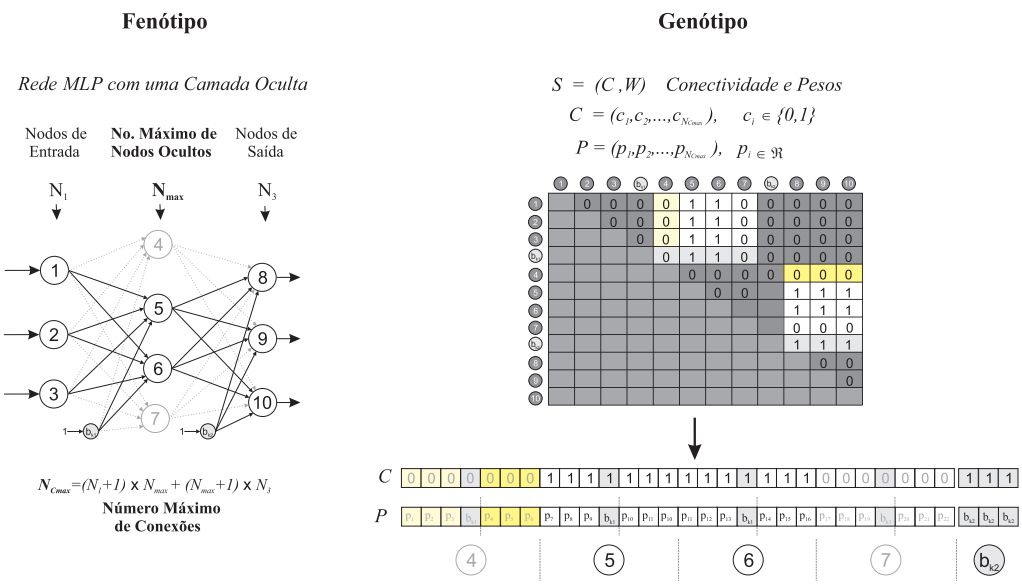


Figura 3.4: Codificação dos vetores de conectividade e pesos que representam o cromossomo para uma arquitetura de rede MLP no modelo NEW-GA.

À direita da Figura 3.4, apresenta-se a matriz com os elementos 0 e 1 da triangular superior e, abaixo desta, o vetor de conectividade (com o respectivo vetor de pesos) como resultado do empilhamento dos valores da triangular superior da matriz numa ordem determinada pela sequência coluna-linha associada às conexões entrada-saída de cada neurônio na camada oculta da rede. Optou-se por este procedimento de empilhamento com o objetivo de que os pesos codificados de um mesmo neurônio fiquem juntos no cromossomo, uma vez estes se comportam como extratores e detetores de caraterísticas, a separação de seus pesos de entrada pode levar à perda deste comportamento quando aplicados operadores de cruzamento [24]. Deve-se salientar que o empilhamento coluna-linha associado a cada neurônio da camada intermediaria, inclui o respetivo bias de entrada (b_{k1}), o qual forma parte do cromossomo que codifica a rede MLP. Por sua parte, as conexões do bias de entrada na camada de saída (b_{k2}), permanecem sempre ativadas durante o processo evolutivo, isto é, não sofrem modificação por parte dos operadores genéticos de mutação (habilitação e desabilitação).

Desta forma, uma vez definido o *número máximo de neurônios ocultos* e conhecendo as dimensões do vetor de previsões, o tamanho do vetor de conectividade assim como do vetor de pesos ficará definido automaticamente.

3.2.2 Inicialização da População

Uma vez definida a topologia máxima das redes a evoluir - determinada pelo *número máximo de neurônios ocultos* (N_{max}) - ativa-se, para cada indivíduo da população, uma quantidade aleatória de neurônios ocultos, obtida a partir de uma distribuição uniforme discreta $U(1,\ N_{max})$ no intervalo $[1, N_{max}]$. Assim, a ativação de cada neurônio é realizada a partir da ativação dos seus correspondentes bits no vetor de conectividade. Desta forma, novas topologias candidatas são construídas habilitando e desabilitando neurônios ocultos, a partir dos operadores genéticos propostos.

Os valores iniciais do vetor de pesos sinápticos, por sua parte, são extraídos aleatoriamente de uma distribuição normal $\mathcal{N}(0,\ \sigma_P)$, com média em 0 e desvio padrão dado pela equação (3-14), onde m corresponde ao número total de entradas na rede [70].

$$\sigma_P = m^{-1/2} \tag{3-14}$$

3.2.3 Avaliação dos Indivíduos (Funções Objetivo)

Outro conceito essencial que necessita ser estabelecido é o vetor de funções objetivo, encarregado de levar a cabo o mapeamento de cada solução no espaço de objetivos sobre o qual será aplicado o operador de seleção, levando em conta a ordem parcial das soluções conforme o seu grau de dominância (Seção 2.5.1).

De forma sucinta, em 3-15 e 3-16 apresenta-se as funções que conformam o vetor de funções objetivo a minimizar durante o processo de otimização da rede.

$$f_1 = MSE_{\text{Val}} = \frac{1}{M}\sum_{M}\left(\hat{W}^{*}{}_{\text{Val,N}} - \hat{W}_{\text{Val,N}}\right)^2 \tag{3-15}$$

$$f_2 = SMAPE_{\text{Val}}(\%) = \frac{1}{M}\sum_{M}\frac{\left|y_{\text{Val}} - y^{C}_{\text{Val}}\right|}{\left(\frac{|y_{\text{Val}}|+\left|y^{C}_{\text{Val}}\right|}{2}\right)}100 \tag{3-16}$$

Onde, M corresponde ao tamanho do conjunto de validação (equação (3-6)). A função representada por (3-15) é uma medida de erro dada pela comparação entre os vetores à saída da rede $\left(\hat{W}_{\text{Val,N}}\right)$ com respeito aos vetores de ponderação esperados $\left(\hat{W}^{*}{}_{\text{Val,N}}\right)$, enquanto que a função em (3-16) é uma medida de erro dada pela comparação entre as observações originais da série (y_{Val}) com respeito ao vetor de previsões combinadas $\left(y^{C}_{\text{Val}}\right)$ obtido a partir da combinação convexa entre os vetores de previsões individuais $(\hat{y}_{\text{Val,N}})$ com os vetores de ponderação à saída da rede $\left(\hat{W}_{\text{Val,N}}\right)$. A Figura 3.5 expõe de forma geral o processo do mapeamento de cada solução sobre o espaço de objetivos.

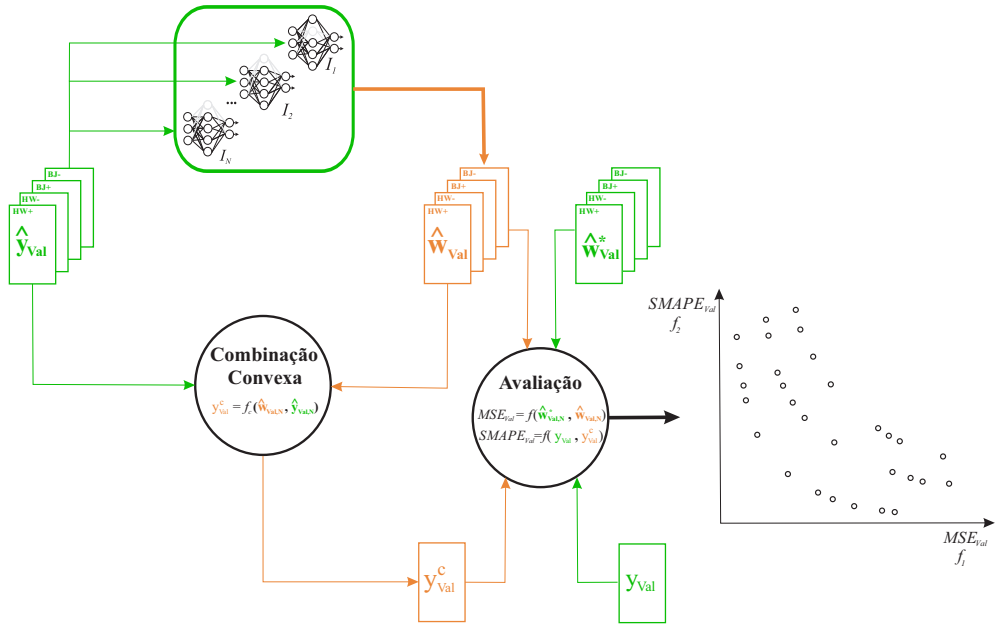


Figura 3.5: Mapeamento das soluções sobre o espaço de objetivos. Os vetores de previsão originais (HW e BJ) são substituídos pelos seus correspondentes previsores limiares (Seção 2.2).

Neste ponto é importante mencionar que o mapeamento de cada solução sobre o espaço de objetivos requer previamente um processo de ajuste dos pesos sinápticos das redes codificadas nos cromossomos que compõem a população de indivíduos. Este processo é feito a partir da execução do algoritmo backpropagation por um número k_1 pequeno de iterações. Em cada iteração, todos os padrões do conjunto de treinamento (equação (3-5)) são apresentados à rede neural.

3.2.4 Operadores Genéticos

3.2.4.1 Operador de Seleção

No modelo proposto é utilizado o *torneio de 2* (k=2) [56], como operador de seleção de progenitores, isto é, dois indivíduos (obtidos aleatoriamente da população) competem entre si e o ganhador (o de maior aptidão) é selecionado para a fase de reprodução.

No caso específico dos modelos NEW-GA, a comparação entre dois indivíduos, realizada sobre o espaço de objetivos, utiliza o operador de comparação ($\prec$) abordado na Seção 2.5.1, o qual leva em conta o *Ranqueamento de Não Dominância* e a *Distância de Agrupamento,* associados a cada indivíduo, para estabelecer uma ordem parcial entre eles.

A Figura 3.6 apresenta uma ilustração do processo de comparação proposto na presente abordagem.

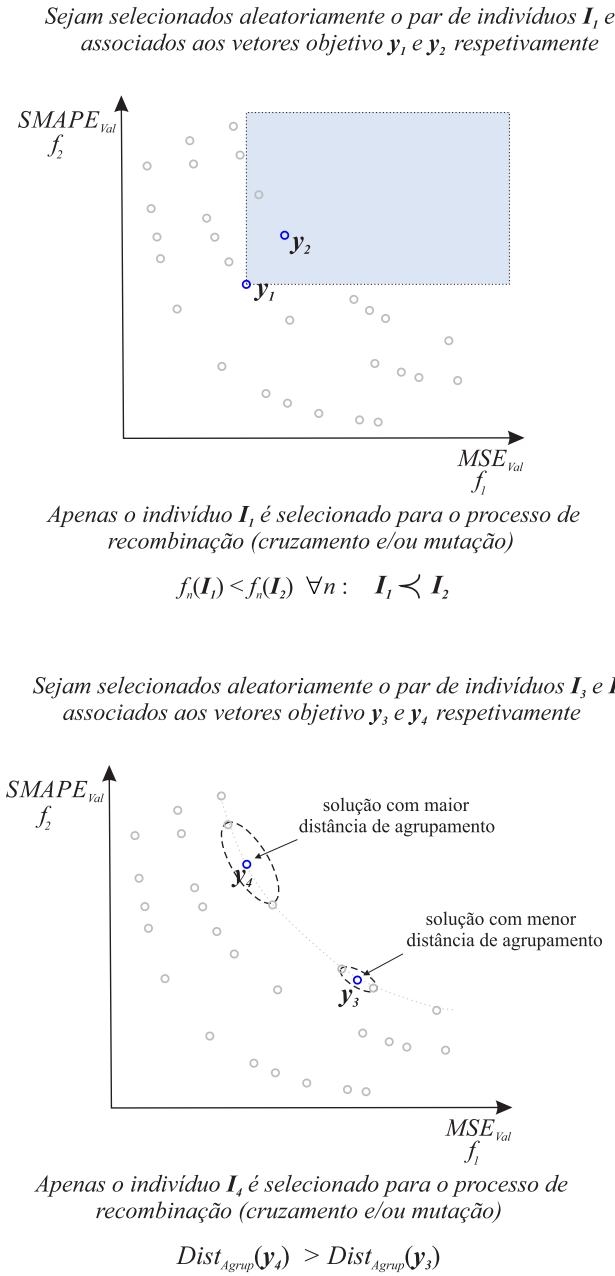


Figura 3.6: Ilustração do operador de seleção usado no modelo NEW-GA.

3.2.4.2 Operador de Cruzamento

Um novo indivíduo filho é produzido por cada par de progenitores via cruzamento entre os correspondentes segmentos de vetores de pesos associados a cada neurônio na camada oculta, onde, dependendo dos estados de ativação

dos neurônios (dados pelos bits no vetor de conectividade), é aplicado um procedimento diferente entre os correspondentes segmentos de vetores de pesos.

Sejam dois indivíduos da população (Pai 1 e Pai 2), selecionados a partir do operador de seleção descrito anteriormente, para cada dupla de neurônios ocupando a mesma posição na camada oculta de cada individuo, temos que:

1. Caso os dois neurônios tenham ativados os seus bits de conectividade, os pesos do respetivo neurônio no indivíduo filho serão obtidos a partir do *Cruzamento Aritmético* entre os pesos associados à dupla dos neurônios ativados:
$$P_{iFilho} = \alpha P_{iPai1} + (1 - \alpha) P_{iPai2} \tag{3-17}$$
O valor de α é fixado de modo a atribuir um maior peso para o indivíduo pai dominante, levando em conta o ranqueamento de não dominância e a distância de agrupamento associada a cada progenitor.

2. Caso só um dos neurônios tenha ativados os seus bits de conectividade, o respectivo neurônio no indivíduo filho será ativado com uma probabilidade proporcional ao *Ranqueamento de Não Dominância,* associado ao indivíduo Pai que apresentar os seus bits de conectividade ativados. Da mesma forma, os valores dos seus pesos são transmitidos para o indivíduo filho.

3. Caso os dois neurônios tenham desativados os seus bits de conectividade, o respectivo neurônio no indivíduo filho será também desativado mas os seus pesos correspondentes serão inicializados da mesma forma que os pesos nos indivíduos da população inicial (Seção 3.2.2).

Com o objetivo de exemplificar este procedimento, a Figura 3.7 ilustra o mecanismo de como o operador de cruzamento atua sobre os indivíduos.

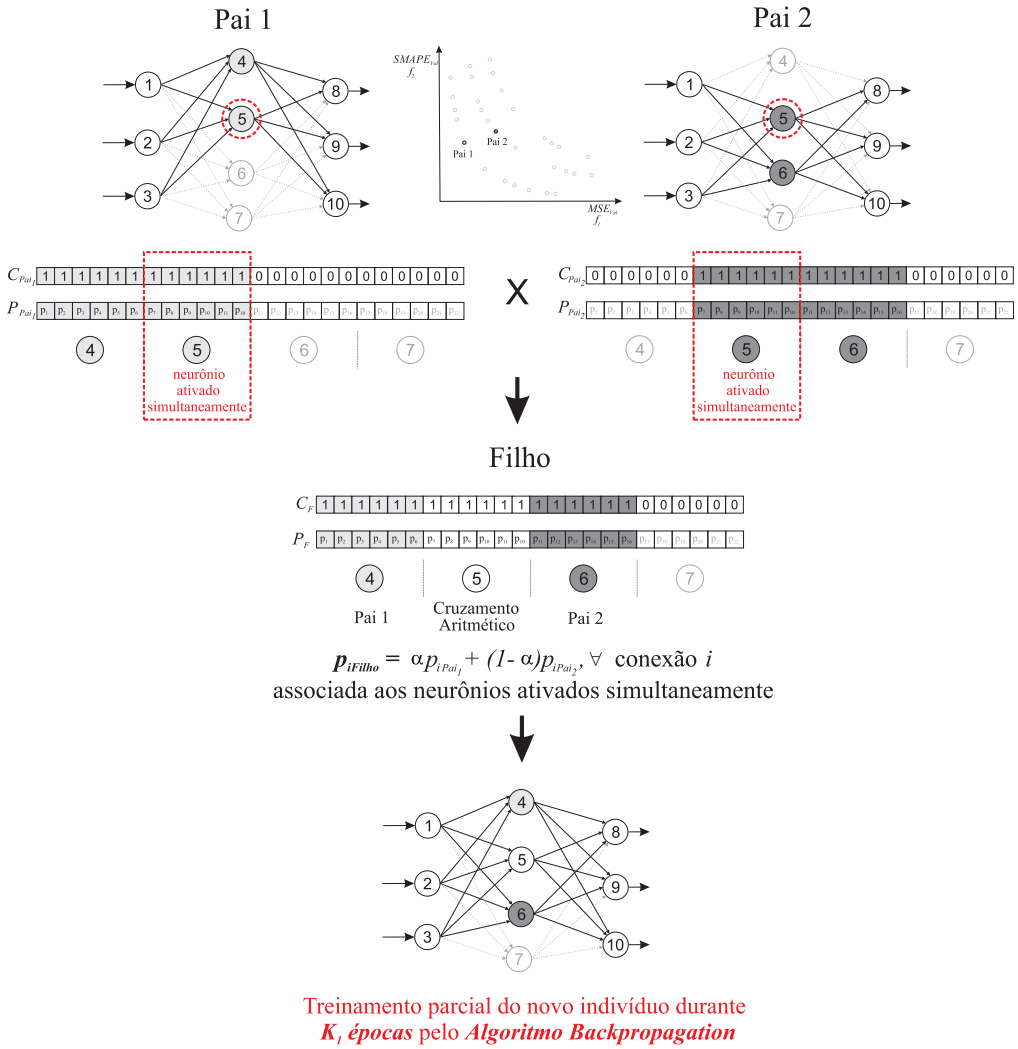


Figura 3.7: Ilustração do operador de cruzamento usado no modelo NEW-GA.

Uma vez que o novo indivíduo filho é gerado, este é submetido a uma etapa de ajuste fino dos seus pesos sinápticos a partir do algoritmo local de treinamento backpropagation, o qual é executado durante um número k_1 pequeno de iterações antes de que o novo indivíduo seja mapeado sobre o espaço de objetivos (Seção 3.2.3).

3.2.4.3 Operador de Mutação

A decisão de aplicar o operador de mutação é realizada obedecendo uma *taxa de mutação adaptativa* que aumenta gradativamente com o passar das gerações, com o intuito de que no início do processo evolutivo seja pouca a alteração das partes cromossômicas, deixando o operador de busca local atuar durante esta etapa inicial do processo evolutivo, além de evitar que

as soluções caiam em regiões de mínimo local durante etapas avançadas. Uma das principais funções deste operador é também percorrer o espaço de possíveis arquiteturas nas redes a evoluir, uma vez que o operador atua sobre as soluções habilitando ou desabilitando os neurônios da camada oculta codificados nos cromossomos.

Foram considerados 3 tipos de mutação. Somente um deles é utilizado a cada ocorrência de mutação, com probabilidade uniforme (0,33 para cada um deles). A seguir, a descrição dos 3 tipos de mutação adotados:

Tipo 1: Habilitação de um neurônio oculto. Se existirem neurônios ocultos desabilitados (pelo menos 1) dentro da codificação do indivíduo, escolhe-se de forma aleatória um destes e trocam-se seus correspondentes bits no vetor de conectividade, de modo que o neurônio faça parte da camada oculta na rede. O valor dos pesos associados ao neurônio habilitado são inicializados da mesma forma que os pesos nos indivíduos da população inicial (Seção 3.2.2).

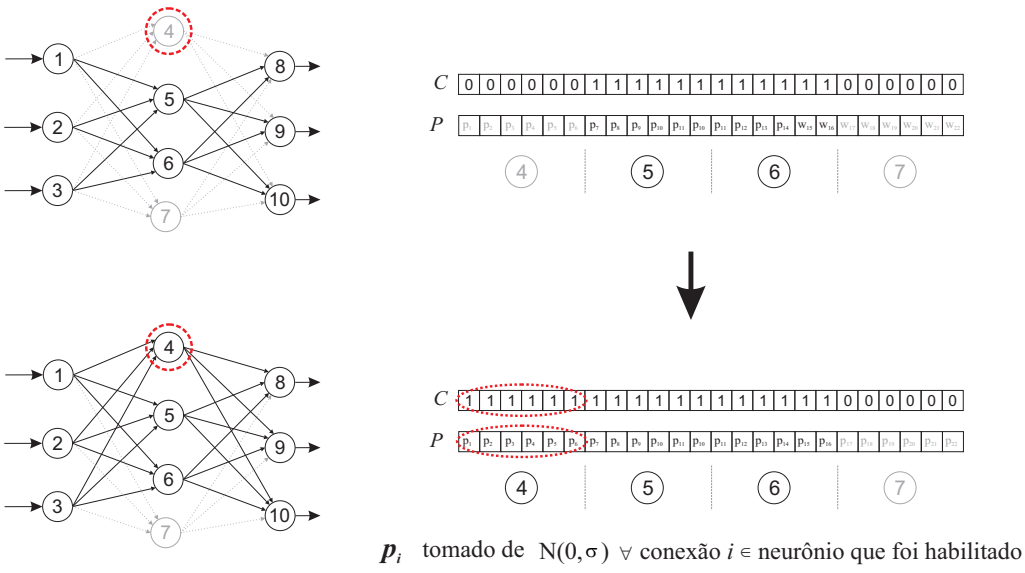


Figura 3.8: Ilustração do primeiro operador de mutação usado no modelo NEW-GA.

Tipo 2: Desabilitação de um neurônio oculto. Se existirem pelo menos 2 neurônios habilitados dentro da codificação do indivíduo, escolhe-se de forma aleatória um destes e trocam-se seus correspondentes bits no vetor de conectividade, de modo que o neurônio não faça mais parte da camada oculta na rede. Os valor dos pesos associados ao neurônio desabilitado não sofrem nenhuma mudança.

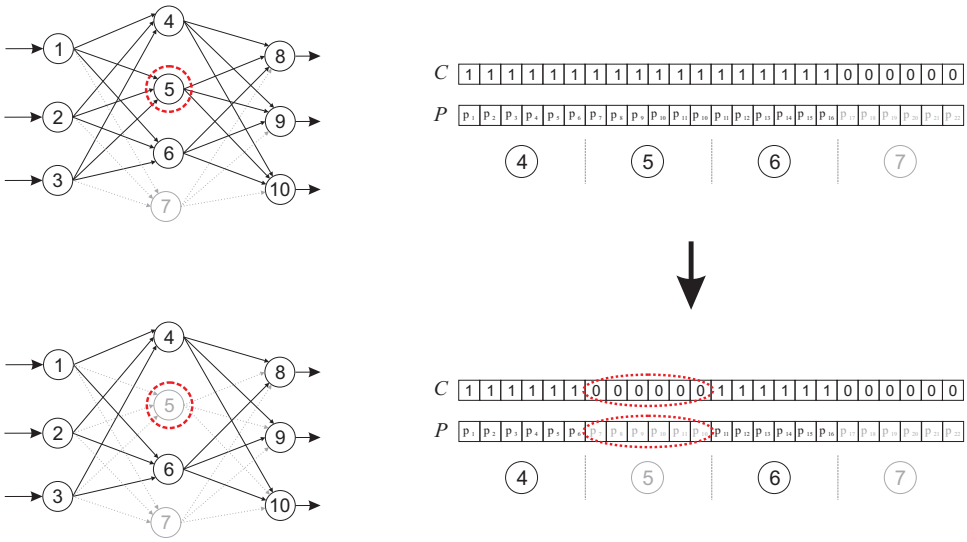


Figura 3.9: Ilustração do segundo operador de mutação usado no modelo NEW-GA.

Tipo 3: Perturbação Gaussiana. Para cada conexão i associada a cada neurônio habilitado dentro da codificação do indivíduo, extrai-se um número aleatório β a partir de uma distribuição normal $\mathcal{N}(0, \sigma_P)$ (onde σ_P é dado por (3-14)), e o valor do novo peso p_i correspondente à conexão i de cada neurônio habilitado é dado por:

$$p_i' = p_i + \beta \tag{3-18}$$

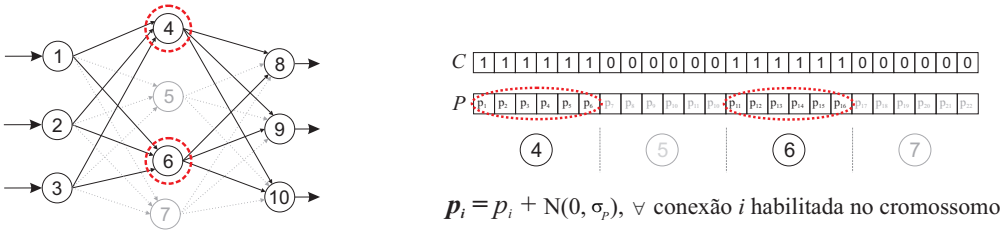


Figura 3.10: Ilustração do terceiro operador de mutação usado no modelo NEW-GA.

Cabe ressaltar que, da mesma forma que no operador de cruzamento, cada indivíduo após o processo de mutação é submetido também a uma etapa

de ajuste dos seus pesos sinápticos a partir do algoritmo local de treinamento backpropagation.

3.2.4.4 Operador de Busca Local

Sabe-se que técnicas de otimização global são relativamente ineficientes para o ajuste fino em buscas locais. Dessa forma, é importante melhorar o desempenho de generalização das redes treinando as topologias com um método de busca local. Na abordagem proposta, a cada certo número de gerações, uma parcela da população (quarta parte, escolhida a partir do operador de seleção proposto na Subseção 3.2.4.1) é submetida a um processo de ajuste dos seus vetores de pesos a partir do algoritmo local de treinamento *backpropagation*, o qual é executado durante um considerável número $k_2 \gg k_1$ de iterações, utilizando os padrões do conjunto de treinamento (equação (3-5)).

3.2.5 Critérios de Parada do Algoritmo

A execução do algoritmo é finalizada se: (1) o indicador de convergência adotado for satisfeito; ou (2) a quantidade máxima de gerações for atingida. Para a implementação do indicador de convergência é importante definir uma das métricas de desempenho mais citadas na avaliação de algoritmos multiobjetivos, sendo esta o indicador de *hipervolume* [71] a qual é apresentada a seguir.

3.2.5.1 Hiper-volume

O hiper-volume é uma métrica qualitativa de convergência e diversidade (as duas principais metas da otimização multiobjetivo) das soluções não dominadas, fornecidas pelo algoritmo a cada geração. Este indicador mede o volume coberto pelas soluções componentes da fronteira não dominada (Q), encontradas pelo algoritmo no espaço definido pelas funções objetivo, sendo aplicável apenas quando todos os objetivos devem ser minimizados. Matematicamente, para cada solução $i \in Q$, é construído um hipercubo v_i (com dimensões igual ao número de objetivos do problema sob análise) com base em um ponto de referência W e usando a solução i como diagonal oposta. Uma maneira fácil de determinar o ponto de referência é construir um vetor com os piores valores das funções objetivo avaliadas no início do processo evolutivo. Posteriormente, é encontrada a união de todos os hipercubos encontrados, sendo esta união o resultado da métrica de hiper-volume. Um alto valor de hiper-volume indica que houve um elevado espalhamento entre as soluções extremas da fronteira não dominada e indica, também, que houve uma maior

convergência, pois a convergência aumenta o volume em relação ao ponto de referência.

A Figura 3.11 ilustra o hiper-volume coberto pelas soluções não dominadas durante diferentes etapas do processo de otimização proposto.

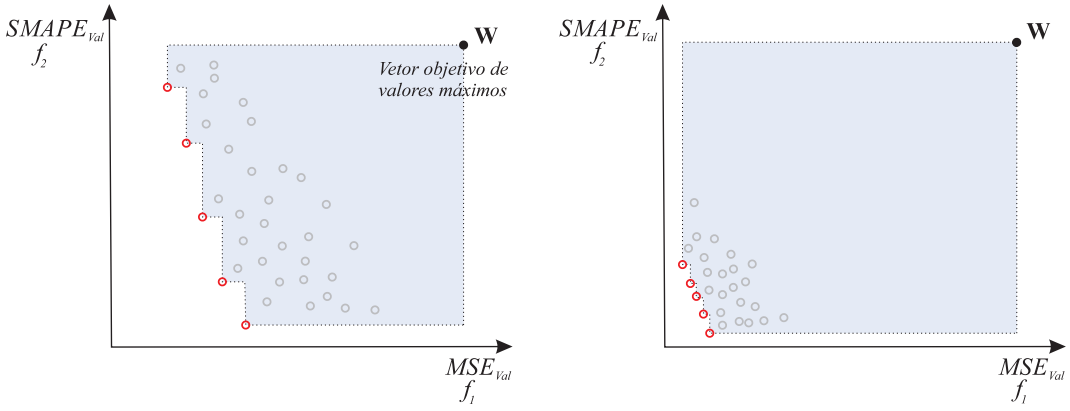


Figura 3.11: Hipervolume coberto pelas soluções não dominadas em diferentes etapas do processo evolutivo.

O hiper-volume é representado a partir de (3-19) e quanto maior o seu valor, melhor será a fronteira de soluções não dominadas encontrada.

$$HV = volume\left(\bigcup_{i=0}^{|Q|} v_i\right) \tag{3-19}$$

3.2.5.2 Indicador de Convergência pelo Hiper-volume

Uma vez exposta a métrica de desempenho do hiper-volume, procede-se a apresentar o indicador de convergência utilizado como parte de um dos critérios de parada do algoritmo proposto. Os seguintes passos explicam de forma geral o procedimento adotado para avaliar este indicador a cada iteração do algoritmo:

1. Estima-se o maior hiper-volume obtido até a iteração atual ***it:***

$$HV_{Opt}(it) = \max_{it' \leq it} HV(it') \tag{3-20}$$

2. Estima-se o ganho do hiper-volume na iteração ***it***, definido como o aumento do mesmo em relação ao maior hiper-volume obtido até a atual ***it*** (em porcentagem):

$$G_{HV}(it) = 100 * \left(\frac{HV(it)}{HV_{Opt}(it)} - 1\right) \tag{3-21}$$

3. Estabelece-se um limiar ε, de modo que quando o ganho do hiper-volume for menor do que ε durante um número determinado τ de iterações, o processo evolutivo é interrompido:

$$G_{HV}(it) < \varepsilon \tag{3-22}$$

3.2.6 Seleção da Solução Final

Uma vez satisfeito um dos critérios de parada descritos anteriormente, não existirá somente uma solução para o problema, mas sim um conjunto de soluções ótimas, denominado *conjunto Pareto ótimo* (que quando mapeado no espaço de objetivos denomina-se *Fronteira de Pareto Ótima*). Isto se deve às caraterísticas próprias dos algoritmos de otimização multiobjetivo abordadas na Seção 2.5. Neste sentido, torna-se necessário um procedimento que permita selecionar uma única solução a partir do conjunto Pareto ótimo fornecido pelo algoritmo.

Define-se como *vetor objetivo ideal* ($\bar{z}$) aquele vetor de funções objetivo contendo valores de mínimo global (assuma-se conhecidos), para todas e cada uma das funções objetivo a minimizar durante o processo de otimização. Por ser o *vetor objetivo ideal* inalcançável, o valor ótimo ou a melhor solução compromisso vem dada pela solução eficiente mais próxima do vetor objetivo ideal. Esta regra de comportamento é normalmente denominada de *axioma de Zeleny* [72]. De acordo com este postulado, dado um conjunto de soluções, a solução eleita (ótima) será aquela que estiver mais próxima do vetor objetivo ideal.

É assim como a *Distância Euclideana Normalizada* (d_s) entre cada solução localizada na Fronteira Pareto Ótima e o vetor objetivo ideal é calculada, permitindo a identificação da solução mais próxima daquele vetor:

$$d_s = \sqrt{\sum_{i=1}^{N} w_i \left(\frac{f_i(s) - \bar{z}_i}{f_i^{max} - f_i^{min}} \right)^2} \tag{3-23}$$

Na equação anterior, f_i^{max} e f_i^{min} correspondem ao valor máximo e mínimo do vetor das N funções objetivo para o critério i respectivamente, $\bar{z}_i$ e w_i são os $i-$ésimo componentes do vetor objetivo ideal e do vetor de pesos respectivamente. A equação (3-23) leva em conta a importância relativa de cada função objetivo a partir do vetor de pesos w. Deste modo, a solução com menor distância do ponto ideal será preferivelmente selecionada.

No caso específico dos modelos NEW-GA, associa-se para cada função objetivo um mesmo valor de peso (0,5), assumindo que os dois objetivos são igualmente importantes no processo de otimização da rede. O vetor objetivo

ideal, por sua parte, está localizado na origem do plano que define o espaço de objetivos. A Figura 3.12 ilustra a ideia.

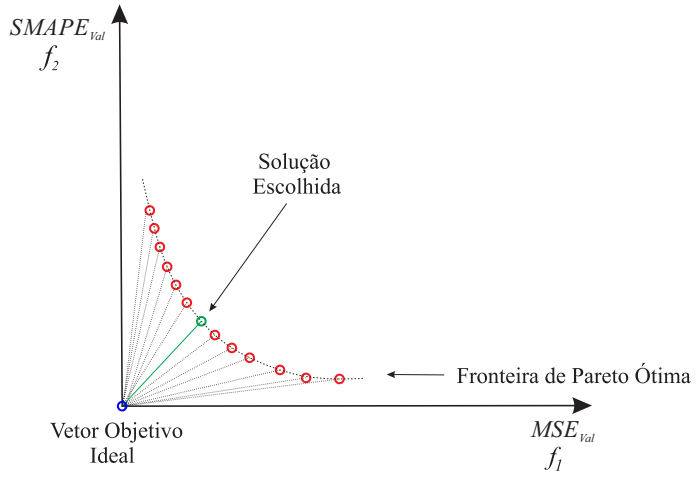


Figura 3.12: Ilustração do processo de seleção da solução final.

3.3 Resumo

Neste capítulo apresentou-se o NEW-GA, um modelo de treinamento híbrido para redes neurais do tipo MLP aplicável ao núcleo do sistema NEW. Na sua elaboração foram usados conceitos presentes na literatura de Algoritmos Genéticos Multiobjetivo, Redes Neurais, Operadores e Otimização. A união deles proporcionou diversos ganhos, tanto teóricos, quanto práticos. O próximo capítulo apresenta diferentes testes do modelo NEW-GA, efetuados a partir de experimentos com séries de vendas mensais de produtos importantes no mercado brasileiro de derivados, assim como também com séries benchmarks presentes na literatura. Os resultados são comparados com os obtidos pelo modelo NEW original e também com os modelos tradicionais de ponderação de previsores.

4
Estudo de Casos

Este capítulo apresenta os estudos de casos realizados para avaliar o desempenho do modelo descrito no capítulo anterior. O intuito principal reside em exibir a qualidade das soluções fornecidas pelo modelo, em termos de acurácia/eficácia, quando comparado com os modelos individuais de previsão (Seção 4.1), com os métodos tradicionais de ponderação (Seção 2.1) e, naturalmente, com respeito ao modelo NEW original. Dois estudos de caso foram selecionados:

- **Derivados do Petróleo:** São analisadas séries de vendas mensais de quatro produtos importantes no mercado brasileiro de derivados - óleo diesel, gasolina corrente, gás liquefeito de petróleo e querosene de aviação.
- **Competição NN3:** Realizada em 2007, como uma sessão especial no IJCNN[1]. O objetivo da competição foi efetuar uma avaliação empírica de diferentes modelos de previsão, com maior ênfase nos de Redes Neurais, de modo a verificar quais deles possuíam maior acurácia fora da amostra de treinamento (out-of-sample). Este trabalho utilizou a versão reduzida (11 séries), como estudo de caso. Enfatiza-se que, nos resultados do período da competição[2] e na literatura disponível após a competição, não foi observado modelo algum do tipo SNE.

Todas as rotinas executadas foram implementadas em MATLAB R2013b [73], executadas em um PC Windows 7 com processador Intel i7 de 3.6GHz e 16 GB de RAM. Também foram utilizadas as rotinas estatísticas do KEEL [74] e o pacote estatístico R na aplicação dos testes de hipótese apresentados na Seção 4.2. O pacote ForecastPro [75], por sua parte, foi utilizado para estimar os parâmetros dos modelos individuais de previsão apresentados na seção a seguir.

[1]http:\\www.ijcnn2007.org\competition.htm
[2]http:\\www.neural-forecasting-competition.com\NN3\results.htm

4.1 Previsores Disponíveis

Toda abordagem de combinação pressupõe a existência de previsores componentes. No modelo proposto, por ser uma extensão do modelo NEW, esses previsores são derivados das mesmas metodologias utilizadas no modelo original:

1. Holt-Winters multiplicativo (HW);
2. ARIMA Box & Jenkins (BJ).

Cumpre destacar que o processo de escolha, segundo [12], seguiu os critérios de busca por *complementaridade* (uma vez que estes previsores apresentam naturezas diferentes entre si), e capacidade de representar tendência e sazonalidade, caraterísticas recorrentes na maioria das séries testadas.

4.2 Metodologia de Avaliação dos Resultados

4.2.1 Métrica de Desempenho

A métrica de avaliação de desempenho utilizada no presente trabalho foi o *Symmetric Mean Absolute Percentage Error* (SMAPE). O SMAPE, medido em pontos percentuais (pp), é uma métrica amplamente utilizada em trabalhos relacionados à previsão de séries temporais, principalmente por seu emprego em competições para modelos de previsão [76,77], e por sua fácil interpretação, devido à adimensionalidade proporcionada pela razão de medidas. Os valores de erro são obtidos a partir da equação (4-1).

$$SMAPE(\%) = \frac{1}{H}\sum_{h=1}^{H}\frac{|y_{\tau+h} - \hat{y}_{\tau+h}|}{\left(\frac{|y_{\tau+h}|+|\hat{y}_{\tau+h}|}{2}\right)}100 \tag{4-1}$$

Onde, H é o horizonte máximo de previsão, $y_{\tau+h}$ é o valor real da série no conjunto de teste, e $\hat{y}_{\tau+h}$ é o valor previsto, obtido a partir de qualquer dos modelos individuais de previsão (HW ou BJ) ou pela combinação convexa destes a partir do vetor de ponderação gerado por:

1. Qualquer dos métodos tradicionais de ponderação (Seção 2.1);
2. O modelo NEW original ou;
3. O modelo NEW quando o seu núcleo é otimizado pelo algoritmo aqui proposto (NEW-GA).

4.2.2 Testes de Hipótese

De modo a garantir a validade das conclusões tomadas, testes de hipótese são aplicados ao modelo NEW-GA sobre o desempenho no conjunto dos dados de teste, assumindo sempre um nível de significância (α) de 5%. Um primeiro grupo de testes - **teste t, teste do Sinal** e **teste de Wilcoxon** [**78**–**81**] - verificam se a mediana[3] das diferenças de desempenho entre dois métodos é (estatisticamente) nula. Como será visto mais adiante, propõe-se para este primeiro grupo uma arquitetura de comparação onde as diferenças de desempenho são medidas para cada um dos horizontes de previsão considerados (de 1 até H passos a frente). A unidade de medida das diferenças é, logicamente, a mesma dos desempenhos sendo comparados (SMAPE). Um segundo grupo de testes - **teste de Friedman (Iman e Davenport)** e **teste (Critério) de Holm** [82] - é aplicado, levando diretamente em conta a métrica de desempenho associada a cada um dos horizontes de previsão estimados para cada modelo.

- ***teste t***: ***'Ho'*** a ***Média*** é zero. Teste paramétrico no qual assume-se que a distribuição das diferenças de desempenho é normal; isto nem sempre pode ser assumido, principalmente se o tamanho da amostra é reduzido (<30). A validade da premissa de distribuição normal pode ser checada pela análise de gráficos Q-Q [39] ou por testes específicos de normalidade, por exemplo, o Jarque-Bera [83].
- ***teste do Sinal e teste de Wilcoxon***: ***'Ho'*** a ***Mediana*** é zero. Testes não paramétricos os quais dispensam a hipótese de normalidade entre as diferenças de desempenhos. A principal diferença entre eles é que o teste do Sinal não leva em conta a magnitude da diferença de desempenhos enquanto que o teste do Wilcoxon sim, tornando este último mais acertado no caso onde a magnitude da diferença de desempenhos é plausível.
- ***teste de Friedman***: É uma generalização do teste do Sinal. Este se propõe a avaliar, dada uma medida de desempenho, qual o modelo, ou modelos, obtiveram o posto significativamente diferente dos demais métodos em comparação. Portanto, ao comparar 3 ou mais modelos de forma simultânea, o teste de Friedman examina qual obteve o menor SMAPE em cada um dos horizontes de previsão.

[3] A **mediana** é considerada uma medida mais robusta (resistente a *outliers*) do que a **média**.

– ***teste de Holm***: De fato não é necessariamente um teste, mais sim um critério de correção no nível de significância α estabelecido previamente pelo usuário. Mas por conveniência, será usado este rótulo.

4.3 Parâmetros Iniciais do Algoritmo

A Tabela 4.1 apresenta a configuração inicial dos principais parâmetros no modelo NEW-GA, utilizados durante a otimização da rede MLP para cada uma das séries avaliadas.

Parâmetro	Valor
Tamanho da população	60
Número máximo de Gerações	1000
Número máximo de neurônios ocultos	20
Épocas de treinamento parcial pelo algoritmo BP (k_1)	10
Épocas de treinamento durante a execução do operador de busca local (k_2)	100
Taxa de seleção do operador de busca local	25%
Frequência de execução do operador de busca local (gerações)	20
Limiar do ganho mínimo de hiper-volume (ε)	0.5%
Número de gerações de parada antecipada (τ)	50
Taxa de seleção do operador de cruzamento	Equação (4-2)
Taxa de seleção do operador de mutação	Equação (4-3)

Tabela 4.1: Principais configurações do modelo NEW-GA.

As taxas de cruzamento e mutação, adaptadas de Barbosa et al. [84], são definidas pelas equações (4-2) e (4-3) respectivamente, onde ng é o número total de gerações e t é a geração corrente.

$$T_{cru} = \frac{0,8}{1 + \exp^{\left(-15\left(\frac{it}{ng} - 0,3\right)\right)}} + 0,1 \tag{4-2}$$

$$T_{mut} = \frac{0,8}{1 + \exp^{\left(-8\left(\frac{it}{ng} - 0,5\right)\right)}} + 0,1 \tag{4-3}$$

O comportamento da taxa de cruzamento é mostrado na Figura 4.1. Com o passar das gerações a taxa de cruzamento aumenta, de forma que os novos indivíduos contenham as partes boas dos indivíduos antigos.

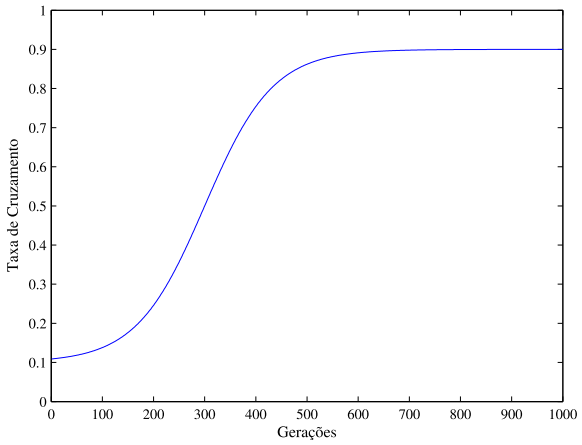


Figura 4.1: Taxa de cruzamento dinâmica utilizada no modelo NEW-GA.

Por sua parte, o comportamento da taxa de mutação, mostrado na Figura 4.2, evita que o algoritmo caia em regiões de mínimo local durante etapas avançadas do processo evolutivo.

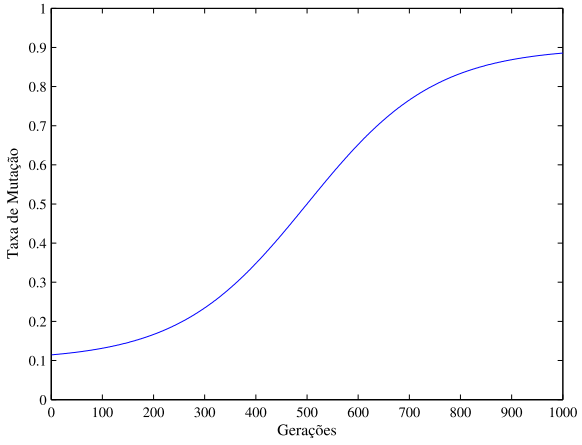


Figura 4.2: Taxa de mutação dinâmica utilizada no modelo NEW-GA.

É importante mencionar que durante a construção dos conjuntos de dados (treinamento, validação e teste), que serão utilizados pelo algoritmo durante o processo de otimização das redes, os hiper-parâmetros do sistema - janela de tempo utilizada na estimação dos vetores de pesos e tipo de previsores (limiares ou não limiares) - são configurados da mesma forma que no modelo NEW original. Isto deve-se ao objetivo de levar a cabo uma análise comparativa não enviesada do modelo proposto com respeito do modelo NEW original, levando em conta apenas a influência na configuração automática dos parâmetros no núcleo do sistema.

4.4 CASO 1: Séries Derivados do Petróleo

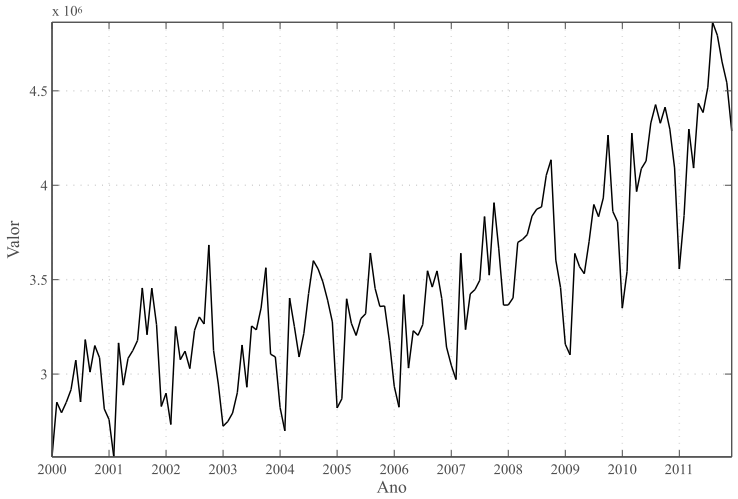
Para as empresas integradas de Petróleo & Gás, o uso de técnicas de séries temporais é útil nas atividades de planejamento relacionadas ao marketing e comercialização de derivados. Não obstante, pode-se encontrar aplicações destas técnicas em outras áreas: financeira, materiais, Gás & Energia e Exploração & Produção.

Com o objetivo de estudar a aplicação das combinações de previsores ao mercado brasileiro de derivados do petróleo, o presente trabalho utiliza dados reais publicados pela Agência Nacional do Petróleo, Gás Natural e Biocombustíveis (ANP), relativos às vendas mensais (em MMm^3) de **óleo diesel** (DIESEL), **gasolina convencional** (GASOLINA), **gás liquefeito de petróleo** (GLP) e **querosene de aviação** (QAV), no período compreendido desde Janeiro de 2000 até Dezembro de 2011 [85]. Na Figura 4.3 exibem-se estas séries.

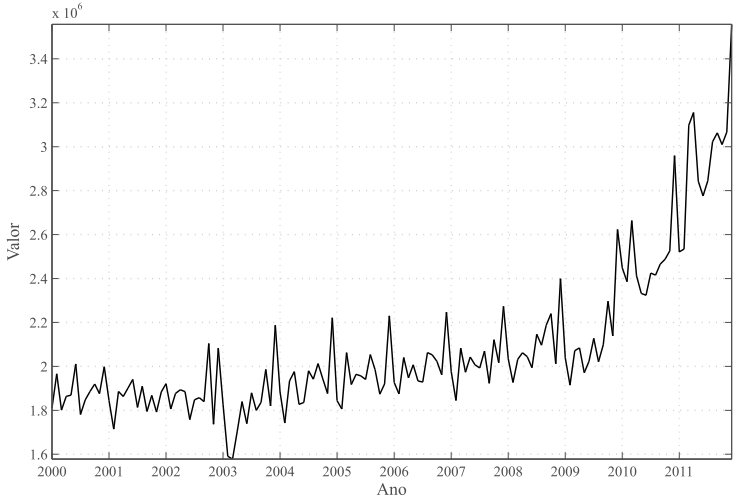
Numa primeira etapa, como parte da sequência de experimentos abordada no trabalho NEW original [12], as metodologias de previsão citadas na Seção 4.1 são aplicadas às séries disponíveis (DIESEL, GASOLINA, QAV e GLP), separando-se sempre as últimas 12 observações para o conjunto de teste (série de teste). Com os modelos ajustados são geradas previsões até 12 passos a frente, as quais são comparadas com as observações da série teste a partir da métrica de desempenho em (4-1).

Posteriormente, sobre os vetores de previsão gerados para cada observação da série de teste a partir das metodologias individuais de previsão HW e BJ - podendo ser substituídas pelos correspondentes previsores limiares (Seção 2.2) - aplicam-se os diferentes métodos tradicionais de ponderação citados na Seção 2.1, sendo selecionado o método que apresentar menor erro quando comparadas as previsões combinadas resultantes com respeito às observações da série de teste segundo a métrica de desempenho em (4-1).

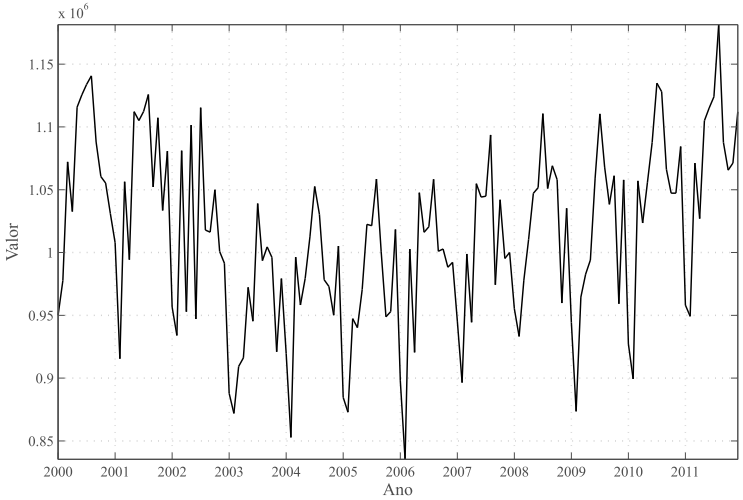
Finalmente, sobre os vetores de previsão (associados a cada observação da série de teste), aplicam-se também os vetores de ponderação estimados tanto pelo modelo NEW original quanto pelo modelo NEW otimizado (NEW-GA), de modo que os seus desempenhos são avaliados a partir da comparação entre os respetivos vetores de previsão combinada resultantes e as observações reais da série de teste, utilizando a métrica de desempenho em (4-1).



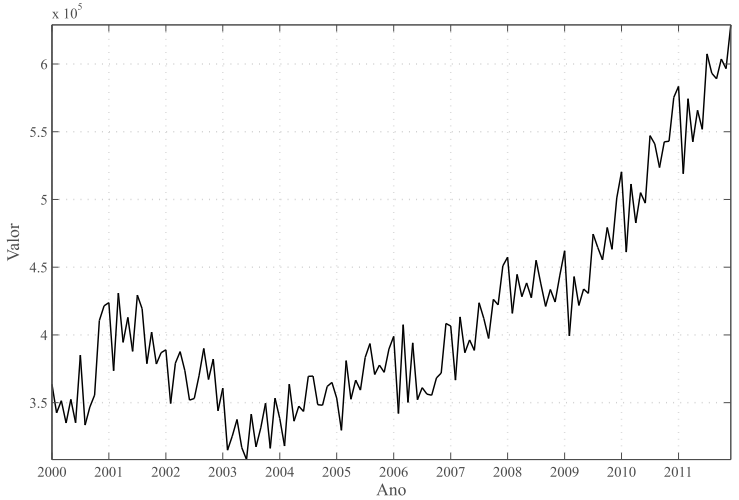
4.3(a): Série DIESEL



4.3(b): Série GASOLINA



4.3(c): Série GLP



4.3(d): Série QAV

Figura 4.3: Séries de vendas mensais de produtos derivados do petróleo no Brasil (Jan/2000 a Dez/2011).

A Tabela 4.2 exibe os desempenhos **totais** do modelo NEW-GA, obtidos fora da amostra (12 meses), para as diferentes séries de derivados do petróleo. Exibe-se também, sobre a mesma tabela, tanto os desempenhos totais dos previsores individuais, quanto os desempenhos totais do melhor modelo tradicional e do modelo NEW original, sendo que as últimas linhas apresentam as estatísticas (média e desvio padrão) da métrica de desempenho SMAPE para os diferentes modelos.

Série	HW	BJ	TRAD	NEW	NEW-GA
DIESEL	3,02	2,82	3,29	2,73	**2,32**
GASOLINA	12,03	10,46	7,82	6,74	**6,17**
GLP	1,66	1,97	1,55	1,64	**1,43**
QAV	3,57	2,25	3,04	1,40	**1,02**
Média	5,07	4,37	3,93	3,13	**2,73**
Desvio Padrão	4,71	4,07	2,71	2,48	**2,35**

Tabela 4.2: Desempenhos totais para as séries de derivados do petróleo. Valores em negrito indicam os melhores resultados obtidos em termos de desempenho.

De forma similar, a Figura 4.4 apresenta o diagrama de dispersão (*boxplot*) correspondente aos desempenhos totais registrados na Tabela 4.2. Neste gráfico é possível observar que os erros obtidos a partir do modelo NEW-GA apresentam baixa variabilidade, além de ter o valor médio mais baixo de todos os modelos.

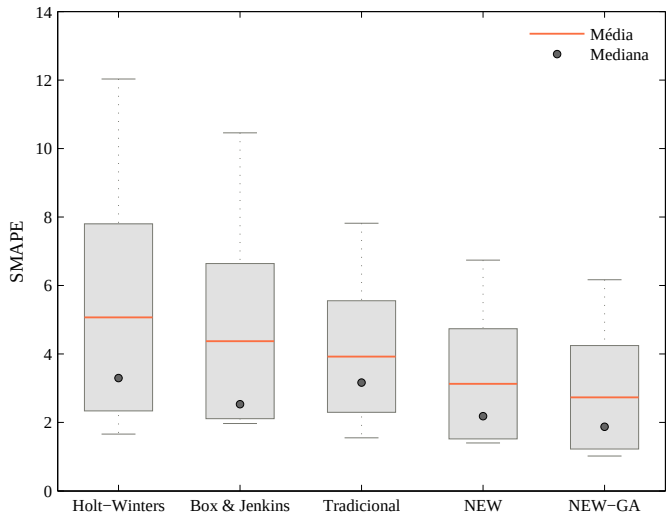


Figura 4.4: SMAPEs fora da amostra para as séries de derivados do petróleo: *boxplots*.

A Tabela 4.3 apresenta a configuração do modelo NEW-GA obtido para cada série avaliada. É importante observar que os hiper–parâmetros utilizados para a construção dos conjuntos de treinamento, validação e teste são configurados levando em conta os resultados reportados pelo modelo NEW original [12], uma vez que a finalidade do presente trabalho, como já foi mencionado, é avaliar a nova abordagem para a configuração automática dos parâmetros no núcleo do modelo NEW. A Tabela 4.3 apresenta também, para cada série avaliada, os tempos de execução requeridos pela abordagem de treinamento proposta.

Série	Hiper–parâmetros	Arquitetura RNA	Tempo de Execução
Diesel	HW+/HW-/BJ+/BJ- Janela expansiva	5:19:4	1.44 h
Gasolina	HW+/HW-/BJ+/BJ- Janela expansiva	5:20:4	1.51 h
GLP	HW+/HW-/BJ+/BJ- Janela expansiva	5:15:4	1.41 h
QAV	HW/BJ Janela expansiva	3:20:2	2.01 h

Tabela 4.3: Modelos NEW-GA obtidos: A coluna hiper–parâmetros indica os previsores individuais e a janela de tempo utilizada para a estimação dos conjuntos de treinamento, validação e teste. A coluna da arquitetura RNA indica a estrutura final da rede MLP, onde o número de neurônios oculto é determinado pelo algoritmo de treinamento híbrido proposto. A última coluna indica o tempo de execução utilizado pelo modelo NEW-GA durante a modelagem da rede para cada série avaliada. Todos os experimentos foram executados em um PC Windows 7 com processador Intel i7 de 3.6 GHz.

A Tabela 4.4 exibe a evolução do SMAPE médio acumulativo, - tomado período a período ao longo do horizonte de previsão h - considerando os desempenhos tanto dos métodos individuais de previsão quanto dos modelos de combinação. Cada valor do SMAPE na Tabela 4.4 é uma medida de erro acumulada, de modo que os valores na última linha equivalem exatamente aos desempenhos totais exibidos na Tabela 4.2.

h	HW	BJ	TRAD	NEW	NEW-GA
1	2,48	3,74	2,75	3,80	4,21
2	2,74	2,57	2,75	2,96	3,32
3	3,64	3,16	3,84	3,45	3,31
4	4,28	3,83	4,03	4,01	3,61
5	4,42	3,76	4,16	3,63	3,20
6	4,36	3,55	4,03	3,33	2,82
7	4,34	3,51	3,88	3,18	2,57
8	4,66	3,91	4,08	3,21	2,67
9	4,88	4,14	4,11	3,12	2,69
10	4,84	4,10	4,01	3,10	2,66
11	4,97	4,24	3,95	3,04	2,67
12	**5,07**	**4,37**	**3,93**	**3,13**	**2,73**

Tabela 4.4: Evolução do SMAPE médio acumulativo ao longo do horizonte de previsão para as séries de derivados do petróleo.

Na Figura 4.5, gerada a partir dos dados registrados na Tabela 4.4, é interessante observar uma caraterística empírica das metodologias de combinação: *o erro médio pode decair à medida que o horizonte de previsão aumenta, em oposição ao comportamento natural dos previsores individuais* [6].

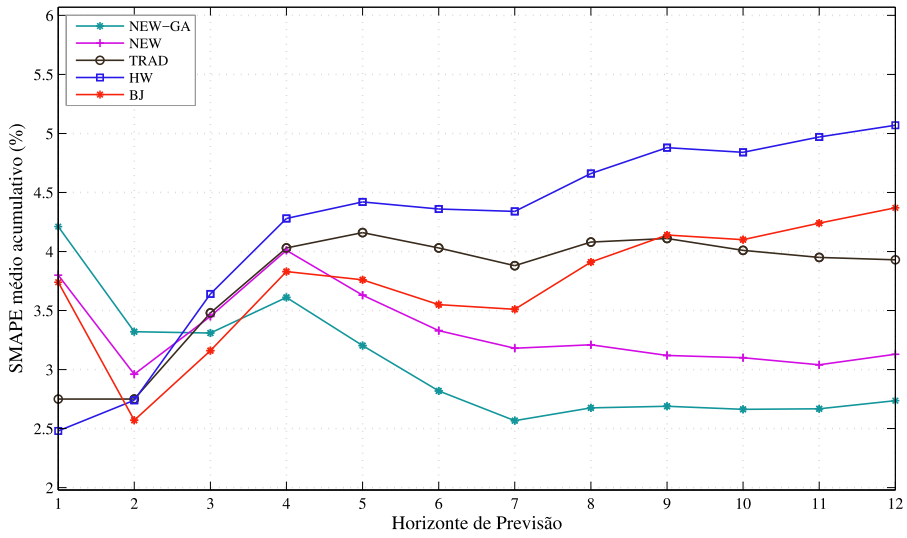


Figura 4.5: Evolução do SMAPE médio acumulativo ao longo do horizonte de previsão para as séries de derivados do petróleo.

Observa-se também, na Figura 4.5, que os modelos de combinação (Tradicional, NEW e NEW-GA) conseguem vencer os modelos individuais de

previsão, isto quando observado o desempenho total obtido por cada modelo no horizonte máximo de previsão.

Na Tabela 4.5 são apresentadas as diferenças de desempenho médio tomadas período a período ao longo do horizonte de previsão, na ordem "erro do modelo NEW-GA **menos** erro do modelo de referência". Quanto mais negativa é esta diferença, melhor o modelo NEW-GA proposto.

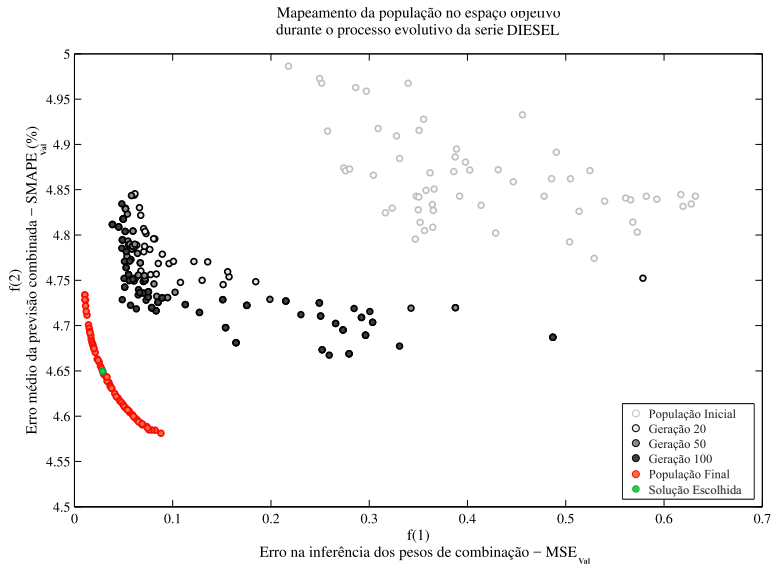
h	HW	BJ	TRAD	NEW
1	1,73	0,47	1,46	0,41
2	0,58	0,75	0,57	0,36
3	-0,33	0,15	-0,17	-0,14
4	-0,67	-0,22	-0,42	-0,40
5	-1,22	-0,56	-0,96	-0,43
6	-1,54	-0,73	-1,21	-0,51
7	-1,77	-0,94	-1,31	-0,61
8	-1,98	-1,23	-1,40	-0,53
9	-2,19	-1,45	-1,42	-0,43
10	-2,18	-1,44	-1,35	-0,44
11	-2,30	-1,57	-1,28	-0,37
12	-2,33	-1,63	-1,19	-0,39
Média	**-1,18**	**-0,70**	**-0,72**	**-0,29**
Mediana	**-1,66**	**-0,84**	**-1,20**	**-0,41**

Tabela 4.5: Diferenças de desempenho médio do modelo NEW-GA com respeito dos modelos de referência (individuais/combinação), ao longo do horizonte de previsão para as séries de derivados do petróleo.

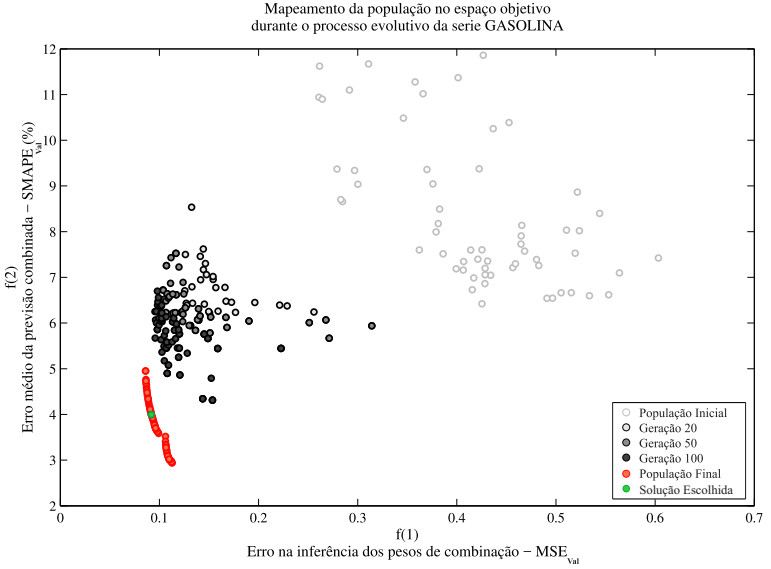
Antes de prosseguir com os testes de hipótese, propostos para avaliar a significância no desempenho do modelo NEW-GA, é interessante observar a Figura 4.6, a qual apresenta a capacidade do algoritmo para achar aquele conjunto de soluções compromisso (não dominadas entre sim) que minimizam simultaneamente as duas métricas de desempenho - avaliadas sobre o conjunto de validação - propostas na Seção 3.2.3.

Nas Figuras 4.7 a 4.10 exibe-se também, para cada série avaliada no primeiro estudo de casos, a evolução dos pesos de combinação ao longo do horizonte de previsão, considerando a solução escolhida (pelo critério de seleção abordado na Seção 3.2.6) a partir do conjunto de soluções ótimas fornecido pelo algoritmo uma vez satisfeito algum dos critérios de parada expostos na Seção 3.2.5. Em cada figura apresenta-se, para cada série, as previsões componentes e a previsão combinada que é gerada dinamicamente quando os vetores de previsão em cada ponto do horizonte são ponderados a partir

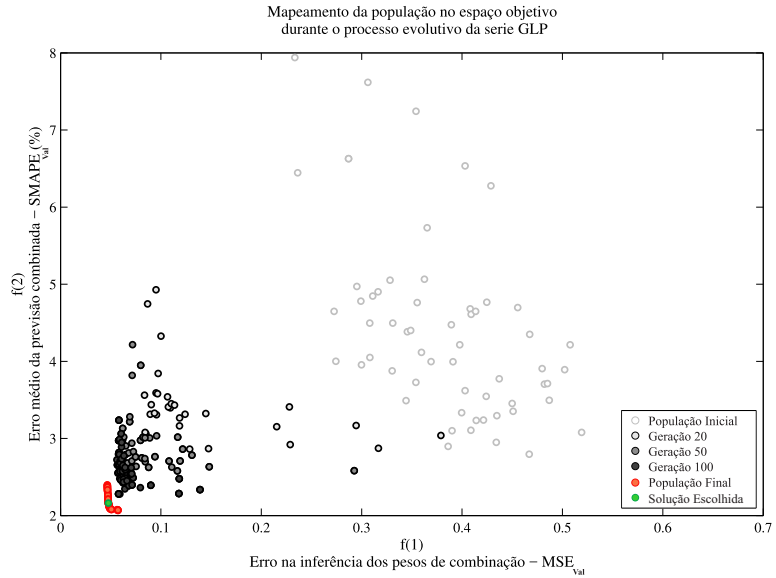
dos correspondentes vetores de pesos estimados naquele ponto. Igualmente, são exibidas as previsões dos modelos (individuais/combinação) avaliados, assim como a evolução dos seus respetivos SMAPEs ao longo do horizonte de previsão.



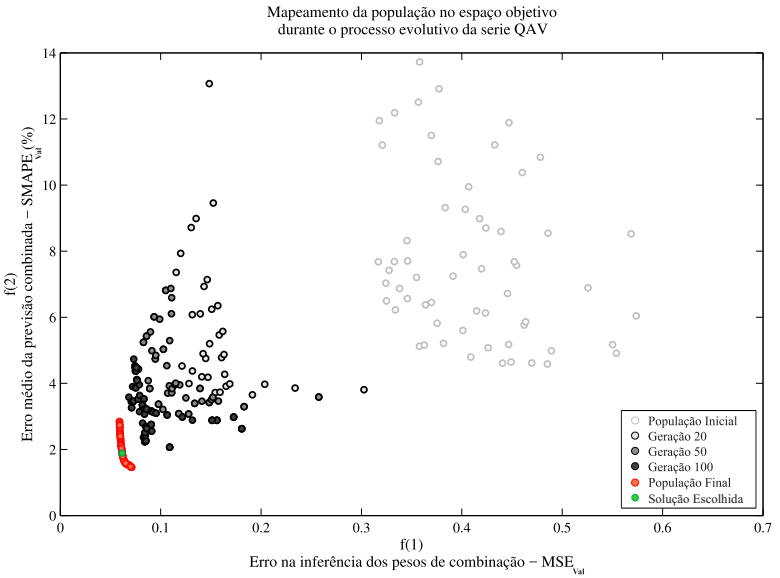
4.6(a): Progresso evolutivo na série DIESEL



4.6(b): Progresso evolutivo na série GASOLINA

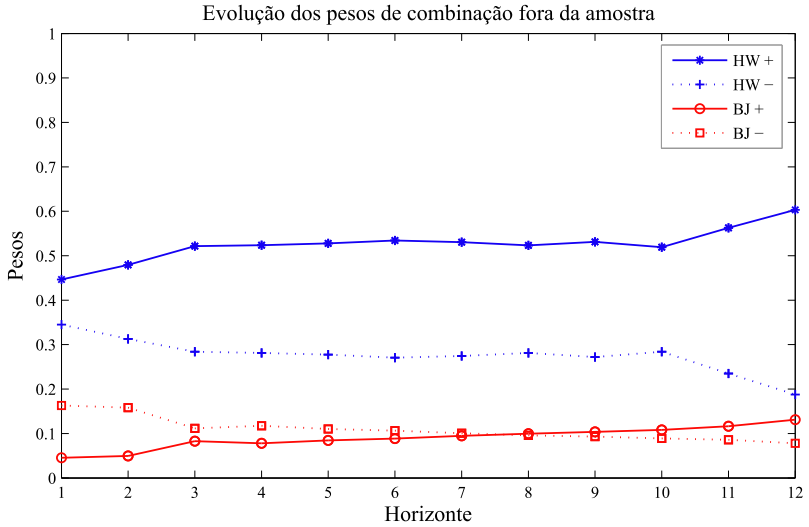


4.6(c): Progresso evolutivo na série GLP

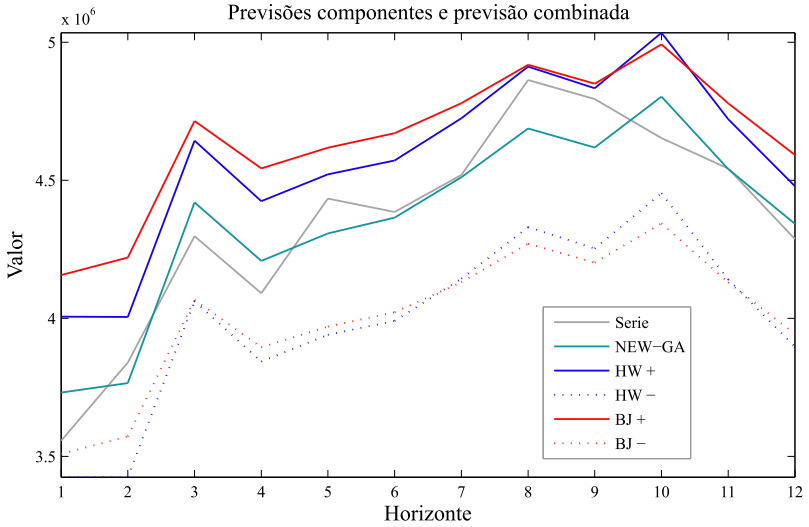


4.6(d): Progresso evolutivo na série QAV

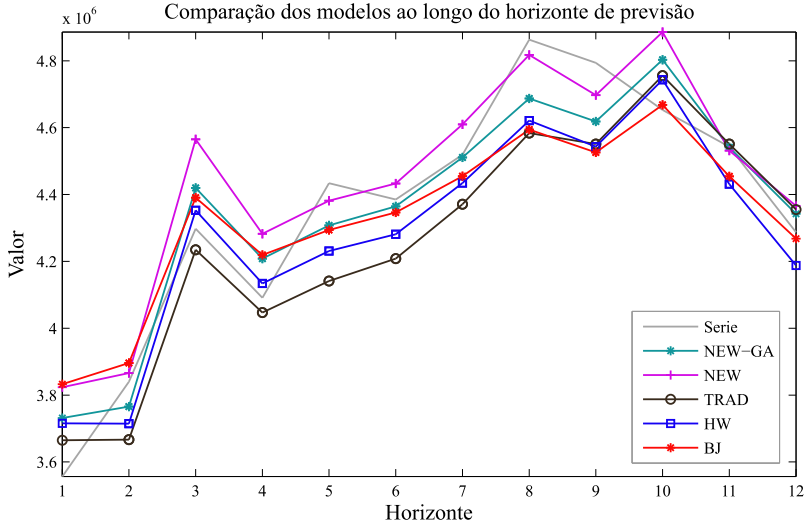
Figura 4.6: Mapeamento da população sobre o espaço de objetivos durante diferentes etapas do processo evolutivo.



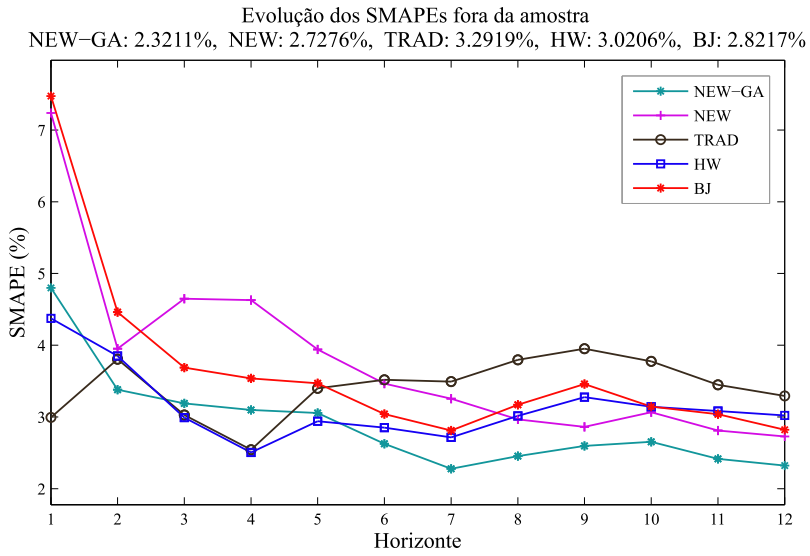
4.7(a): Evolução dos pesos de combinação



4.7(b): Previsões componentes e previsão combinada

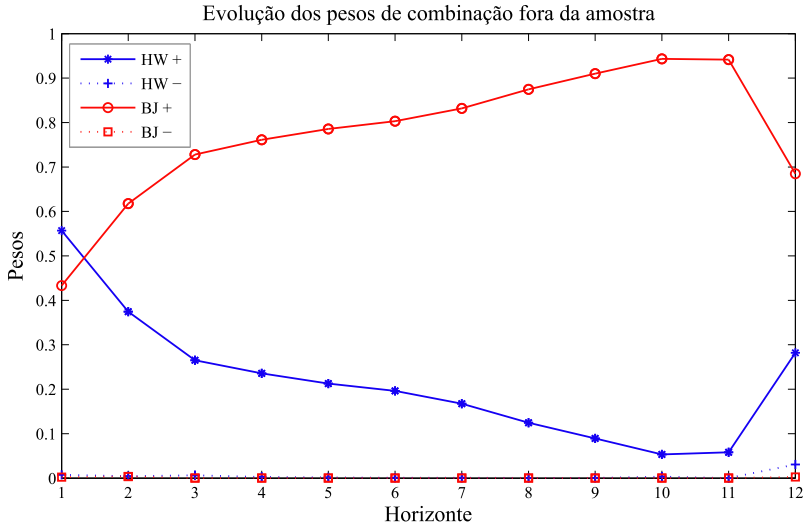


4.7(c): Previsões dos modelos de previsão avaliados

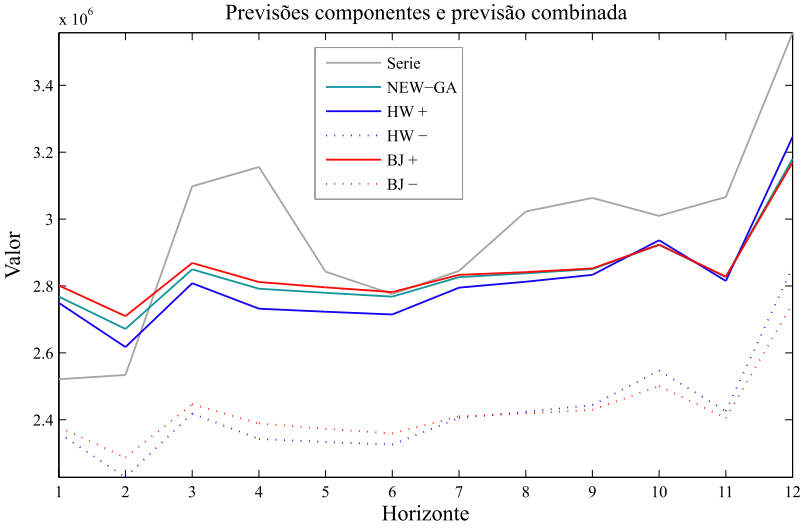


4.7(d): Evolução dos SMAPEs para cada modelo

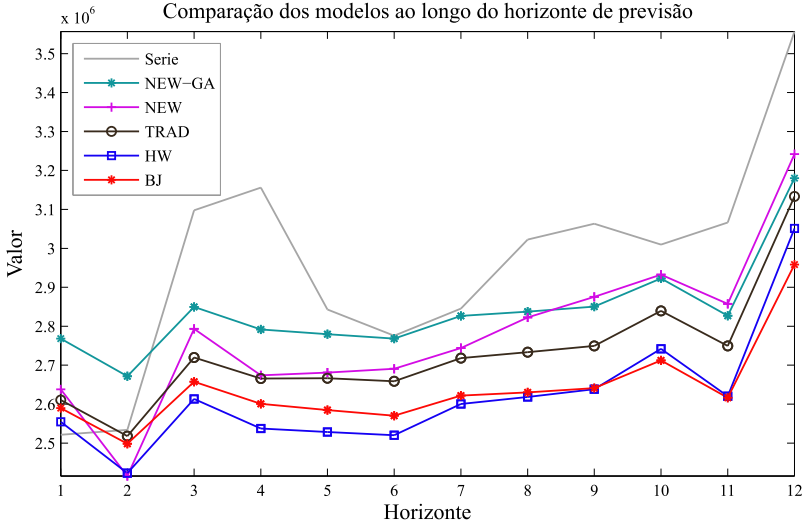
Figura 4.7: Resumo dos resultados obtidos para a série DIESEL.



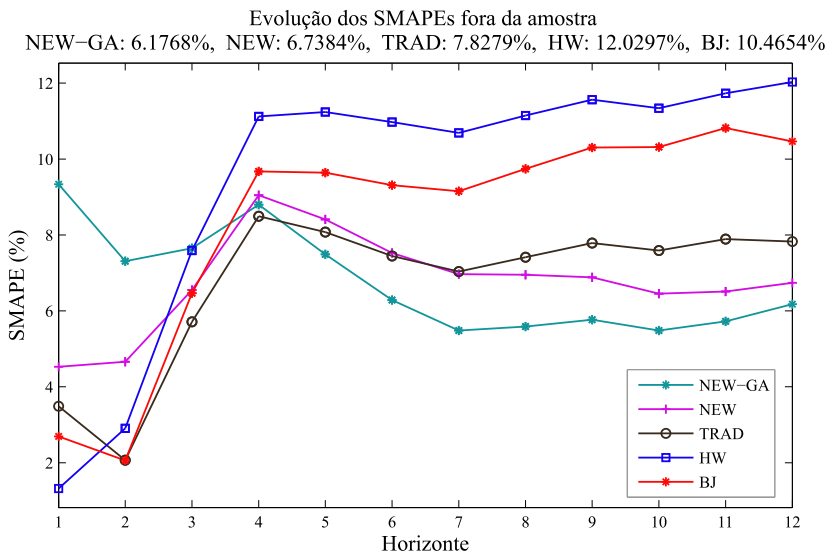
4.8(a): Evolução dos pesos de combinação



4.8(b): Previsões componentes e previsão combinada

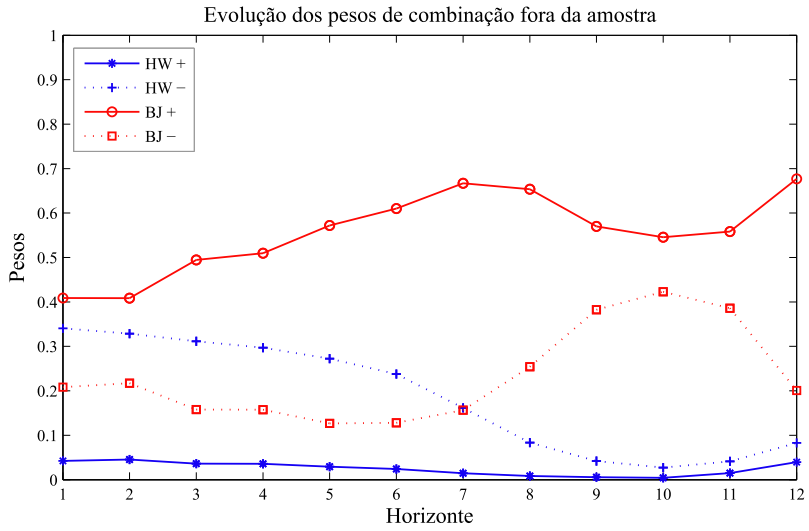


4.8(c): Previsões dos modelos de previsão avaliados

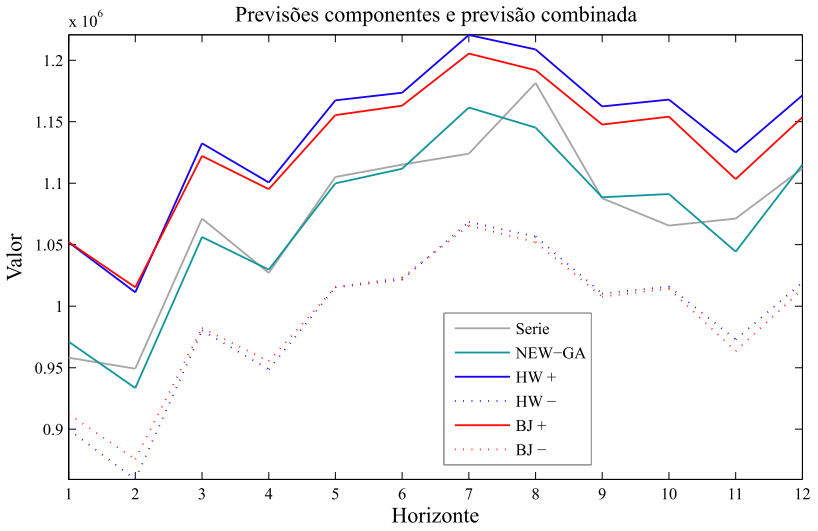


4.8(d): Evolução dos SMAPEs para cada modelo

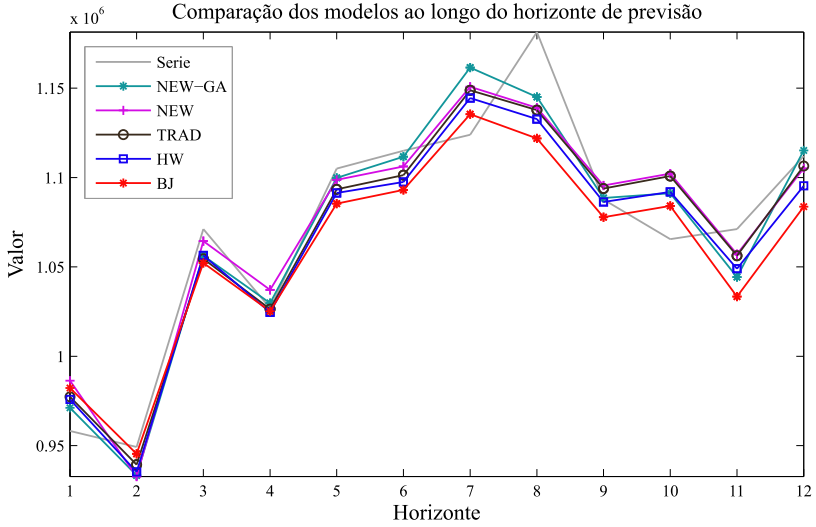
Figura 4.8: Resumo dos resultados obtidos para a série GASOLINA.



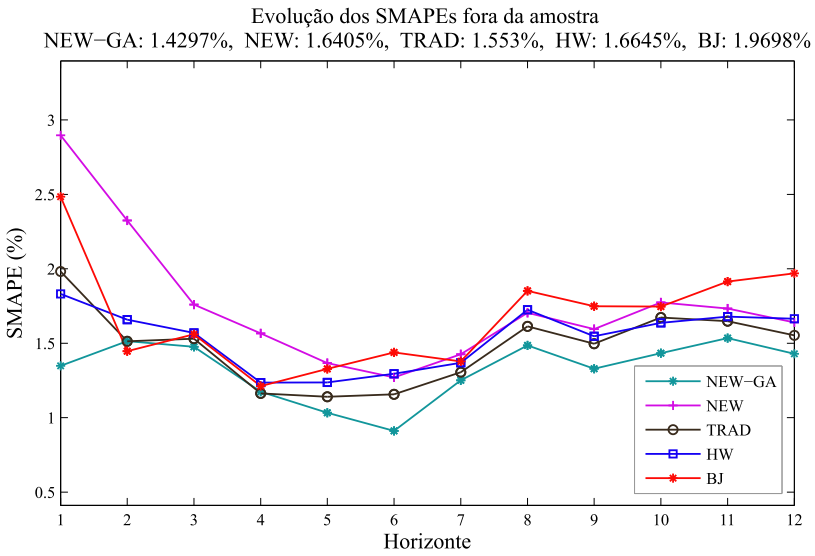
4.9(a): Evolução dos pesos de combinação



4.9(b): Previsões componentes e previsão combinada

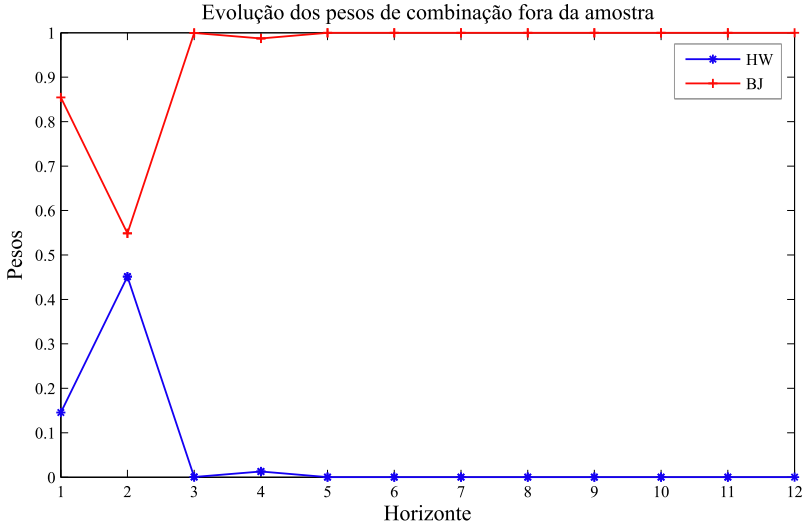


4.9(c): Previsões dos modelos de previsão avaliados

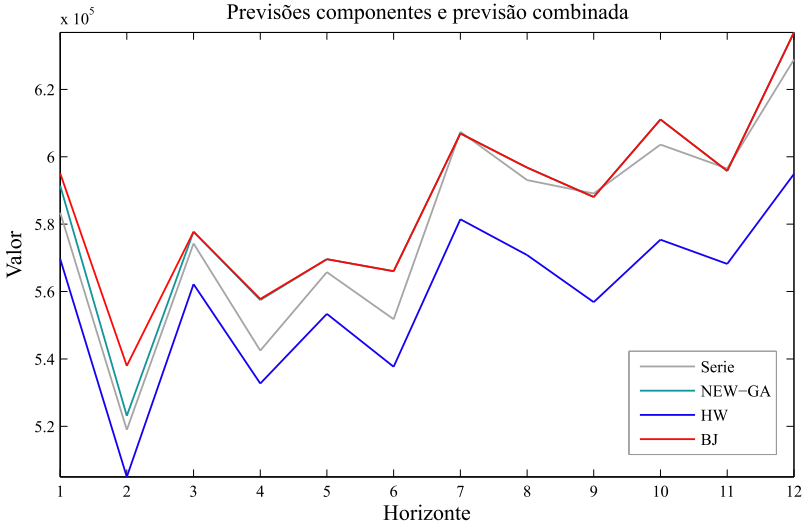


4.9(d): Evolução dos SMAPEs para cada modelo

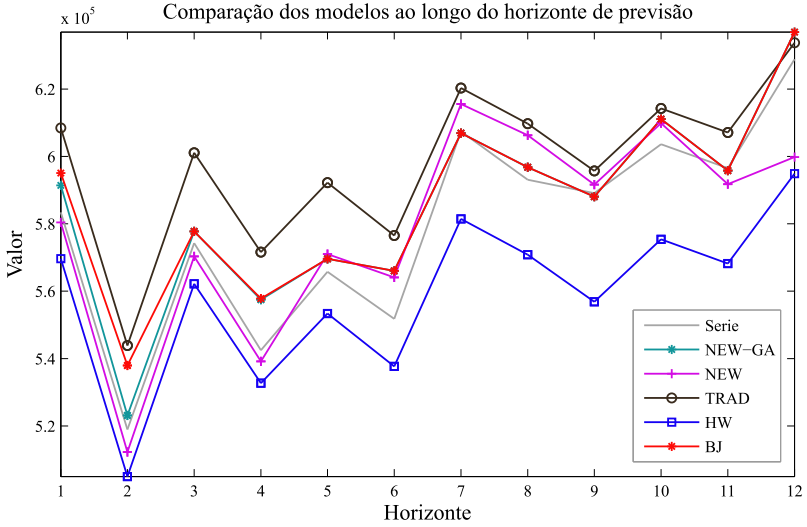
Figura 4.9: Resumo dos resultados obtidos para a série GLP.



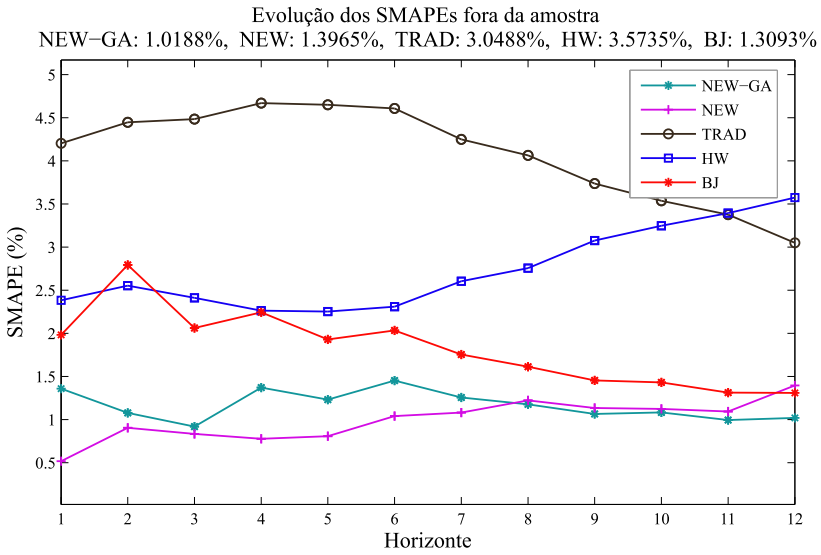
4.10(a): Evolução dos pesos de combinação



4.10(b): Previsões componentes e previsão combinada



4.10(c): Previsões dos modelos de previsão avaliados



4.10(d): Evolução dos SMAPEs para cada modelo

Figura 4.10: Resumo dos resultados obtidos para a série QAV.

4.4.1 Testes de Hipóteses para Séries Derivados do Petróleo

Para verificar se os desempenhos observados no conjunto de teste são significativamente diferentes (a favor ou não da combinação de previsores pelo modelo NEW-GA), deve-se testar, para cada um dos modelos de referência (individual/combinação), a partir do primeiro grupo de testes proposto (Seção 4.2.2), a seguinte hipótese nula (H_0): a mediana das diferenças de desempenho entre o modelo NEW-GA e o modelo de referência individual/combinação é zero.

A Tabela 4.6 exibe os resultados do primeiro grupo de testes de hipótese sugeridos na Seção 4.2.2. Nessa tabela, a coluna H_0 pode assumir três valores: **0**, se a hipótese nula não for rejeitada (intervalo de confiança incluindo zero); **1** se há indicativo de que a combinação dada pelo modelo NEW-GA é melhor do que o modelo de referência (intervalo de confiança negativo); **-1** se há indicativo de que o modelo de referência é melhor (intervalo de confiança positivo). Na última linha são exibidos os **saldos de rejeição da hipótese nula** (***srh0***); indicadores propostos em [12] e utilizados no presente trabalho, constituídos pela soma das células H_0 para cada modelo de referência; quanto maior este indicador, mais vezes o modelo NEW-GA proposto foi considerado melhor.

O indicador *srh0* pode assumir valores inteiros entre **-3** (o modelo NEW-GA é sempre pior) e **3** (o modelo NEW-GA é sempre melhor); um valor de **0** indica indiferença total com respeito ao modelo de referência. Valores entre **-1** e **1** (inclusive) constituem uma zona de indecisão, na qual não é possível apontar diferença significativa entre os métodos. A partir das considerações anteriores, pode-se chegar às conclusões da Tabela 4.7, onde a última linha informa o ***srh0*** **acumulado** (***srh0***+), i.e., a soma de todos os indicadores **positivos** observados.

Teste t					
Modelo de Referência	H_0	p-valor	inf	sup	Jarque-Bera
HW	1	0,0087	-2,0035	-0,3648	Normal
BJ	1	0,0136	-1,2266	-0,1750	Normal
TRAD	1	0,0199	-1,3101	-0,1382	Não Normal
NEW	1	0,0121	-0,5041	-0,0775	Não Normal
Teste do Sinal					
Modelo de Referência	H_0	p-valor	inf	sup	Jarque-Bera
HW	1	0,0386	-2,1910	-0,3310	Normal
BJ	0	0,1460	-1,4510	0,1490	Normal
TRAD	1	0,0386	-1,3472	-0,1710	Não Normal
NEW	1	0,0386	-0,5114	-0,1410	Não Normal
Teste de Wilcoxon					
Modelo de Referência	H_0	p-valor	inf	sup	Jarque-Bera
HW	1	0,0122	-2,0807	-0,2826	Normal
BJ	1	0,0210	-1,3357	-0,1304	Normal
TRAD	0	0,0640	-1,3130	0,0282	Não Normal
NEW	1	0,0122	-0,4826	-0,0192	Não Normal
***srh0*(HW)=3 *srh0*(BJ)=2 *srh0*(TRAD)=2 *srh0*(NEW)=3**					

Tabela 4.6: Para todos os testes são exibidos o *p-valor*, os limites de confiança inferior (*inf*) e superior (*sup*) para a mediana observada e o status do teste de normalidade *Jarque-Bera* para a distribuição estatística das diferenças de desempenho: *normal* ou *não*

.

Modelo de Referência	*srh*0	Conclusão
HW	3	O NEW-GA é melhor
BJ	2	O NEW-GA é melhor
TRAD	2	O NEW-GA é melhor
NEW	3	O NEW-GA é melhor
***srh0*+=10**		

Tabela 4.7: Conclusões para o modelo NEW-GA (séries de derivados do petróleo).

Como mencionado na Seção 4.2.2, um segundo grupo de testes é proposto de forma a avaliar não só a comparação um-para-um entre os mode-

los, mas também a comparação múltipla entre todos eles, a partir do teste da seguinte hipótese nula (H_0): Os posicionamentos obtidos por cada modelo são igualmente prováveis (ou seja, os diferentes modelos de previsão individual/combinação possuem desempenho semelhante). Neste sentido, a Tabela 4.8 expõe os resultados do teste de Friedman e Holm, baseado no desempenho tomado período a período ao longo do horizonte de previsão de cada modelo (individual/combinação) avaliado. É possível verificar que o posto médio do NEW-GA foi o menor de todos (1,75), enquanto que o do NEW foi o segundo menor (2,5). Como o modelo NEW-GA tem menor posto e o resultado do teste de Friedman aponta que há diferença significativa com respeito ao desempenho obtido por cada modelo individual/combinação de previsão (p-valor<0,05), este foi selecionado para ser o modelo de controle para comparação com os demais modelos. O teste de Holm para múltiplas comparações evidencia que o modelo NEW-GA é, com base no desempenho apresentado nas séries de derivados do petróleo, substancialmente superior aos modelos HW e TRADICIONAL, além de atestar a ausência de diferença significativa com respeito aos modelos BJ e NEW (p-valor>0,05).

i	Modelo	Posto
4	HW	4,4167
3	TRAD	3,4167
2	BJ	2,9167
1	NEW	2,5
0	NEW-GA	**1,75**

Teste	p-valor
Friedman	0,0007
Iman e Daveport	0,0001

Modelo de Referência	$z = (R_0 - R_i)/SE$	p-valor	Holm
HW	4,131182	0,000036	0,0125
TRAD	2,581989	0,009823	0,016667
BJ	1,807392	0,070701	0,025
NEW	1,161895	0,245278	0,05

Tabela 4.8: Resultados do teste de Friedman e Holm para a comparação entre os modelos individuais/combinação (séries de derivados do petróleo).

4.4.2 Resumo

Os resultados obtidos a partir do primeiro estudo de casos (Seção 4.4), trazem indícios razoáveis de que a técnica de treinamento híbrida para redes MLP, aqui proposta, pode agregar valor ao modelo NEW original. Esta conclusão vem da análise conjunta dos resultados nas Tabelas 4.2, 4.7 e 4.8, assim como do indicador *srh0* que apresenta valores altos para cada um dos modelos de previsão individuais/combinação: **3** para o modelo HW, **2** para o modelo Box & Jankins, **2** para o melhor modelo tradicional de ponderação e **3** para o modelo NEW original, todos numa escala que vai até **3**. De fato, estes resultados reforçam uma conclusão recorrente na literatura: *há vantagem prática em combinar previsores* [12].

Nota-se que, pela avaliação feita a partir do teste de Holm, o modelo NEW-GA não apresentou evidência significativa que demonstre superioridade com respeito dos modelos BJ e NEW. Esse comportamento deve-se, em parte, pelo fato do modelo NEW-GA apresentar o pior ranqueamento no horizonte $h<3$ (Figura 4.5), enquanto que os modelos BJ e NEW jamais ocuparam aquela posição ao longo do horizonte de previsão.

Ao considerar o teste de Friedman, percebe-se que o modelo NEW-GA apresenta-se no primeiro lugar. Observando-se isoladamente os desempenhos totais fora da amostra (acumulados em 12 meses), o modelo NEW-GA também apresentou os melhores resultados: SMAPEs de **2,32** para DIESEL, **6,17** para GASOLINA, **1,43** para GLP e **1,02** para QAV (Tabela 4.2).

4.5 CASO 2: Séries da Competição NN3

A competição NN3 foi uma competição entre métodos de previsão conduzida durante os anos de 2006 e 2007, criada principalmente para avaliar modelos baseados em redes neurais ou inteligência computacional [77].

O banco de dados da NN3 é constituído, na sua forma completa, de **111** séries temporais mensais, relacionadas a atividades de transporte tais como: tráfego em autoestradas, tráfego de carros em túneis, tráfego em pedágios, tráfego de pessoas em estações de metrô, voos domésticos, entregas de importação, cruzamento de fronteiras, fluxo em dutos e transporte ferroviário. Há também uma versão **reduzida** do banco, que reúne apenas as últimas **11** séries do conjunto (apresentadas no Anexo B), sendo esta a versão utilizada como parte do segundo estudo de casos no presente trabalho. A Tabela 4.9 mostra de forma resumida os principais detalhes da competição.

Número de competidores	25
Número de *benchmarks* de previsão	8
Quantidade de séries	111
Quantidade de séries na versão reduzida	11
Periodicidade	Mensal
Tamanho mínimo/máximo das séries	69/114
Métrica de avaliação	SMAPE
Horizonte de previsão	18

Tabela 4.9: Principais detalhes da competição NN3.

Basicamente, três passos deviam ser seguidos para participar na NN3:

1. Desenvolver um método de previsão bem documentado, que possa ser automatizado;

2. Testar o método em todas as séries do banco de dados fornecido pelos organizadores, produzindo para cada série de tamanho τ, previsões para $\tau + 1$, $\tau + 2$, ..., $\tau + 18$;

3. Submeter as previsões geradas para julgamento.

Conhecedores das realizações fora da amostra, os julgadores puderam calcular, para cada competidor, a média (para todas as séries) dos desempenhos SMAPE acumulados **18** passos à frente. Foi declarado vencedor o método com melhor desempenho médio. A Tabela 4.10 apresenta os 10 melhores colocados na competição, considerando os resultados para as 11 séries da versão reduzida.

Colocação	Autor	SMAPE
-	*CI Benchmark* - Theta AI (Nikolopoulos)	13,07%
-	*Stat. Benchmark* - Autobox (Reily)	13,49%
-	*Stat. Benchmark* - ForecastPro (Stellwagen)	13,52%
1	**Yan**	**13,68%**
-	*Stat. Benchmark* - Theta (Nikolopoulos)	13,70%
2	llies Jäger, Kosuchinas, Rincon, Sakenas e Vaskevcius	14,26%
3	Chen, Yao	14,46%
4	Yousefi, Miromendi e Lucas	14,49%
5	Ahmed, Atiya, Gayar e El-Shishiny	14,52%
6	Flores, Anaya, Ramirez e Morales	15,00%
7	Adeodato, Vasconcelos, Arnaud, Chunha e Monteiro	15,10%
-	*Stat. Contender* - Wildi	15,32%
8	Luna, Soares e Ballini	15,35%
9	Theodosiou e Swamy	16,19%
10	Hwang, Song e Kasabov	16,31%

Tabela 4.10: Dez primeiros melhores colocados na competição NN3, considerando os resultados para as 11 séries da versão reduzida.

Verifica-se, dentre as 10 melhores colocações, a presença de cinco benchmarks de previsão (quatro estatísticos e um de inteligência computacional), incluindo alguns dos modelos de previsão mais reconhecidos pela literatura científica, como o método Theta [86], assim como pela indústria (Autobox e ForecastPro). O competidor campeão (excluindo os benchmarks de previsão) é o de Yan [87]. Este propõe uma Rede Neural baseada em Modelos Lineares Generalizados (GRNN, [88]), apoiado por uma etapa prévia de pré-processamento sobre as observações da série a partir da identificação e tratamento de padrões na tendência e sazonalidade da mesma.

Seguindo o procedimento aplicado sobre o conjunto de séries no primeiro estudo de casos (Seção 4.4), com os modelos individuais de previsão ajustados (HW e BJ, podendo ser substituídos pelos correspondentes previsores limiares), foram geradas previsões até 18 passos a frente. Posteriormente, foram estimados os vetores de pesos (encarregados de ponderar cada vetor de previsões disponível no instante de tempo $\tau + h$), a partir dos métodos tradicionais de ponderação, pelo modelo NEW original e pelo modelo NEW otimizado (NEW-GA), obtendo-se em cada caso um vetor de previsões combinadas que é comparado com o trecho real da série de teste, de modo a avaliar o desempenho de cada modelo. A Tabela 4.11 exibe os desempenhos **totais** - considerando cada uma das 11 séries na competição NN3 - obtidos fora da amostra (18 meses)

para cada modelo de previsão individual e de combinação, incluindo o melhor método de combinação tradicional reportado em [12]. Da mesma forma que no primeiro estudo de casos, a métrica de desempenho utilizada foi o SMAPE.

Série	HW	BJ	TRAD	NEW	NEW-GA
NN3-101	**1,80**	1,99	1,86	1,97	1,85
NN3-102	30,63	12,64	17,73	22,09	**9,17**
NN3-103	29,60	44,01	32,82	28,49	**25,19**
NN3-104	6,11	4,96	5,10	5,20	**4,90**
NN3-105	**1,53**	2,67	2,64	3,09	1,77
NN3-106	5,38	5,31	4,78	4,88	**4,52**
NN3-107	5,10	5,87	3,68	3,36	**2,81**
NN3-108	36,29	29,58	30,80	29,74	**23,84**
NN3-109	7,77	7,02	16,47	7,55	**6,23**
NN3-110	30,48	33,61	29,67	24,53	**21,16**
NN3-111	11,14	17,75	15,09	11,17	**10,63**
Média	15,08	15,04	14,60	12,92	**10,19**
Desvio Padrão	13,57	14,41	12,03	10,99	**8,96**

Tabela 4.11: Desempenhos totais para as 11 séries da competição NN3 (versão reduzida). Valores em negrito indicam os melhores resultados obtidos em termos de desempenho.

A Figura 4.11 apresenta o diagrama de dispersão correspondente aos desempenhos totais registrados na Tabela 4.11. Neste gráfico é possível observar, que, da mesma forma que no primeiro estudo de casos, o modelo NEW-GA teve a menor dispersão de erro com respeito ao desempenho dos outros modelos, ao mesmo tempo que apresentou a média de erro mais baixa ao longo das 11 séries avaliadas.

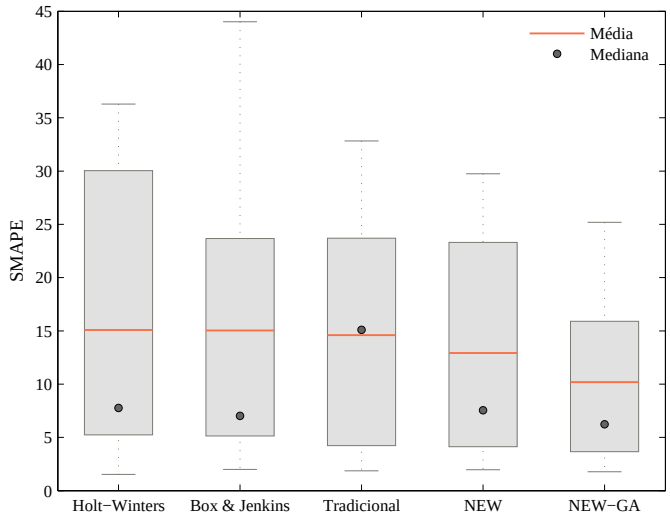


Figura 4.11: SMAPEs fora da amostra para as séries da competição NN3 (versão reduzida): *boxplots*.

Da mesma forma como foi apresentado no primeiro estudo de casos, a Tabela 4.12 exibe a configuração do modelo NEW-GA obtida para cada série avaliada da competição NN3.

Série	Hiper–parâmetros	Arquitetura RNA	Tempo de Execução
NN-101	HW+/HW-/BJ+/BJ- Janela expansiva	5:12:4	1.63 h
NN-102	HW+/HW-/BJ+/BJ- Janela expansiva	5:20:4	1.52 h
NN-103	HW/BJ Janela mínima ν=3	3:19:2	1.10 h
NN-104	HW/BJ Janela mínima ν=3	3:10:2	1.05 h
NN-105	HW/BJ Janela expansiva	3:20:2	1.07 h
NN-106	HW+/HW-/BJ+/BJ- Janela mínima ν=5	5:18:4	1.90 h
NN-107	HW+/HW-/BJ+/BJ- Janela mínima ν=5	5:20:4	1.72 h
NN-108	HW+/HW-/BJ+/BJ- Janela mínima ν=5	5:12:4	1.92 h
NN-109	HW+/HW-/BJ+/BJ- Janela mínima ν=5	5:18:4	1.68 h
NN-110	HW+/HW-/BJ+/BJ- Janela mínima ν=5	5:14:4	1.72 h
NN-111	HW+/HW-/BJ+/BJ- Janela mínima ν=5	5:4:4	1.76 h

Tabela 4.12: Modelos NEW-GA obtidos: A coluna hiper–parâmetros indica os previsores individuais e a janela de tempo utilizada para a estimação dos conjuntos de treinamento, validação e teste. A coluna da arquitetura RNA indica a estrutura final da rede MLP, onde o número de neurônios oculto é determinado pelo algoritmo de treinamento híbrido proposto. A última coluna indica o tempo de execução utilizado pelo modelo NEW-GA durante a modelagem da rede para cada série avaliada. Todos os experimentos foram executados em um PC Windows 7 com processador Intel i7 de 3.6 GHz.

A Tabela 4.13 e a Figura 4.12, por sua parte, exibem a evolução dos SMAPEs médios acumulativos no conjunto de teste, ao longo do horizonte (h) de 18 meses.

h	HW	BJ	TRAD	NEW	NEW-GA
1	10,66	12,84	12,99	8,40	9,91
2	9,32	12,93	12,23	8,50	9,51
3	8,59	13,79	12,24	9,32	8,74
4	8,71	13,60	12,37	9,73	8,42
5	9,39	13,32	12,84	10,07	8,94
6	10,07	12,99	12,96	10,61	8,83
7	10,78	12,85	13,18	10,66	8,84
8	11,03	13,42	12,99	11,22	8,72
9	11,09	14,96	13,16	11,80	9,06
10	11,14	15,88	14,02	12,06	8,97
11	12,17	16,09	14,49	12,63	9,41
12	12,98	16,24	15,02	13,08	9,92
13	13,88	16,44	15,37	13,56	10,13
14	14,19	16,16	15,26	13,51	10,19
15	14,32	15,56	14,79	12,98	9,85
16	14,73	15,22	14,64	12,92	9,92
17	14,82	15,00	14,51	12,87	9,95
18	**15,08**	**15,04**	**14,60**	**12,92**	**10,19**

Tabela 4.13: Evolução do SMAPE médio acumulativo ao longo do horizonte de previsão para as séries da competição NN3 (versão reduzida).

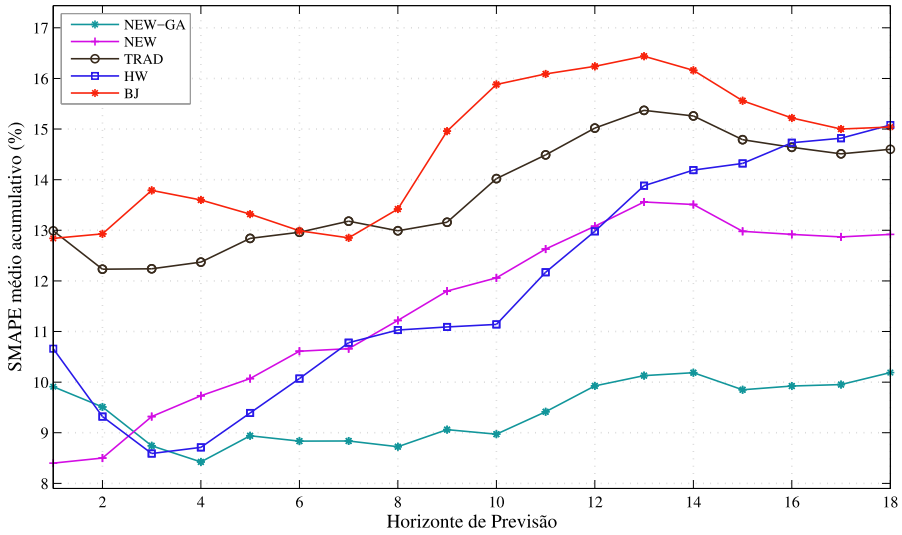


Figura 4.12: Evolução do SMAPE médio acumulativo ao longo do horizonte de previsão para as séries da competição NN3 (versão reduzida).

É interessante observar, a partir da Figura 4.12, que o NEW-GA é o modelo mais estável dos modelos em análise, isto dado pela baixa variabilidade na estimação dos seus erros ao longo do horizonte de previsão. Observa-se também que os modelos de combinação (Tradicional, NEW e NEW-GA) conseguem novamente vencer aos modelos individuais de previsão, da mesma forma como foi observado nas séries avaliadas no primeiro estudo de casos.

A Tabela 4.14 exibe as diferenças de desempenho médio tomadas período a período, fora da amostra, na ordem "erro do modelo NEW-GA **menos** erro do modelo de referência". Quanto mais negativa for esta diferença, melhor o modelo NEW-GA.

Calculadas as diferenças, todos os testes de hipótese sugeridos na Seção 4.2.2 foram executados, sempre com o objetivo de verificar as seguintes hipóteses nulas:

- A mediana das diferenças de desempenho (última linha da Tabela 4.14) entre o modelo NEW-GA e o modelo individual/combinação de previsão é zero.
- Os posicionamentos obtidos por cada modelo individual/combinação de previsão são igualmente prováveis;

dependendo se está sendo aplicado o primeiro ou segundo grupo de testes, respectivamente.

Antes de prosseguir com os resultados fornecidos pelos testes de hipótese, cabe ressaltar que, do mesmo modo como foi apresentado no primeiro estudo

h	HW	BJ	TRAD	NEW
1	-0,75	-2,93	-3,08	1,51
2	0,19	-3,42	-2,72	1,01
3	0,15	-5,05	-3,50	-0,58
4	-0,29	-5,18	-3,95	-1,31
5	-0,45	-4,38	-3,90	-1,13
6	-1,24	-4,16	-4,13	-1,78
7	-1,94	-4,01	-4,34	-1,82
8	-2,31	-4,70	-4,27	-2,50
9	-2,03	-5,90	-4,10	-2,73
10	-2,17	-6,91	-5,05	-3,09
11	-2,76	-6,68	-5,08	-3,22
12	-3,01	-6,32	-5,10	-3,16
13	-3,75	-6,31	-5,24	-3,43
14	-4,01	-5,98	-5,08	-3,33
15	-4,47	-5,71	-4,94	-3,13
16	-4,81	-5,30	-4,72	-3,00
17	-4,87	-5,05	-4,56	-2,92
18	-4,89	-4,85	-4,41	-2,73
Média	**-2,41**	**-5,16**	**-4,34**	**-2,07**
Mediana	**-2,24**	**-5,11**	**-4,38**	**-2,73**

Tabela 4.14: Diferenças de desempenho médio do modelo NEW-GA com respeito dos modelos de referência (individuais/combinação), ao longo do horizonte de previsão para as séries da competição NN3 (versão reduzida).

de casos (Seção 4.4), o Anexo C inclui, para cada série avaliada no segundo estudo de casos, o progresso do algoritmo durante diferentes etapas do processo evolutivo. O Anexo D, por sua vez, exibe a evolução dos pesos de combinação ao longo do horizonte de previsão, considerando a solução escolhida a partir do conjunto de soluções ótimas fornecido pelo algoritmo proposto. Nas figuras do Anexo D apresenta-se também, para cada série, as previsões componentes e a previsão combinada que é gerada dinamicamente quando os vetores de previsão em cada ponto do horizonte são ponderados a partir dos correspondentes vetores de pesos estimados naquele ponto. Exibem-se igualmente, as previsões dos modelos (individuais/combinação) avaliados, assim como a evolução dos seus respetivos SMAPEs ao longo do horizonte de previsão.

4.5.1 Testes de Hipóteses para Séries da Competição NN3

Do mesmo modo que na Seção 4.4.1, para o primeiro grupo de testes proposto, deve-se testar com respeito a cada modelo de referência (individuais/combinação), a seguinte hipótese nula (H_0): a mediana das diferenças de desempenho entre o modelo NEW-GA e o modelo individual/combinação é zero.

A Tabela 4.15 exibe os resultados do primeiro grupo de testes sugeridos na Seção 4.2.2. Com base nos valores observados nessa tabela para os indicadores *srh0* (Seção 4.4.1), comprova-se a significância da abordagem NEW-GA (p-valores $< 0{,}05$), resultando nas conclusões apresentadas na Tabela 4.16.

Teste t					
Modelo de Referência	H_0	p-valor	inf	sup	Jarque-Bera
HW	1	0,0001	-3,2935	-1,5326	Normal
BJ	1	0,0001	-5,7065	-4,6062	Normal
TRAD	1	0,0001	-4,7010	-3,9817	Normal
NEW	1	0,0001	-2,8057	-1,3415	Não Normal
Teste do Sinal					
Modelo de Referência	H_0	p-valor	inf	sup	Jarque-Bera
HW	1	0,0013	-4,0050	-0,7516	Normal
BJ	1	0,0001	-5,9750	-4,3804	Normal
TRAD	1	0,0001	-5,0485	-3,9467	Normal
NEW	1	0,0013	-3,1306	-1,3067	Não Normal
Teste de Wilcoxon					
Modelo de Referência	H_0	p-valor	inf	sup	Jarque-Bera
HW	1	0,0001	-3,4161	-1,3779	Normal
BJ	1	0,0001	-5,7634	-4,6023	Normal
TRAD	1	0,0001	-4,7294	-3,9816	Normal
NEW	1	0,0001	-2,9734	-1,2001	Não Normal
***srh0*(HW)=3**	***srh0*(BJ)=3**		***srh0*(TRAD)=3**		***srh0*(NEW)=3**

Tabela 4.15: Para todos os testes são exibidos o *p-valor*, os limites de confiança inferior (*inf*) e superior (*sup*) para a mediana observada e o status do teste de normalidade *Jarque-Bera* para a distribuição estatística das diferenças de desempenho: *normal* ou *não*.

Modelo de Referência	$srh0$	Conclusão
HW	3	O NEW-GA é melhor
BJ	3	O NEW-GA é melhor
TRAD	3	O NEW-GA é melhor
NEW	3	O NEW-GA é melhor
	***srh0*+=12**	

Tabela 4.16: Conclusões para o modelo NEW-GA nas séries da competição NN3 (versão reduzida).

A Tabela 4.17 exibe os resultados do teste de Friedman e Holm para comparar simultaneamente os 5 modelos individuais/combinação de previsão. De fato, como o modelo NEW-GA obteve o menor posto (1,2222), este é selecionado para ser objeto de comparação com os demais modelos. Neste sentido, este auferiu os melhores resultados (p-valor<0,05) com respeito a todos os modelos de previsão sob análise. O teste de Holm para múltiplas comparações evidencia que o modelo NEW-GA é, para as séries da competição NN3, substancialmente superior tanto aos modelos individuais de previsão HW e BJ, quanto aos modelos de combinação incluindo o modelo NEW original.

i	Modelo	Posto	
4	BJ	4,8333	
3	TRAD	3,9444	
2	HW	2,6111	
1	NEW	2,3889	
0	NEW-GA	**1,2222**	
Teste		p-valor	
Friedman		<0,0001	
Iman e Daveport		<0,0001	
Modelo de Referência	$z = (R_0 - R_i)/SE$	p-valor	Holm
BJ	6,851602	0,000001	0,0125
TRAD	5,165054	0,000001	0,016667
HW	2,635231	0,008408	0,025
NEW	2,213594	0,026857	0,05

Tabela 4.17: Resultados do teste de Friedman e Holm para a comparação entre os modelos individuais/combinação de previsão.

4.5.2 Resumo

Considerando o critério de avaliação período a período, realizado pelo teste estatístico da variável **diferença de desempenho**, os resultados deixam a abordagem NEW-GA em vantagem considerável sobre os outros modelos comparados (incluindo o modelo NEW original). Outros critérios indicando vantagem da abordagem proposta foram: (i) o NEW-GA apresentou o menor SMAPE médio total fora da amostra (acumulado em 18 meses) - **10,19** (Tabela 4.13) - melhorando o desempenho do modelo original em **2,73** pp e (ii) foi o modelo que prevaleceu no maior número de séries da competição: 9 em 11 (Tabela 4.11). Em particular, a análise da Tabela 4.11 mostra que o NEW-GA foi apenas superado nas séries de números **1** e **5**[4] (Anexo B) nas quais prevaleceu o modelo de previsão individual HW.

O teste de Friedman, por sua parte, colocou o modelo NEW-GA novamente no primeiro lugar, conseguindo evidência significativa (p-valor<0,0001) que demonstrou superioridade com respeito tanto aos modelos individuais quanto aos modelos de combinação avaliados.

4.5.3 Comparação com os Demais Modelos

A partir dos melhores modelos NEW-GA, obtidos pela abordagem de treinamento híbrida aqui proposta, captam-se os resultados referentes ao SMAPE obtido no conjunto de séries da versão reduzida da competição NN3, para efeito da comparação com os outros modelos. A Tabela 4.18 apresenta a colocação do modelo NEW-GA com respeito aos resultados da competição reduzida[5].

Nota-se a presença do modelo NEW-GA na primeira colocação, com um SMAPE de aproximadamente **3,49** pp menor do que o do primeiro colocado, uma Rede Neural baseada em Modelos Lineares Generalizados proposta por Yan. Contudo, deve-se mencionar que nos resultados do período da competição e na literatura disponível apos a competição, não foi observado modelo algum do tipo SNE, como é o caso do modelo NEW-GA. É importante lembrar também que os modelos de combinação testados no presente trabalho (incluindo o modelo NEW-GA), não consideram nenhum tipo de **pré-processamento** estatístico (e.g. eliminação de *outliers*) sobre as séries originais. Este pode não ser o caso de alguns dos métodos apresentados no *ranking* final da competição.

[4] A análise desse resultado é uma das sugestões para estudos futuros.
[5] http:\\www.neural-forecasting-competition.com\NN3\results.htm

Colocação	Autor	SMAPE
1	**NEW-GA**	**10,19%**
-	*CI Benchmark* - Theta AI (Nikolopoulos)	13,07%
-	*Stat. Benchmark* - Autobox (Reily)	13,49%
-	*Stat. Benchmark* - ForecastPro (Stellwagen)	13,52%
2	Yan	13,68%
-	*Stat. Benchmark* - Theta (Nikolopoulos)	13,70%
3	llies Jäger, Kosuchinas, Rincon, Sakenas e Vaskevcius	14,26%
4	Chen, Yao	14,46%
5	Yousefi, Miromendi e Lucas	14,49%
6	Ahmed, Atiya, Gayar e El-Shishiny	14,52%
7	Flores, Anaya, Ramirez e Morales	15,00%
8	Adeodato, Vasconcelos, Arnaud, Chunha e Monteiro	15,10%
-	*Stat. Contender* - Wildi	15,32%
9	Luna, Soares e Ballini	15,35%
10	Theodosiou e Swamy	16,19%

Tabela 4.18: Dez primeiros melhores colocados na competição NN3, considerando os resultados para as 11 séries da versão reduzida.

5
Conclusões e Considerações Finais

Esta dissertação apresentou uma nova abordagem de treinamento para redes neurais do tipo MLP aplicadas ao núcleo do sistema NEW. O Neural Expert Weighting - Genetic Algorithm (NEW-GA), como foi batizado, é uma metodologia para geração dinâmica de vetores de ponderação em sistemas multi-previsores, baseada em modelos de redes MLP otimizadas a partir de uma abordagem de treinamento híbrida, que combina mecanismos de busca local e global numa mesma técnica de treinamento. Foram exibidas suas principais caraterísticas, tais como (*i*) gerar, de forma automática, modelos neurais de ponderação competitivos sem a necessidade de um conhecimento profundo do treinamento de redes; (*ii*) agregar valor ao procedimento NEW original em relação a seu desempenho e/ou à automação no processo de configuração da rede MLP; e (*iii*) ser capaz de identificar caminhos mais promissores para se chegar a arquiteturas finais de tamanho adequado de acordo com a complexidade do problema.

O modelo proposto foi avaliado em dois estudos de casos diferentes, considerando séries de vendas mensais dos produtos derivados do petróleo mais importantes no mercado brasileiro, assim como o conjunto reduzido de séries da competição NN3. Os resultados obtidos, como discutidos nas Seções 4.4.2 e 4.5.2, dão suporte às seguintes conclusões:

1. A hibridização entre técnicas de treinamento local e global, no processo de configuração dos parâmetros internos da rede, permitiram percorrer regiões do espaço de busca onde os modelos neurais conseguiram identificar de melhor forma as relações existentes entre as entradas e saídas da rede, isto é, serem capazes de identificar a contribuição que cada previsor pode fornecer dependendo do horizonte de previsão avaliado.

2. O fato de considerar uma segunda métrica de desempenho (além da métrica obtida diretamente pelas saídas da rede, Seção 3.2.3), como parte do vetor de funções objetivo, encarregado de direcionar o processo evolutivo, forneceu um ganho considerável na capacidade de generalização das redes. Este comportamento, em parte, pode ser atribuído ao fato dessa métrica avaliar diretamente a capacidade preditiva do modelo, uma vez

que a sua estimação considera as diferenças entre os valores previstos e os valores reais da série. Este comportamento reforça também a consideração exposta na Seção 2.2.2, referindo-se a que o modelo NEW original poderia apresentar perda no desempenho quando considerada apenas a minimização de uma das métricas de erro disponíveis.

3. A terceira conclusão vêm ao encontro de uma afirmativa recorrente na literatura: há vantagem prática em combinar previsores (Seção 1). Essa conclusão é reforçada pelos resultados obtidos, a partir dos estudos de caso propostos, nos quais foi observado que nenhum modelo individual conseguiu melhorar o desempenho médio de algum dos modelos de combinação.

A configuração automática do núcleo no sistema NEW, a partir da abordagem de treinamento híbrida proposta, gerou um melhor comportamento nos modelos de ponderação dinâmica ao longo do horizonte de previsão (12 e 18 meses à frente). Isto se deve às caraterísticas particulares com que cada técnica consegue percorrer o espaço de busca, durante o processo de otimização.

A abordagem proposta conseguiu melhorar notavelmente o desempenho do modelo NEW original, fornecendo, assim, resultados positivos nos testes de hipótese realizados. A ponderação dinâmica de pesos fornecida pelo NEW-GA, portanto, conseguiu também melhores resultados do que os modelos de ponderação tradicionais.

Há extensas possibilidades para estudos futuros relacionados ao modelo proposto na presente dissertação. De forma geral, elas podem ser organizadas nas seguintes vertentes:

- Alteração dos previsores componentes no esquema de combinação, podendo, por exemplo, ser utilizados modelos não lineares baseados em técnicas de IC, como é o caso das RNAs, ou de modelos de previsão inspirados na lógica Fuzzy;
- Explorar a influência dos hiperparâmetros no desempenho do modelo. Assim, por exemplo, pode-se testar novos métodos para geração de pesos históricos e novas janelas de tempo;
- Avaliação do modelo com séries de outra natureza (e.g. semanais, diárias, ou mesmo horárias);
- Explorar a influência dos parâmetros no desempenho do modelo. Pode-se, por exemplo, avaliar o comportamento do algoritmo quando outras métricas de convergência são consideradas; avaliar a influência do tamanho da população no desempenho ou ainda testar o modelo com diferente numero de gerações.

Referências Bibliográficas

[1] RODRIGUES, L. C; SILVA, P ; LINDEN, R. **Séries temporais no consumo de energia elétrica no estado do rio de janeiro**. Revista Visões, 2, 2007.

[2] ZOMAYA, A. Y; ANDERSON, J. A; FOGEL, D. B; MILBURN, G. J ; ROZENBERG, G. **Nonconventional computing paradigms in the new millenium: A roundtable**. Computing in Science & Engineering, (6):82–99, 2001.

[3] HORNIK, K; STINCHCOMBE, M ; WHITE, H. **Multilayer feedforward networks are universal approximators**. Neural networks, 2(5):359–366, 1989.

[4] CORTEZ, P. **Modelos Inspirados na Natureza para a Previsão de Séries Temporais. 2002. 188 f**. PhD thesis, Tese (Doutorado em Informática)–Departamento de Informática, Universidade do Minho, Braga, 2002.

[5] WERNER, L. **Um modelo composto para realizar previsão de demanda através da integração da combinação de previsões e do ajuste baseado na opinião**. PhD thesis, Tese (Doutorado em Produção–Departamento de Produção, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2004.

[6] CHASE JR, C. W. **Demand-driven forecasting: a structured approach to forecasting**. John Wiley & Sons, 2013.

[7] TIMMERMANN, A. **Forecast combinations**. Handbook of economic forecasting, 1:135–196, 2006.

[8] BATES, J. M; GRANGER, C. W. **The combination of forecasts**. Or, p. 451–468, 1969.

[9] BREIMAN, L. **Bagging predictors**. Machine learning, 24(2):123–140, 1996.

[10] BAUER, E; KOHAVI, R. **An empirical comparison of voting classification algorithms: Bagging, boosting, and variants**. Machine learning, 36(1-2):105–139, 1999.

[11] FREUND, Y; SCHAPIRE, R ; ABE, N. **A short introduction to boosting**. Journal-Japanese Society For Artificial Intelligence, 14(771-780):1612, 1999.

[12] VALLE DOS SANTOS, R. D. O; VELLASCO, M. M. **Neural expert weighting**. Expert Systems with Applications: An International Journal, 42(22):8625–8636, 2015.

[13] ALADAG, C. H; EGRIOGLU, E ; YOLCU, U. **Forecast combination by using artificial neural networks**. Neural Processing Letters, 32(3):269–276, 2010.

[14] BAJO, M; UMGIESSER, G. **Storm surge forecast through a combination of dynamic and neural network models**. Ocean Modelling, 33(1):1–9, 2010.

[15] DONALDSON, R. G; KAMSTRA, M. **Forecast combining with neural networks**. Journal of Forecasting, 15(1):49–61, 1996.

[16] ZHANG, Y; SHAN, R; WANG, H ; JIN, F. **A new wavelet-neural network-arima shares index combination forecast model**. In: AUTOMATIC CONTROL AND ARTIFICIAL INTELLIGENCE (ACAI 2012), INTERNATIONAL CONFERENCE ON, p. 199–201. IET, 2012.

[17] ADHIKARI, R. **A neural network based linear ensemble framework for time series forecasting**. Neurocomputing, 157:231–242, 2015.

[18] PRUDENCIO, R. B; LUDERMIR, T. B. **Learning weights for linear combination of forecasting methods**. In: NEURAL NETWORKS, 2006. SBRN'06. NINTH BRAZILIAN SYMPOSIUM ON, p. 113–118. IEEE, 2006.

[19] SÁNCHEZ, I. **Adaptive combination of forecasts with application to wind energy**. International Journal of Forecasting, 24(4):679–693, 2008.

[20] YANG, Y. **Combining forecasting procedures: some theoretical results**. Econometric Theory, 20(01):176–222, 2004.

[21] MARTINS, V. L. M; WERNER, L. **Forecast combination in industrial series: A comparison between individual forecasts and its combinations with and without correlated errors**. Expert Systems with Applications, 39(13):11479–11486, 2012.

[22] HAYKIN, S. **Neural networks. 1999**. A Comprehensive Foundation, 1995.

[23] YAO, X; LIU, Y. **Evolving artificial neural networks for medical applications**. In: PROC. OF, p. 1–16. Citeseer, 1995.

[24] YAO, X. **Evolving artificial neural networks**. Proceedings of the IEEE, 87(9):1423–1447, 1999.

[25] BRANKE, J. **Evolutionary algorithms for neural network design and training**. In: IN PROCEEDINGS OF THE FIRST NORDIC WORKSHOP ON GENETIC ALGORITHMS AND ITS APPLICATIONS. Citeseer, 1995.

[26] GOLDBERG, D. **Genetic algorithms in search optimization and machine learning**. Addison Wesley Reading Menlo Park, 1989.

[27] HOLLAND, J. H. **Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence.** U Michigan Press, 1975.

[28] DA CRUZ, A. V. A. **Algoritmos evolutivos com inspiração quântica para problemas com representação numérica**. PhD thesis, PUC-Rio, 2007.

[29] YAO, X; LIU, Y. **A new evolutionary system for evolving artificial neural networks**. Neural Networks, IEEE Transactions on, 8(3):694–713, 1997.

[30] YAO, X; LIU, Y. **Making use of population information in evolutionary artificial neural networks**. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 28(3):417–425, 1998.

[31] PUMA-VILLANUEVA, W. J; ZUBEN, F. J. V. **Evolving arbitrarily connected feedforward neural networks via genetic algorithms**. In: NEURAL NETWORKS (SBRN), 2010 ELEVENTH BRAZILIAN SYMPOSIUM ON, p. 127–132. IEEE, 2010.

[32] RUMELHART, D; HINTON, G ; WILLIAMS, R. **Learning internal representation by error propagation and parallel distributed processing**, 1986.

[33] HERTZ, J. A; KROGH, A. S ; PALMER, R. G. **Introduction to the theory of neural computation**, 1991.

[34] MØLLER, M. F. **A scaled conjugate gradient algorithm for fast supervised learning**. Neural networks, 6(4):525–533, 1993.

[35] HORNE, B. G. **Progress in supervised neural networks**. Signal Processing Magazine, IEEE, 10(1):8–39, 1993.

[36] DE GOOIJER, J. G; HYNDMAN, R. J. **25 years of time series forecasting**. International Journal of Forecasting, 22(3):443–473, 2006.

[37] JOSE, V. R. R; WINKLER, R. L. **Simple robust averages of forecasts: Some empirical results**. International Journal of Forecasting, 24(1):163–169, 2008.

[38] CLEMEN, R. T. **Combining forecasts: A review and annotated bibliography**. International Journal of Forecasting, 5(4):559–583, 1989.

[39] JOHNSON, R; WICHERN, D. **Applied multivariate statistical analysis**. Upper Saddle River, NJ, 2007.

[40] GILL, P. E; MURRAY, W; SAUNDERS, M. A ; WRIGHT, M. H. **Procedures for optimization problems with a mixture of bounds and general linear constraints**. ACM Transactions on Mathematical Software (TOMS), 10(3):282–298, 1984.

[41] HENDERSON, C. R. **Best linear unbiased estimation and prediction under a selection model**. Biometrics, p. 423–447, 1975.

[42] NEWBOLD, P; GRANGER, C. W. **Experience with forecasting univariate time series and the combination of forecasts**. Journal of the Royal Statistical Society. Series A (General), p. 131–165, 1974.

[43] STOCK, J. H; WATSON, M. W. **A comparison of linear and nonlinear univariate models for forecasting macroeconomic time series**. Technical report, National Bureau of Economic Research, 1998.

[44] WITTEN, I. H; FRANK, E. **Data Mining: Practical machine learning tools and techniques**. Morgan Kaufmann, 2005.

[45] CYBENKO, G. **Approximation by superpositions of a sigmoidal function**. Mathematics of control, signals and systems, 2(4):303–314, 1989.

[46] DE PÁDUA BRAGA, A; DE LEON FERREIRA, A. C. P ; LUDERMIR, T. B. **Redes neurais artificiais: teoria e aplicações**. LTC Editora, 2007.

[47] ZHANG, G; PATUWO, B. E ; HU, M. Y. **Forecasting with artificial neural networks:: The state of the art**. International journal of forecasting, 14(1):35–62, 1998.

[48] TZAFESTAS, S; TZAFESTAS, E. **Computational intelligence techniques for short-term electric load forecasting**. Journal of Intelligent and Robotic Systems, 31(1-3):7–68, 2001.

[49] ADYA, M; COLLOPY, F. **How effective are neural networks at forecasting and prediction? a review and evaluation**. J. Forecasting, 17:481–495, 1998.

[50] KASABOV, N. K. **Foundations of neural networks, fuzzy systems, and knowledge engineering**. Marcel Alencar, 1996.

[51] CYBENKO, G. **Continuous valued neural networks with two hidden layers are sufficient**. Center for Supercomputing Research and Development - University of Illinois, 1988.

[52] KOZA, J. R. **Hierarchical genetic algorithms operating on populations of computer programs.** In: IJCAI, p. 768–774. Citeseer, 1989.

[53] DEB, K. **Multi-objective optimization using evolutionary algorithms**, volumen 16. John Wiley & Sons, 2001.

[54] EIBEN, A. E; SMITH, J. E. **Introduction to evolutionary computing**, volumen 53. Springer, 2003.

[55] GOLDBERG, D. E. **The design of innovation: Lessons from and for competent genetic algorithms**. Springer Science & Business Media, 2013.

[56] FACELI, K; LORENA, A. C; GAMA, J ; CARVALHO, A. **Inteligência artificial: Uma abordagem de aprendizado de máquina**. Rio de Janeiro: LTC, 2011.

[57] EIBEN, A. E; SCHIPPERS, C. A. **On evolutionary exploration and exploitation**. Fundamenta Informaticae, 35(1-4):35–50, 1998.

[58] DE JONG, K. A. **Evolutionary computation: a unified approach**. MIT press, 2006.

[59] MICHALEWICZ, Z; HARTLEY, S. J. **Genetic algorithms+ data structures= evolution programs**. Mathematical Intelligencer, 18(3):71, 1996.

[60] VON ZUBEN, F. J. **Computação evolutiva: uma abordagem pragmática**. Anais da I Jornada de Estudos em Computação de Piracicaba e Região (1a JECOMP), 1:25–45, 2000.

[61] DUCH, W; KORCZAK, J. **Optimization and global minimization methods suitable for neural networks**. Neural computing surveys, 2:163–212, 1998.

[62] MONTANA, D. J. **Neural network weight selection using genetic algorithms**. Intelligent Hybrid Systems, 8(6):12–19, 1995.

[63] RUMELHART, D. E; MCCLELLAND, J. L; GROUP, P. R ; OTHERS. **Parallel distributed processing**, volumen 1. IEEE, 1988.

[64] MONTANA, D. J; DAVIS, L. **Training feedforward neural networks using genetic algorithms.** In: IJCAI, volumen 89, p. 762–767, 1989.

[65] BRAUN, H; WEISBROD, J. **Evolving neural feedforward networks**. In: ARTIFICIAL NEURAL NETS AND GENETIC ALGORITHMS, p. 25–32. Springer, 1993.

[66] SCHAFFER, J. D. **Multiple objective optimization with vector evaluated genetic algorithms**. In: PROCEEDINGS OF THE 1ST INTERNATIONAL CONFERENCE ON GENETIC ALGORITHMS, p. 93–100. L. Erlbaum Associates Inc., 1985.

[67] SRINIVAS, N; DEB, K. **Muilti–objective optimization using non-dominated sorting in genetic algorithms**. Evolutionary computation, 2(3):221–248, 1994.

[68] DEB, K; AGRAWAL, S; PRATAP, A ; MEYARIVAN, T. **A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA–II**. In: INTERNATIONAL CONFERENCE ON PARALLEL PROBLEM SOLVING FROM NATURE, p. 849–858. Springer, 2000.

[69] ZANCHETTIN, C; LUDERMIR, T. B. **Sistemas neurais híbridos para reconhecimento de padrões em narizes artificiais**. Sba: Controle & Automação Sociedade Brasileira de Automatica, 16(2):159–172, 2005.

[70] MONTAVON, G; MÜLLER, K.-R. **Better representations: Invariant, disentangled and reusable**. In: NEURAL NETWORKS: TRICKS OF THE TRADE, p. 559–560. Springer, 2012.

[71] DURILLO, J. J; ZHANG, Y; ALBA, E ; NEBRO, A. J. **A study of the multi-objective next release problem**. In: PROCEEDINGS OF THE 1ST INTERNATIONAL SYMPOSIUM ON SEARCH BASED SOFTWARE ENGINEERING (SSBSE'09), p. 49–58, 2009.

[72] ZELENY, M; COCHRANE, J. L. **Multiple criteria decision making**. University of South Carolina Press, 1973.

[73] MATLAB. **version 8.2.0.701 (R2013b)**. The MathWorks Inc., Natick, Massachusetts, 2013.

[74] SÁNCHEZ RAMOS, L; ALCALÁ FERNÁNDEZ, J; FERNÁNDEZ HILARIO, A; LUENGO MARTÍN, J; DERRAC RUS, J; GARCÍA LÓPEZ, S ; HERRERA TRIGUERO, F. **Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework**. Journal of Multiple-Valued Logic and Soft Computing, 2011.

[75] FORECASTPRO. **version 3.000D Extended Edition**. Business Forecast Systems Inc., Belmont, Massachusetts, 2013.

[76] MAKRIDAKIS, S; HIBON, M. **The m3-competition: results, conclusions and implications**. International Journal of Forecasting, 16(4):451–476, 2000.

[77] HOME, N. **Artificial neural networks & computational intelligence forecasting competition**, 2007.

[78] KACHIGAN, S. K. **Statistical analysis: An interdisciplinary introduction to univariate & multivariate methods**. Radius Press, 1986.

[79] FLORES, B. E. **Use of the sign test to supplement the percentage better statistic**. International Journal of Forecasting, 2(4):477–489, 1986.

[80] FLORES, B. E. **The utilization of the wilcoxon test to compare forecasting methods: A note**. International Journal of Forecasting, 5(4):529–535, 1989.

[81] GIBBONS, J. D. **Nonparametric statistics: An introduction**. Número 90. Sage, 1993.

[82] DERRAC, J; GARCÍA, S; MOLINA, D ; HERRERA, F. **A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms**. Swarm and Evolutionary Computation, 1(1):3–18, 2011.

[83] CROMWELL, J. B. **Multivariate tests for time series models**. Número 100. Sage, 1994.

[84] BARBOSA, A. M; DE CARVALHO RIBEIRO, L ; DE OLIVEIRA ARANTES, J. M. **Algoritmo genético multiobjetivo: sistema adaptativo com elitismo**. 9th Brazilian Conference on Dynamics, Control and their Applications, 2:16, 2010.

[85] ANP. **Agencia nacional do petróleo, gás natural e biocombustíveis**, 2011.

[86] ASSIMAKOPOULOS, V; NIKOLOPOULOS, K. **The theta model: a decomposition approach to forecasting**. International journal of forecasting, 16(4):521–530, 2000.

[87] YAN, W. **Toward automatic time-series forecasting using neural networks**. IEEE Transactions on Neural Networks and Learning Systems, 23(7):1028–1039, 2012.

[88] SPECHT, D. F. **A general regression neural network**. IEEE transactions on neural networks, 2(6):568–576, 1991.

[89] FILDES, R; HIBON, M; MAKRIDAKIS, S ; MEADE, N. **Generalising about univariate forecasting methods: further empirical evidence**. International Journal of Forecasting, 14(3):339–358, 1998.

[90] CHATFIELD, C; YAR, M. **Holt-Winters forecasting: some practical issues**. The Statistician, p. 129–140, 1988.

[91] MAKRIDAKIS, S; WHEELWRIGHT, S. C ; HYNDMAN, R. J. **Forecasting: methods and applications. 1998**.

[92] BOX, G. E; JENKINS, G. M; REINSEL, G. C ; LJUNG, G. M. **Time series analysis: forecasting and control**. John Wiley & Sons, 2015.

[93] HARVEY, A. C. **Forecasting, structural time series models and the Kalman filter**. Cambridge university press, 1990.

[94] SOUZA, R. C; CAMARGO, M. E. **Análise e previsão de séries temporais: os modelos ARIMA**. Ijuí: Sedigraf, 1996.

A
Metodologias de Previsão

A.1 Holt-Winters multiplicativo

O Holt-Winters multiplicativo (HW) é um método adaptativo, largamente utilizado em previsão de demanda [6]. Considerando-se período sazonal de comprimento s e horizonte de previsão h, a equação de previsão do HW é descrita da seguinte maneira:

$$\hat{y}_{t+h|t} = (L_t + b_t h) I_{t-s-h}\,, \tag{A-1}$$

$$\text{onde } L_t = \alpha \frac{y_t}{I_{t-s}} + (1-\alpha)(L_{t-1} + b_{t-1})\,, \tag{A-2}$$

$$b_t = \beta(L_t - L_{t-1}) + (1-\beta) b_{t-1}\,, \tag{A-3}$$

$$I_t = \gamma \frac{y_t}{L_t} + (1-\gamma) I_{t-s}\,. \tag{A-4}$$

Em A-2–A-4 representam-se, respectivamente, os componentes da equação de previsão estimados em t: (i) nível, (ii) tendência e (ii) índices sazonais. A constantes α, β e γ são hiperparâmetros escolhidos na fase de projeto do modelado [89]. Para a previsão múltiplos passos a frente, deve haver s índices sazonais calculados pela análise histórica da série, realizando o que se chama de correção sazonal. Em geral, os índices sazonais são normalizados (transformados) de maneira que a sua soma se iguale a s.

Há diversas variações na implementação do método HW [90]. As principais diferenças ocorrem principalmente na iniciação dos componentes recursivos e na frequência de normalização dos índices sazonais (a todo instante, a cada período completo ou apenas no final do ajuste histórico). O pacote Forecast Pro [75] por exemplo,[1] inicia valores com uma técnica inspirada no *backacasting* [91,92] e só normaliza índices sazonais ao final do ajuste histórico. Considerando-se sazonalidade mensal, i.e., $s = 12$, há exatamente 12 índices para correção sazonal.

[1]Segundo *e-mail* enviado por seus autores.

A.2 ARIMA Box & Jenkins

A metodologia ARIMA Box & Jenkins (BJ) encontra sua base na teoria dos processo estocásticos [92–94].

Usando notação dos modelos de regressão, a equação de previsão dos modelos BJ assume a forma da equação

$$\begin{aligned}\hat{y}_{t+h|t} = \alpha_1\hat{y}_{t+h-1|t} + \alpha_2\hat{y}_{t+h-2|t} + \ldots + \alpha_m\hat{y}_{t+h-m|t} + \\ \beta_1\varepsilon_{t+h-1|t} + \beta_2\varepsilon_{t+h-2|t} + \ldots + \beta_n\varepsilon_{t+h-n|t}\,,\end{aligned} \tag{A-5}$$

$$\text{onde } \varepsilon_{t+h|t} = (y_{t+h|t} - \hat{y}_{t+h|t})\,, \tag{A-6}$$

$$\alpha_i = f^{\,n\tilde{a}o\ linear}\left(\phi_1, \phi_2, \ldots, \phi_p, \theta_1, \theta_2, \ldots, \theta_q, \Phi_1, \Phi_2, \ldots, \Phi_P, \Theta_1, \Theta_2, \ldots, \Theta_Q\right)\,, \tag{A-7}$$

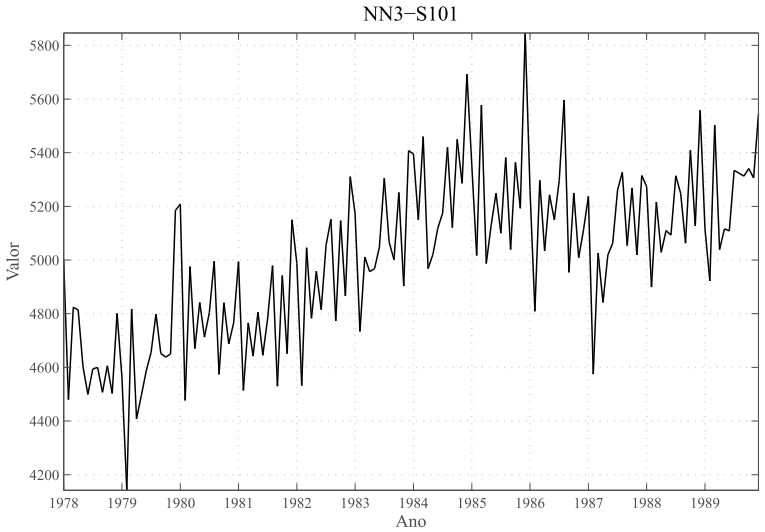
$$\beta_i = f^{\,n\tilde{a}o\ linear}\left(\phi_1, \phi_2, \ldots, \phi_p, \theta_1, \theta_2, \ldots, \theta_q, \Phi_1, \Phi_2, \ldots, \Phi_P, \Theta_1, \Theta_2, \ldots, \Theta_Q\right)\,. \tag{A-8}$$

O fato dos parâmetros $\alpha_1, \alpha_2 \ldots, \alpha_m, \beta_1, \beta_2, \ldots, \beta_n$ dependerem de maneira não linear de outros $p + q + P + Q$ parâmetros[2] torna inviável a sua estimação com métodos iterativos. Ainda, o processo de otimização (não linear) dos parâmetros BJ é restrito: deve-se cuidar para que a equação do modelo especificado atenda às caraterísticas de **inversibilidade** e **estacionalidade** [92–94].

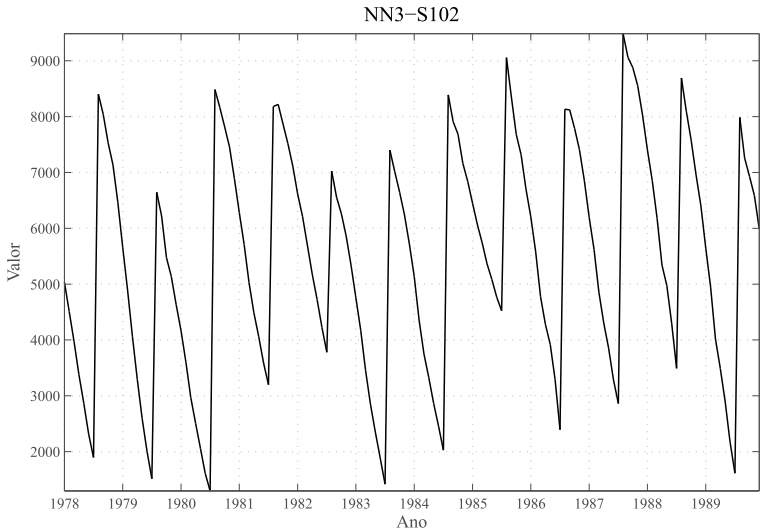
[2] A forma desta dependência varia de acordo com o modelo ARIMA escpecificado.

B
Séries da Competição NN3

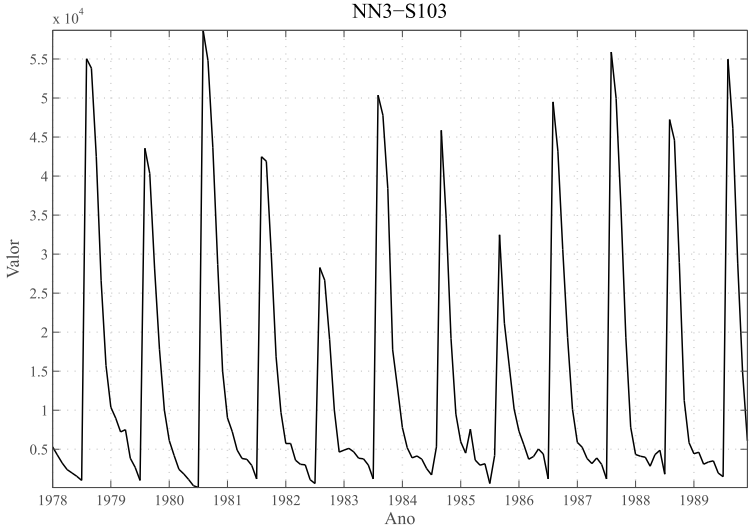
As Figuras B.1–B.3 exibem as 11 séries correspondentes ao conjunto reduzido da Competição NN3 [77], utilizadas como estudo de casos no presente trabalho.



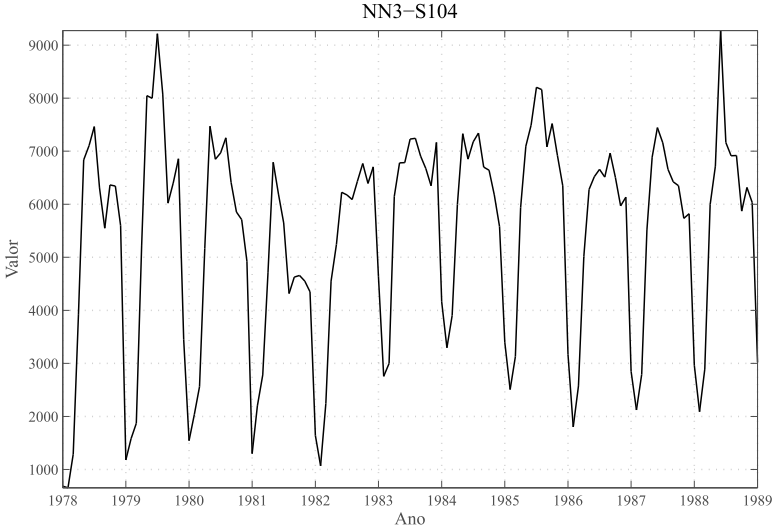
B.1(a): Série NN3-101



B.1(b): Série NN3-102

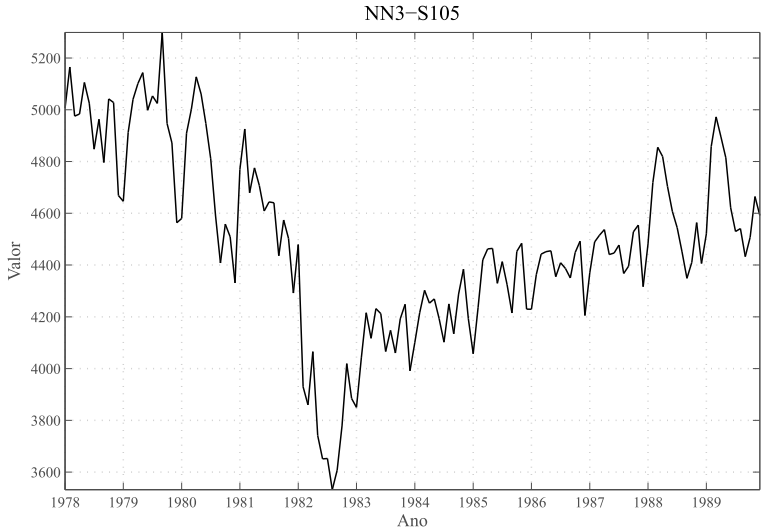


B.1(c): Série NN3-103

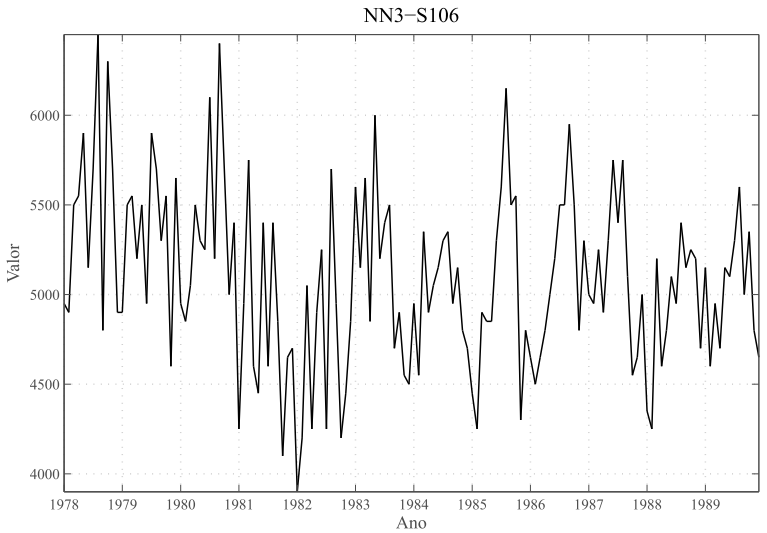


B.1(d): Série NN3-104

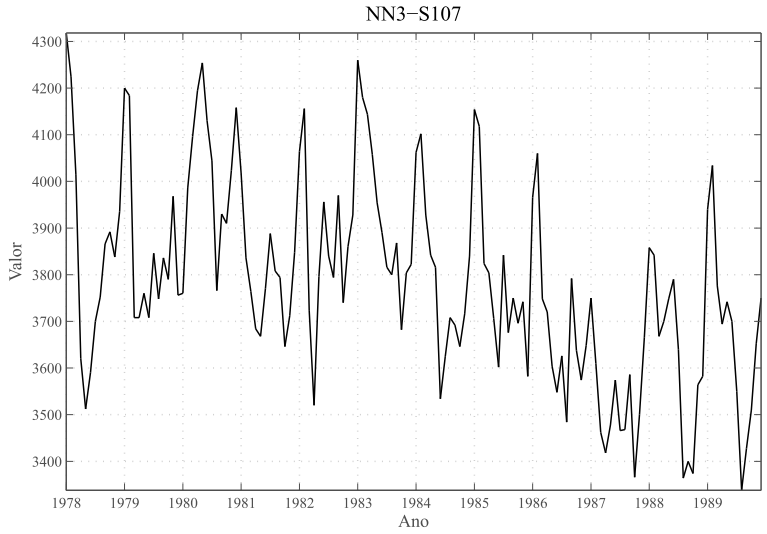
Figura B.1: Séries de Competição NN3 - Série NN3-101 a NN3-104.



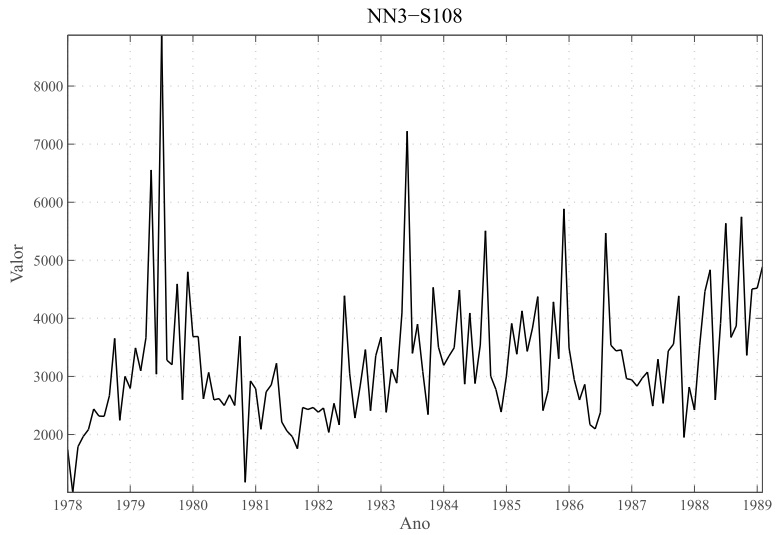
B.2(a): Série NN3-105



B.2(b): Série NN3-106

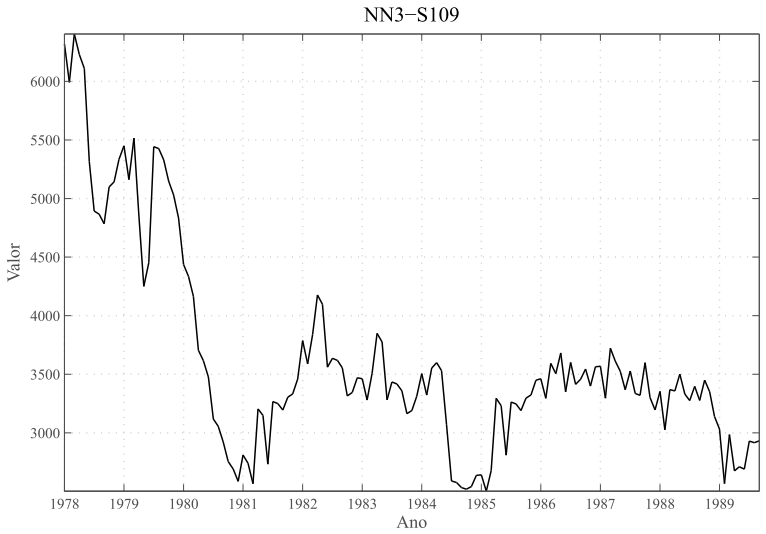


B.2(c): Série NN3-107

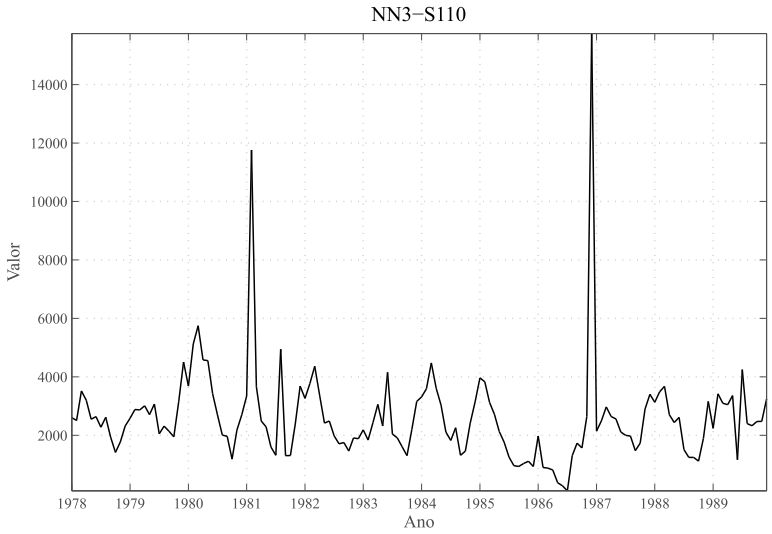


B.2(d): Série NN3-108

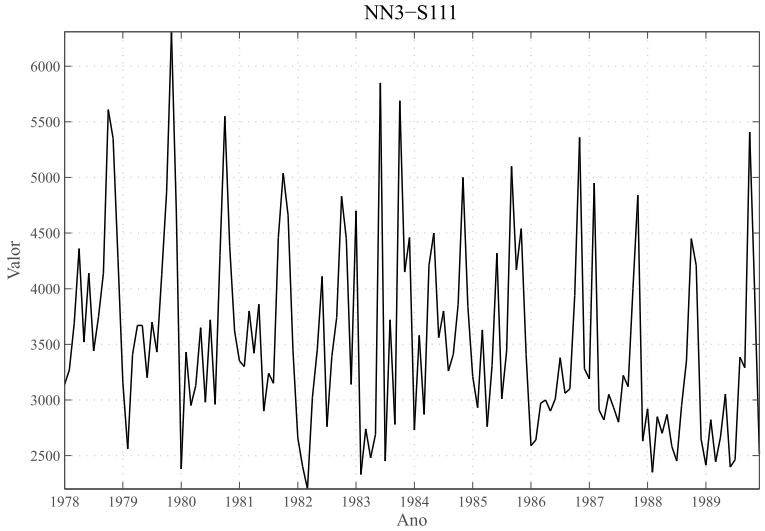
Figura B.2: Séries de Competição NN3 - Série NN3-105 a NN3-108.



B.3(a): Série NN3-109



B.3(b): Série NN3-110

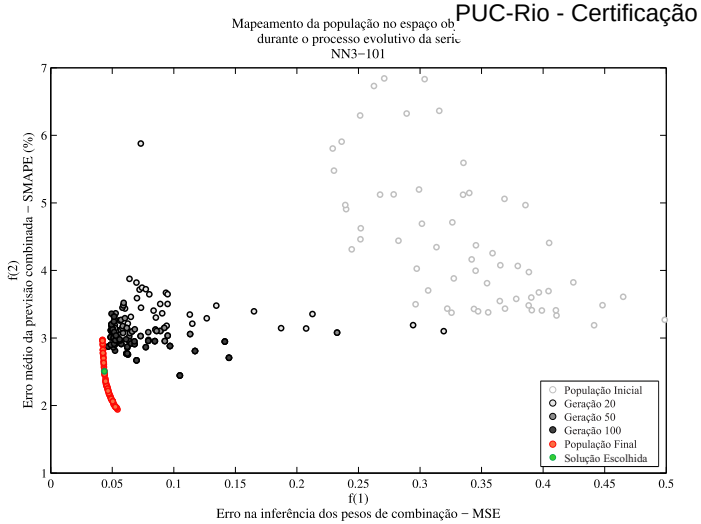


B.3(c): Série NN3-111

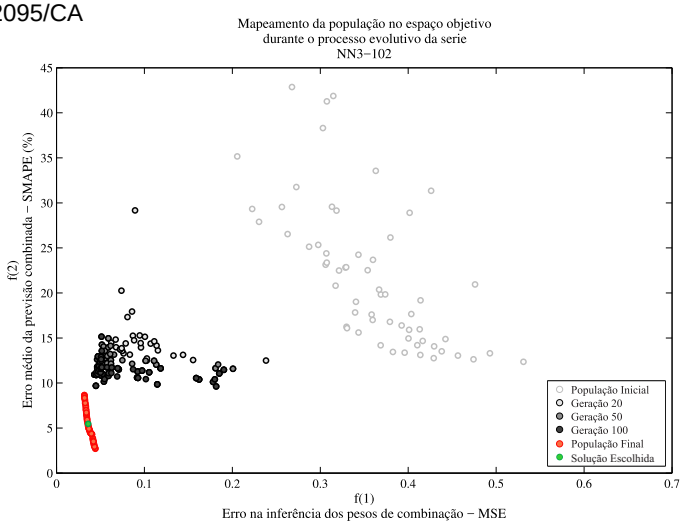
Figura B.3: Séries de Competição NN3 - Série NN3-109 a NN3-111.

C
Progresso Evolutivo nas Séries da Competição NN3

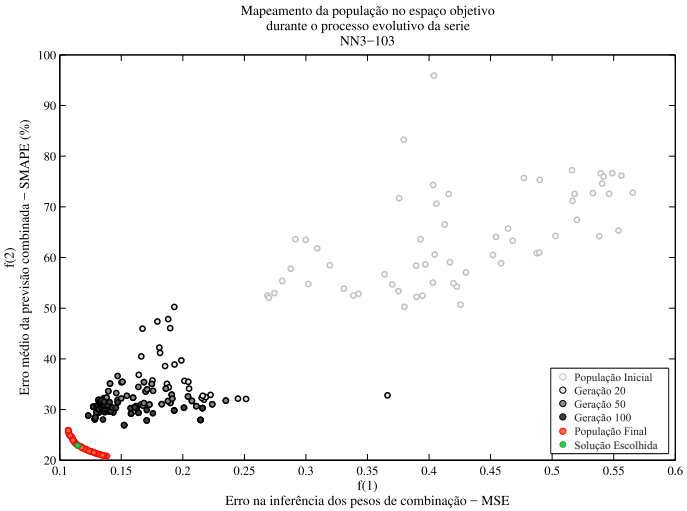
As Figuras C.1–C.3 apresentam, para cada série avaliada no segundo estudo de casos o progresso do algoritmo durante diferentes etapas do processo evolutivo. Pode-se observar em cada figura, a fronteira de Pareto ótima (em vermelho) encontrada pelo algoritmo ao final do processo evolutivo.



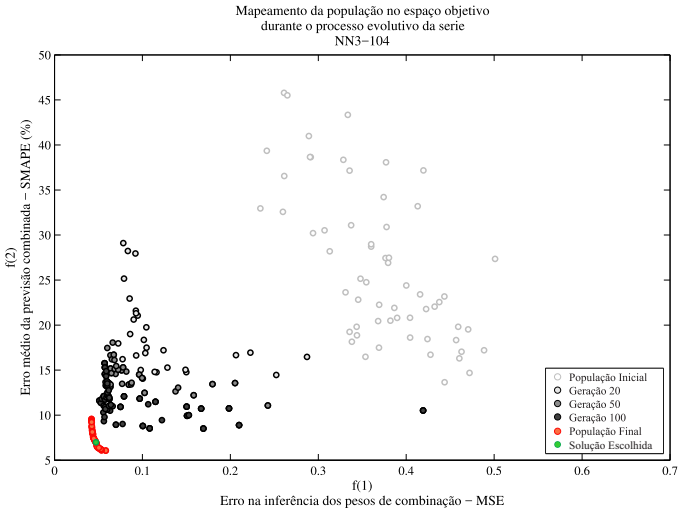
C.1(a): Progresso evolutivo na série NN3-101



C.1(b): Progresso evolutivo na série NN3-102

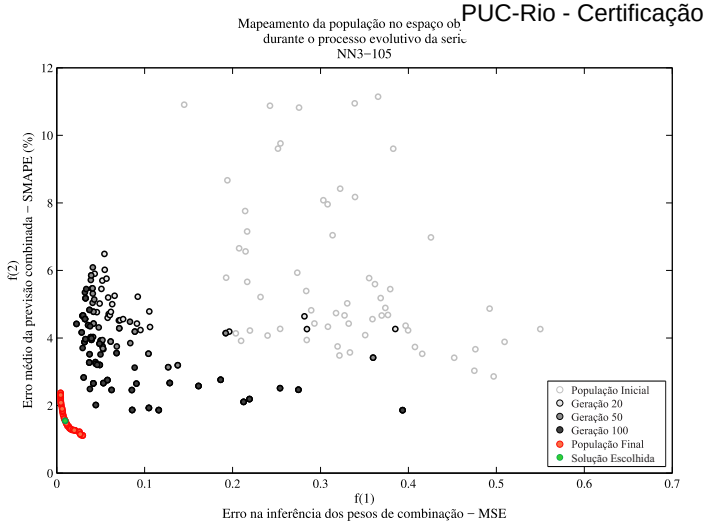


C.1(c): Progresso evolutivo na série NN3-103

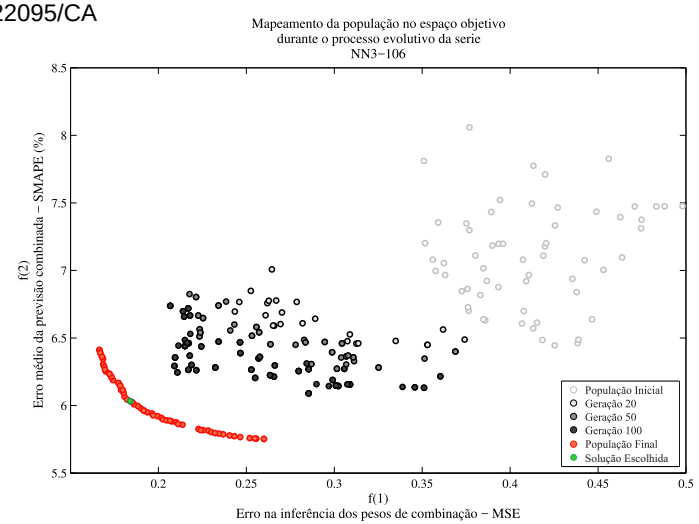


C.1(d): Progresso evolutivo na série NN3-104

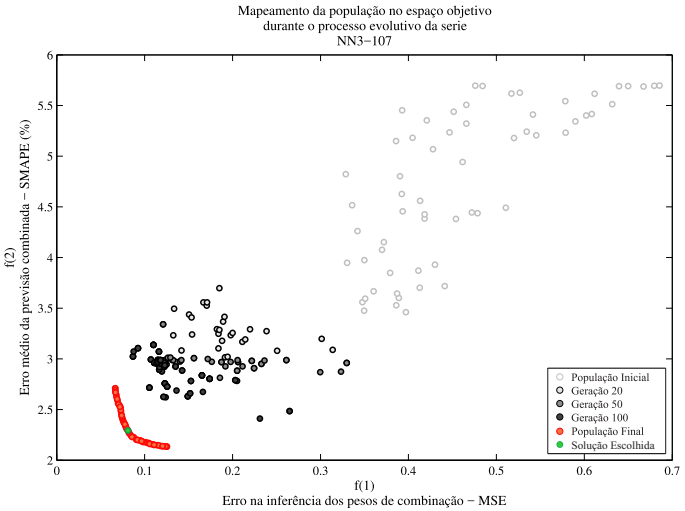
Figura C.1: Mapeamento da população sobre o espaço de objetivos durante diferentes etapas do processo evolutivo - Série NN3-101 a NN3-104.



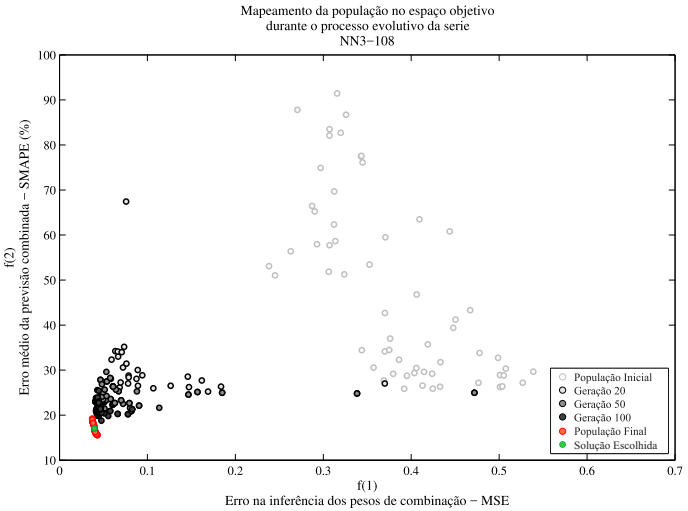
C.2(a): Progresso evolutivo na série NN3-105



C.2(b): Progresso evolutivo na série NN3-106

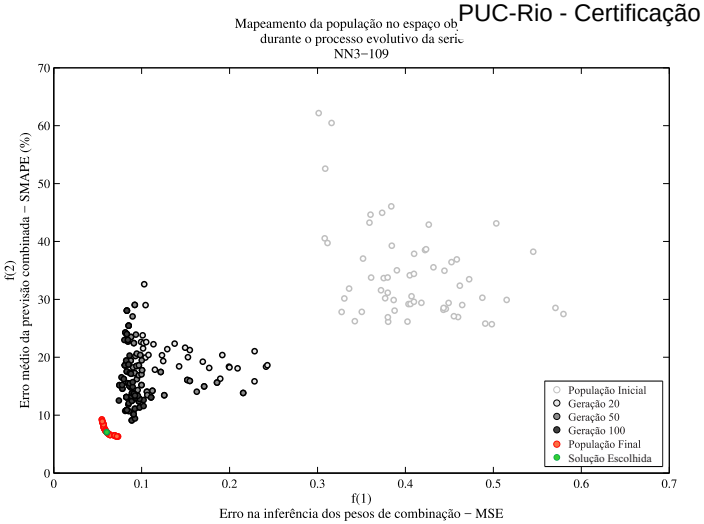


C.2(c): Progresso evolutivo na série NN3-107

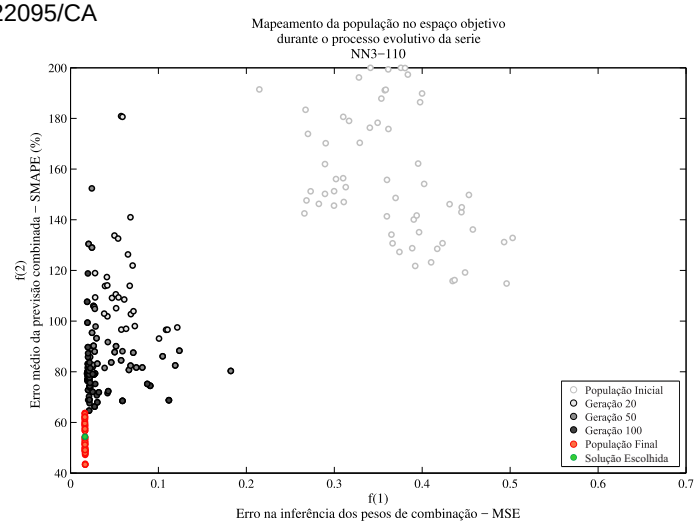


C.2(d): Progresso evolutivo na série NN3-108

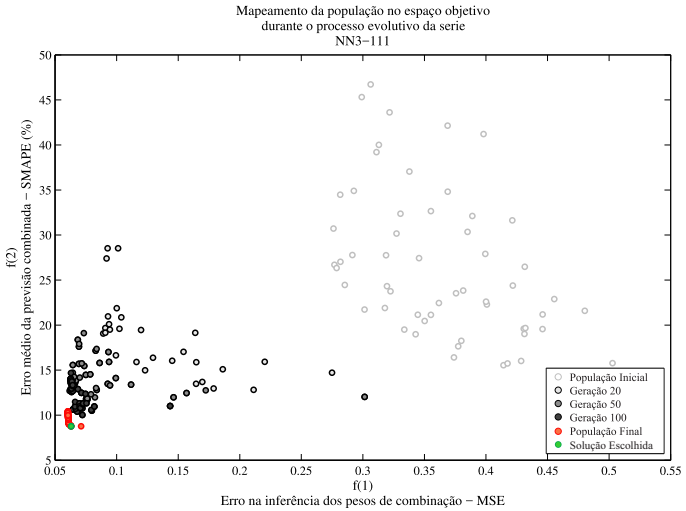
Figura C.2: Mapeamento da população sobre o espaço de objetivos durante diferentes etapas do processo evolutivo - Série NN3-105 a NN3-108.



C.3(a): Progresso evolutivo na série NN3-109



C.3(b): Progresso evolutivo na série NN3-110

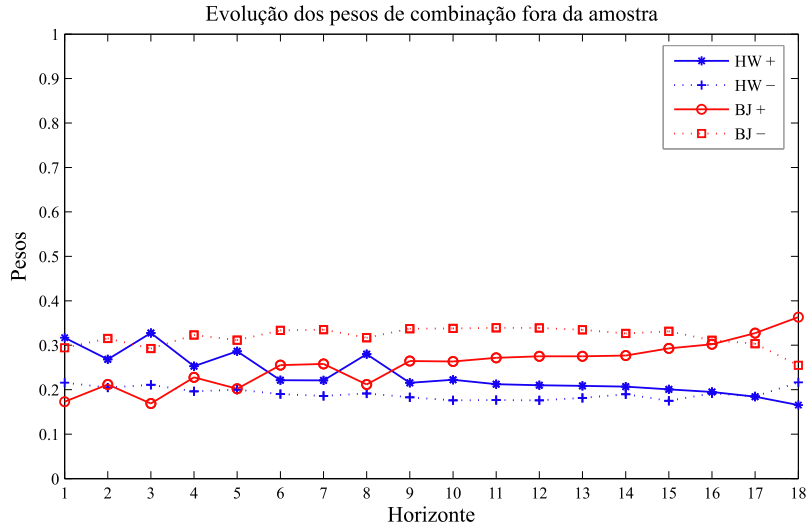


C.3(c): Progresso evolutivo na série NN3-111

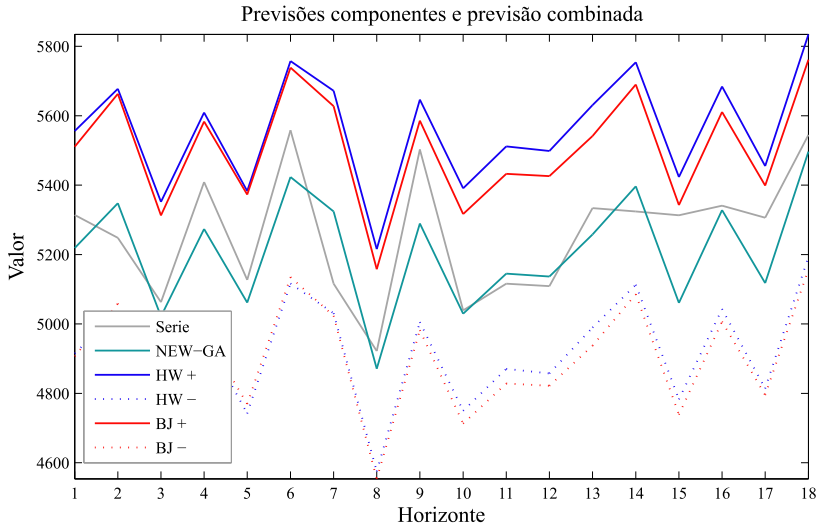
Figura C.3: Mapeamento da população sobre o espaço de objetivos durante diferentes etapas do processo evolutivo - Série NN3-109 a NN3-111.

D
Resultados Individuas nas Séries da Competição NN3

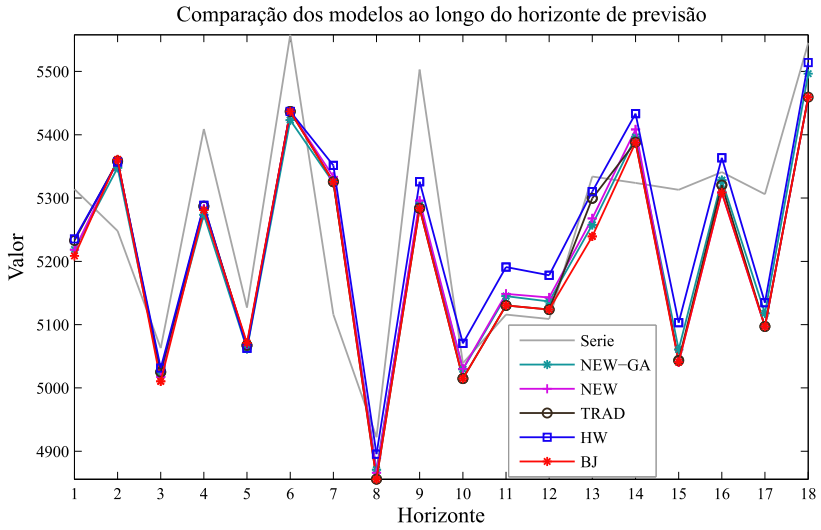
Nas Figuras D.1–D.11 são apresentados os resultados individuais para cada série da competição NN3. Exibem-se a evolução dos pesos de combinação ao longo do horizonte de previsão; considerando a solução escolhida a partir a fronteira de Pareto ótima fornecida pelo algoritmo. Apresenta-se também as previsões componentes e a previsão combinada, gerada dinamicamente quando os vetores de previsão em cada ponto do horizonte são ponderados a partir dos correspondentes vetores de pesos estimados naquele ponto. Exibem-se igualmente, as previsões dos modelos (individuais/combinação), assim como a evolução dos seus respetivos SMAPEs ao longo do horizonte de previsão.



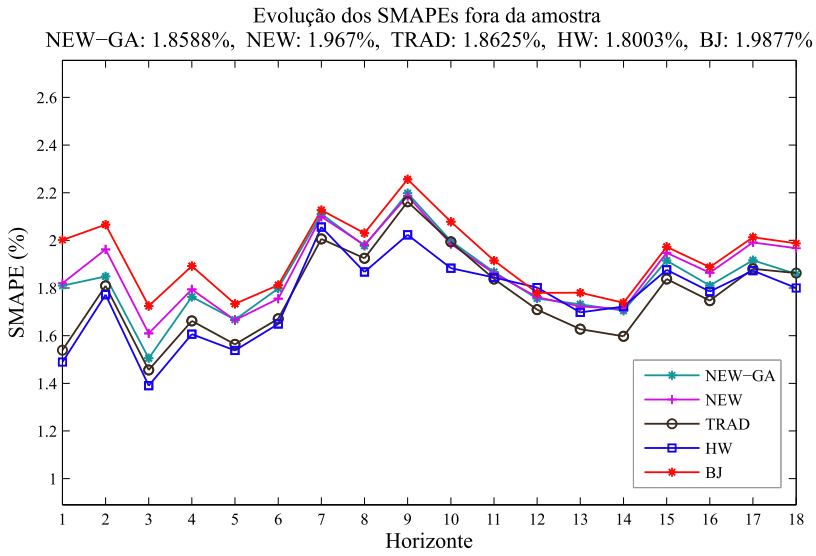
D.1(a): Evolução dos pesos de combinação



D.1(b): Previsões componentes e previsão combinada

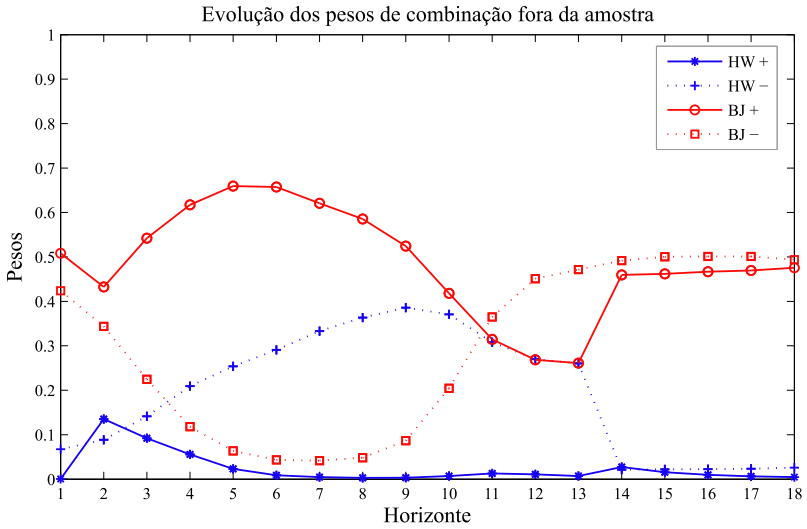


D.1(c): Previsões dos modelos de previsão avaliados

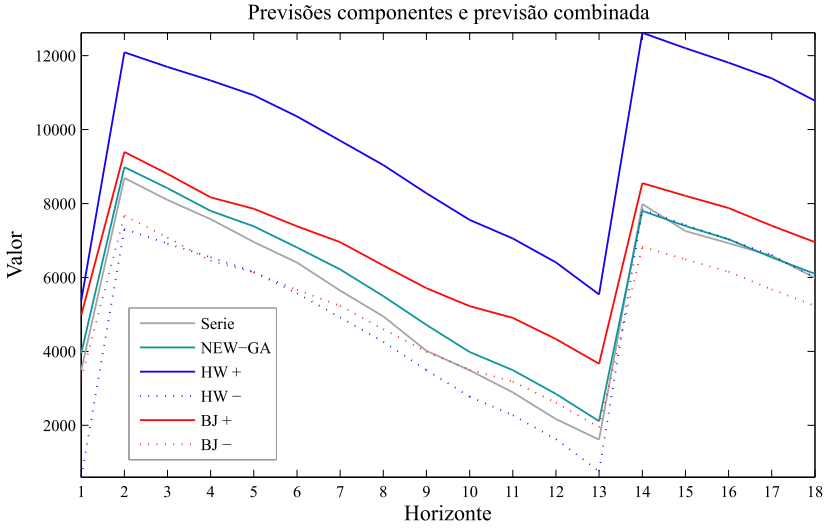


D.1(d): Evolução dos SMAPEs para cada modelo

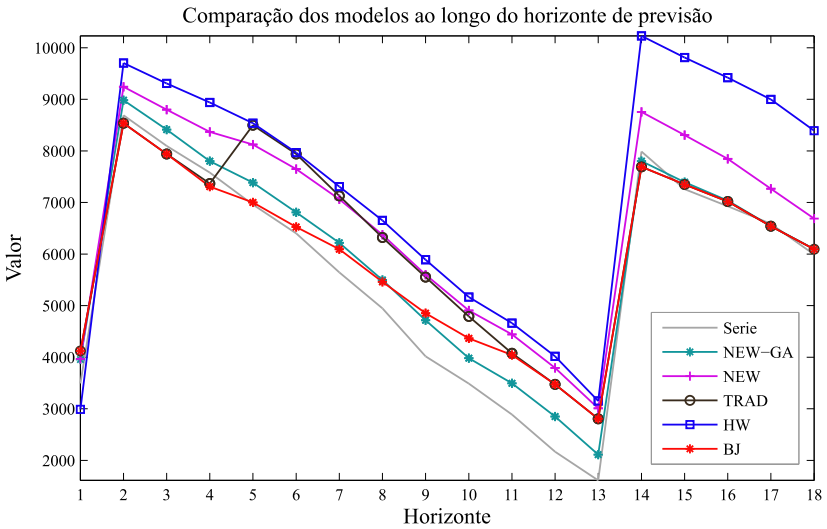
Figura D.1: Resumo dos resultados obtidos para a série NN3-101.



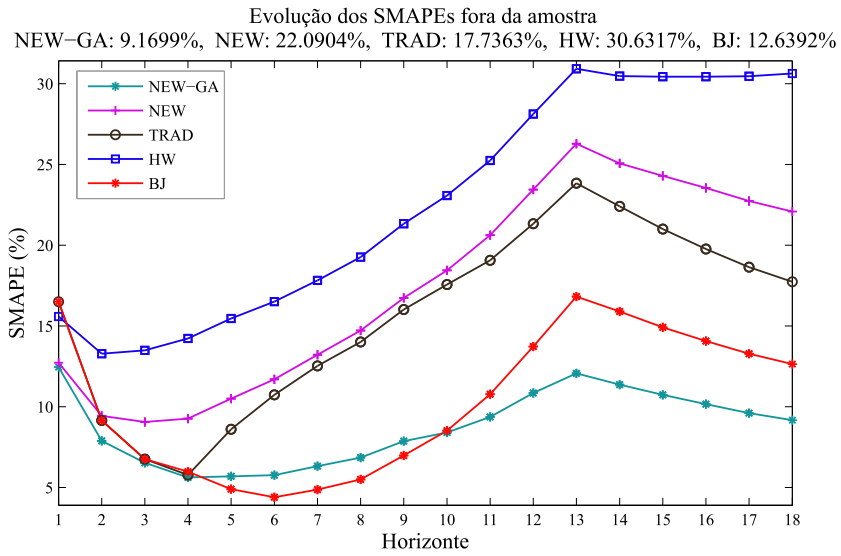
D.2(a): Evolução dos pesos de combinação



D.2(b): Previsões componentes e previsão combinada

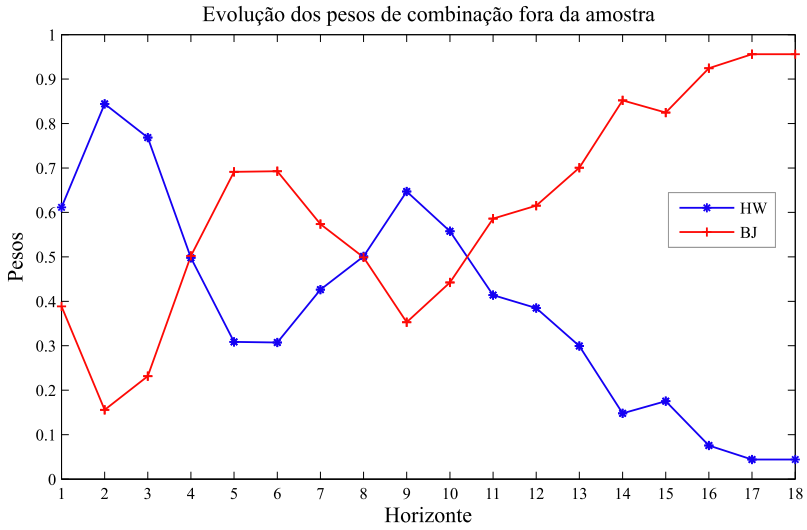


D.2(c): Previsões dos modelos de previsão avaliados

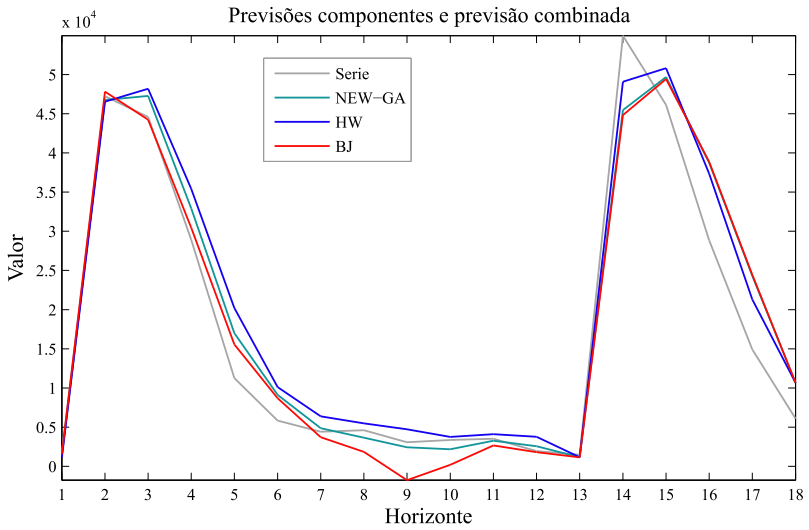


D.2(d): Evolução dos SMAPEs para cada modelo

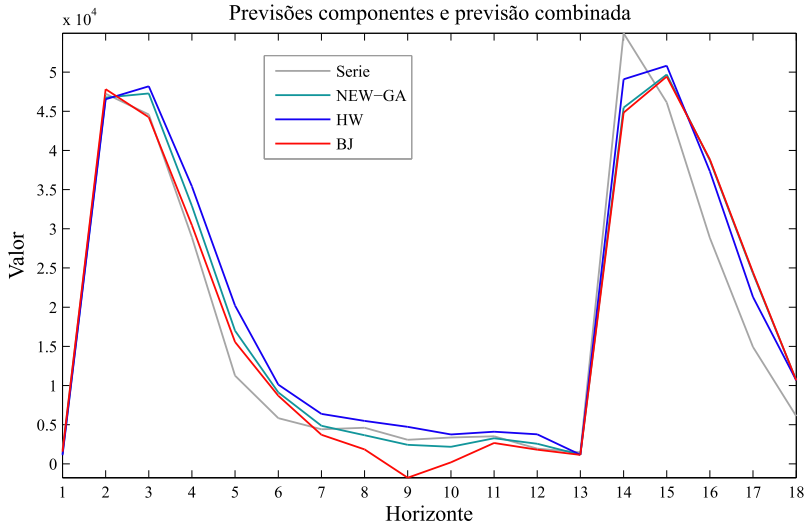
Figura D.2: Resumo dos resultados obtidos para a série NN3-102.



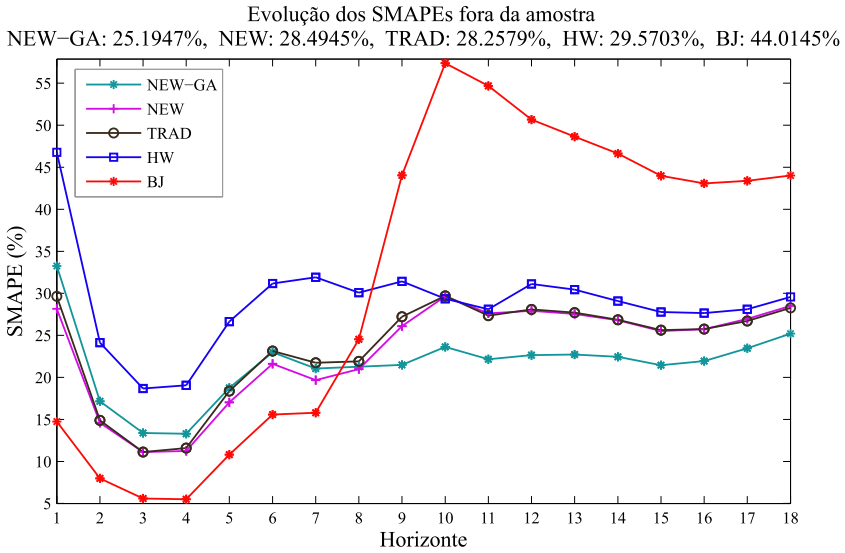
D.3(a): Evolução dos pesos de combinação



D.3(b): Previsões componentes e previsão combinada

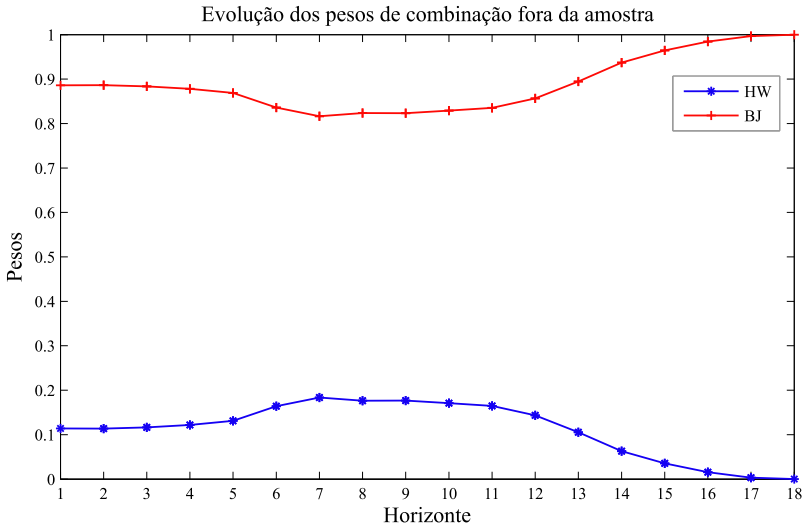


D.3(c): Previsões dos modelos de previsão avaliados

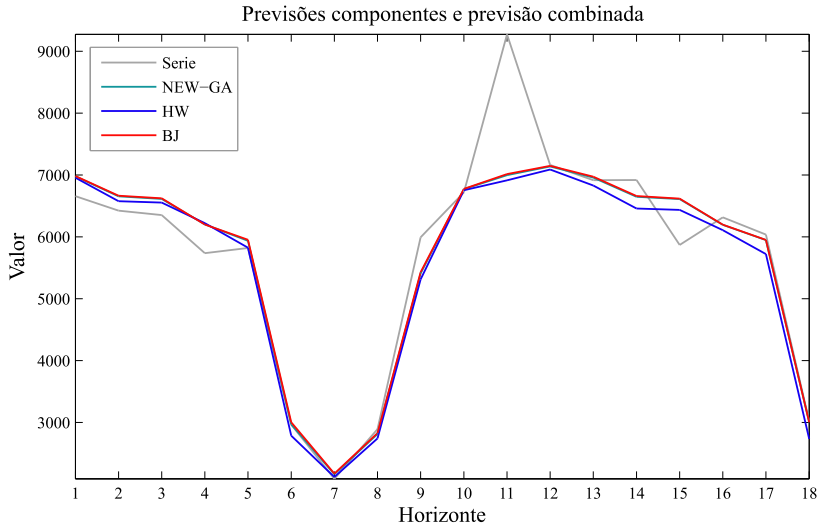


D.3(d): Evolução dos SMAPEs para cada modelo

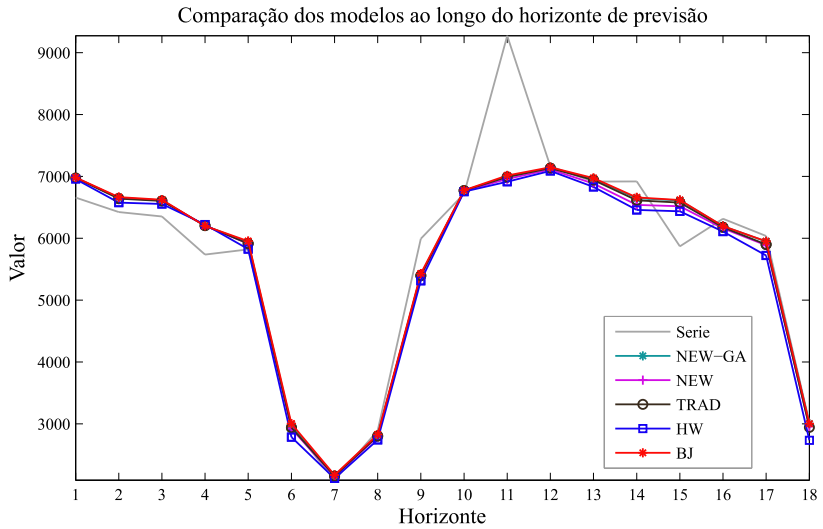
Figura D.3: Resumo dos resultados obtidos para a série NN3-103.



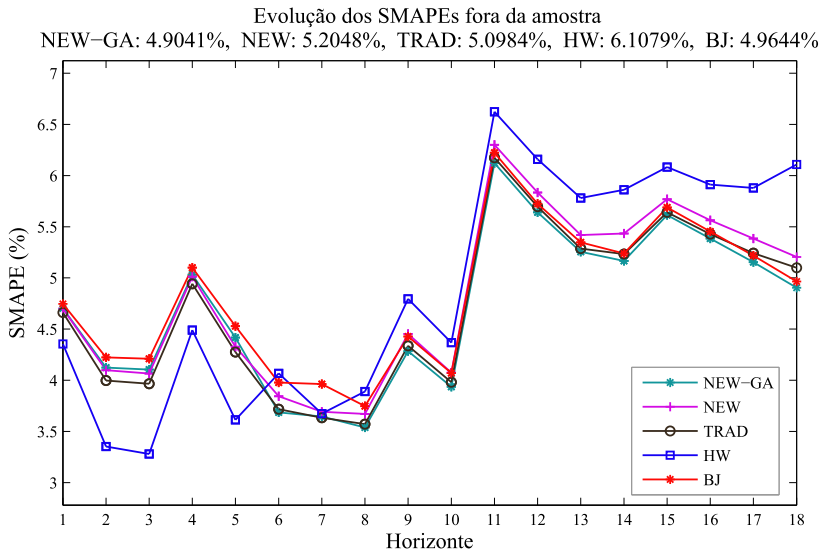
D.4(a): Evolução dos pesos de combinação



D.4(b): Previsões componentes e previsão combinada

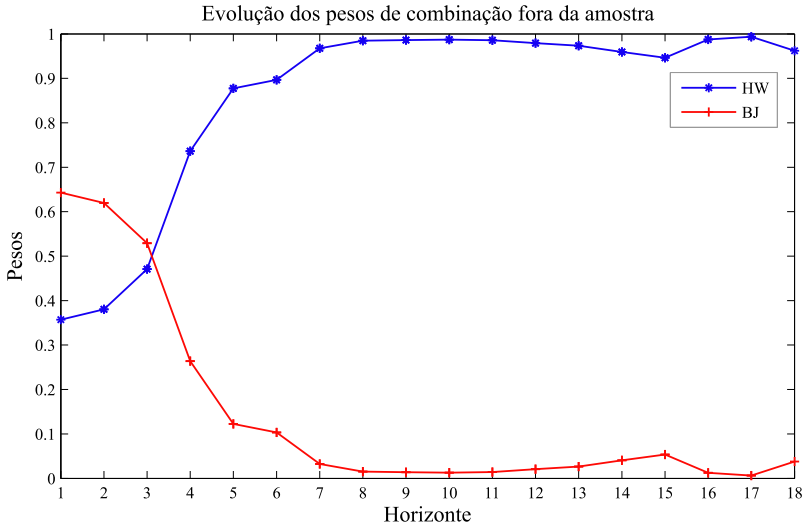


D.4(c): Previsões dos modelos de previsão avaliados

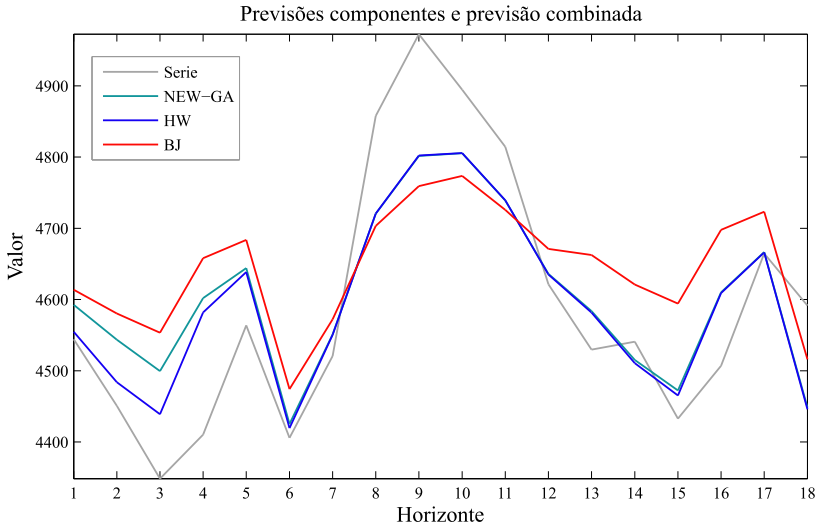


D.4(d): Evolução dos SMAPEs para cada modelo

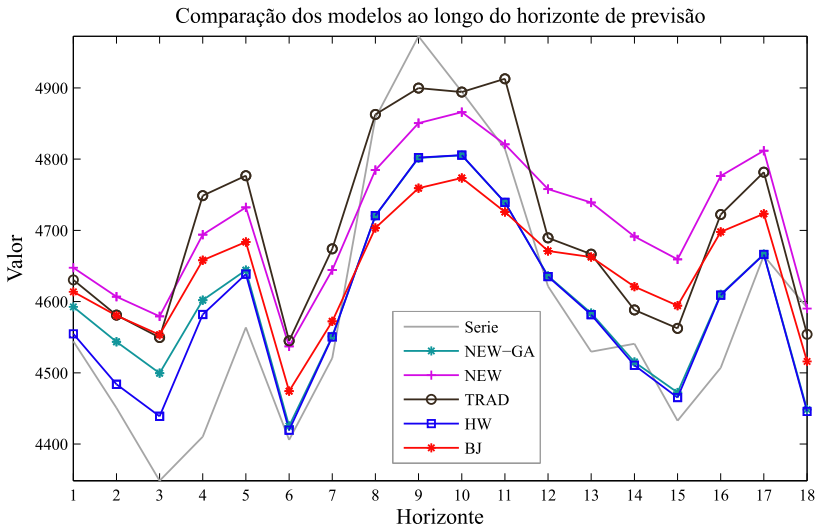
Figura D.4: Resumo dos resultados obtidos para a série NN3-104.



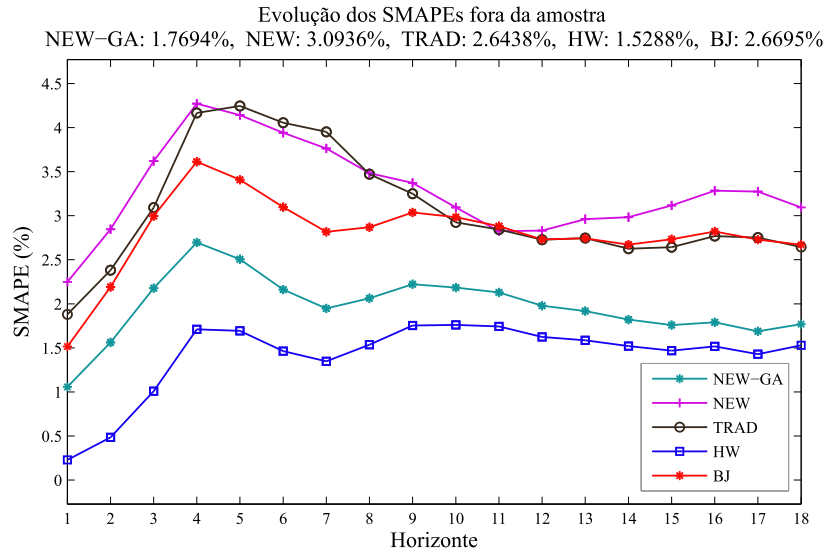
D.5(a): Evolução dos pesos de combinação



D.5(b): Previsões componentes e previsão combinada

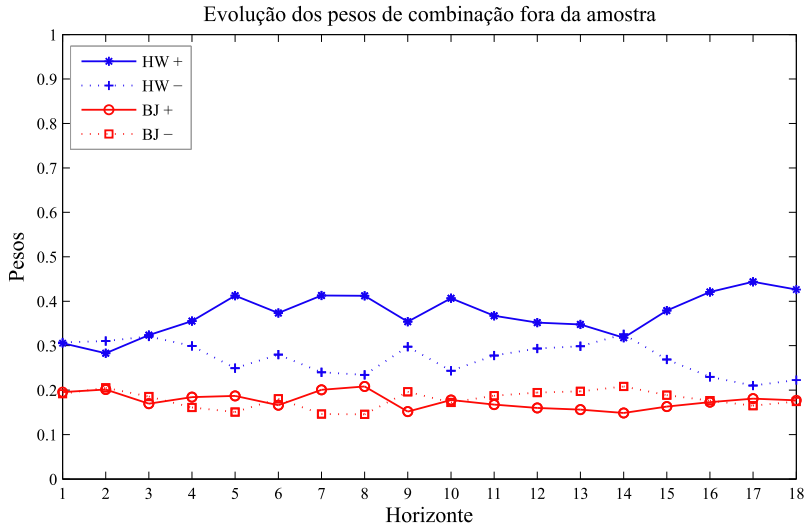


D.5(c): Previsões dos modelos de previsão avaliados

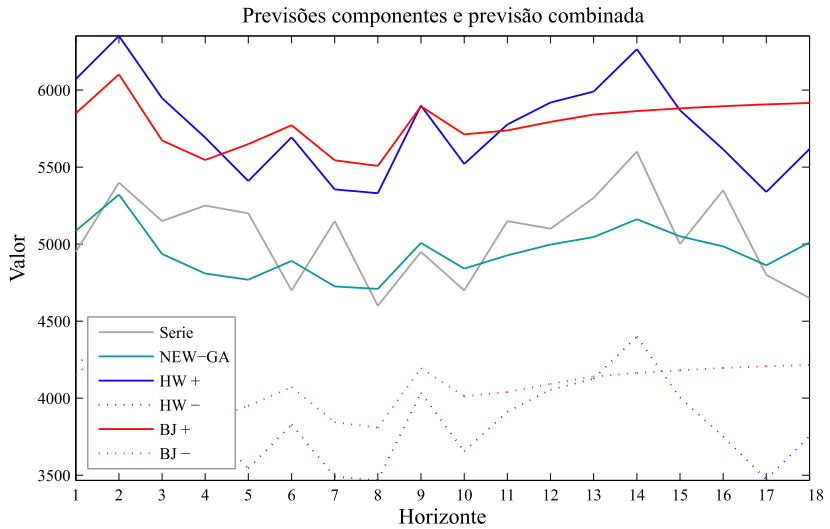


D.5(d): Evolução dos SMAPEs para cada modelo

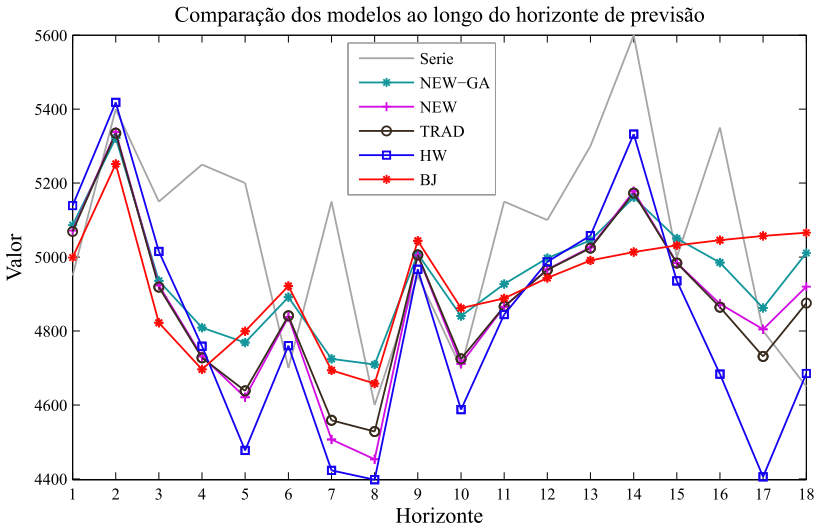
Figura D.5: Resumo dos resultados obtidos para a série NN3-105.



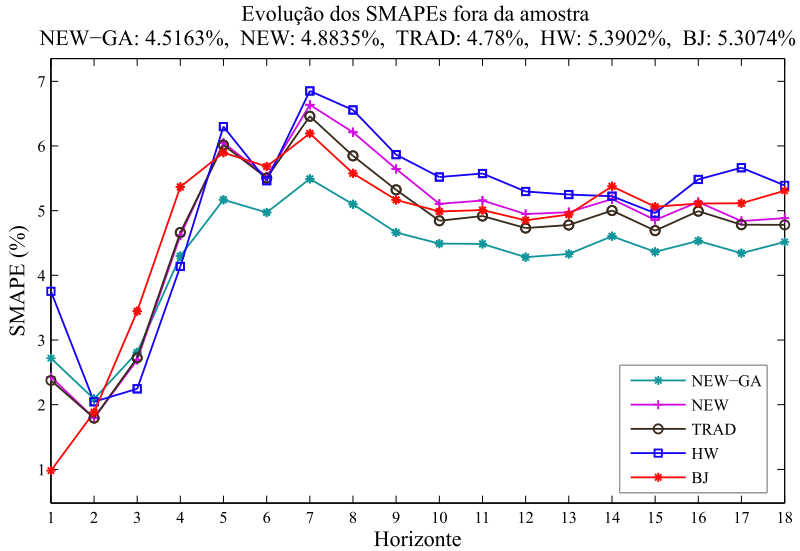
D.6(a): Evolução dos pesos de combinação



D.6(b): Previsões componentes e previsão combinada

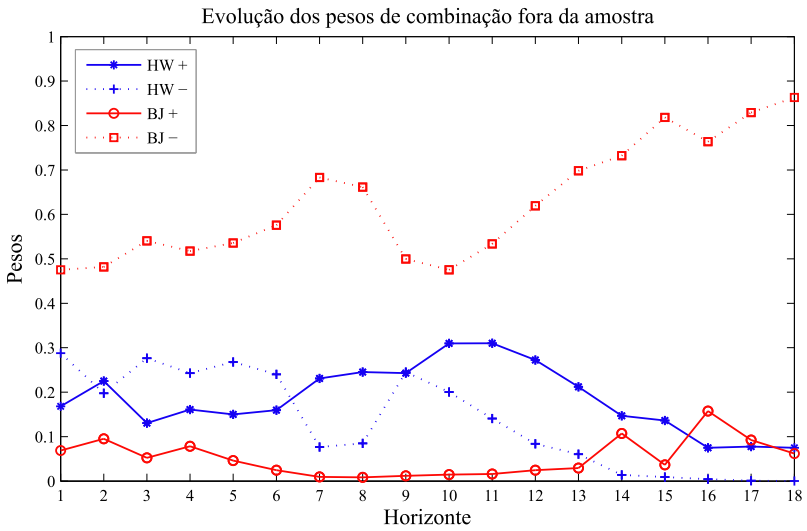


D.6(c): Previsões dos modelos de previsão avaliados

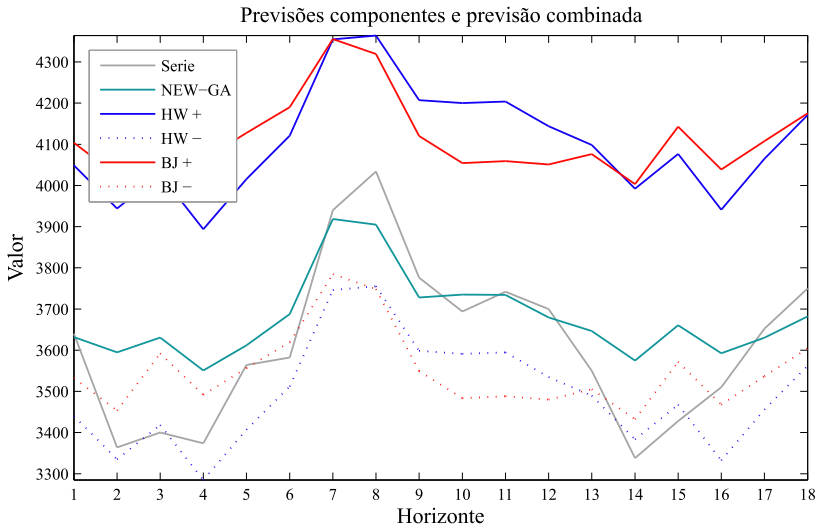


D.6(d): Evolução dos SMAPEs para cada modelo

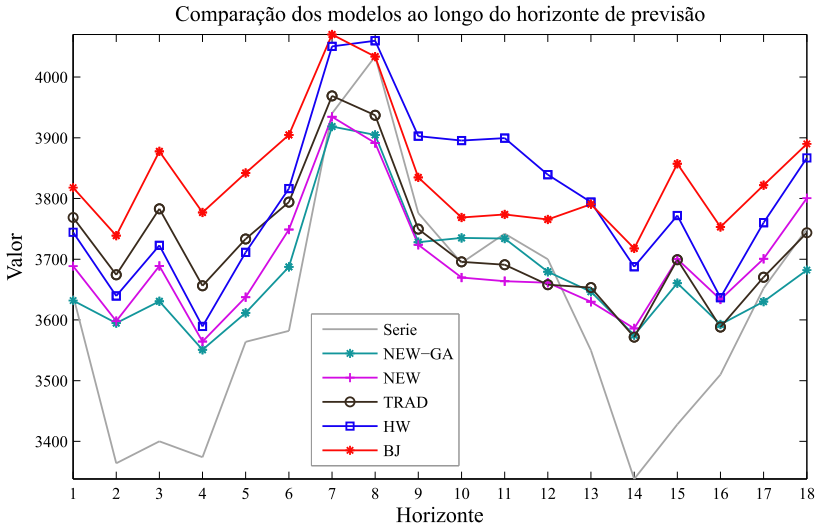
Figura D.6: Resumo dos resultados obtidos para a série NN3-106.



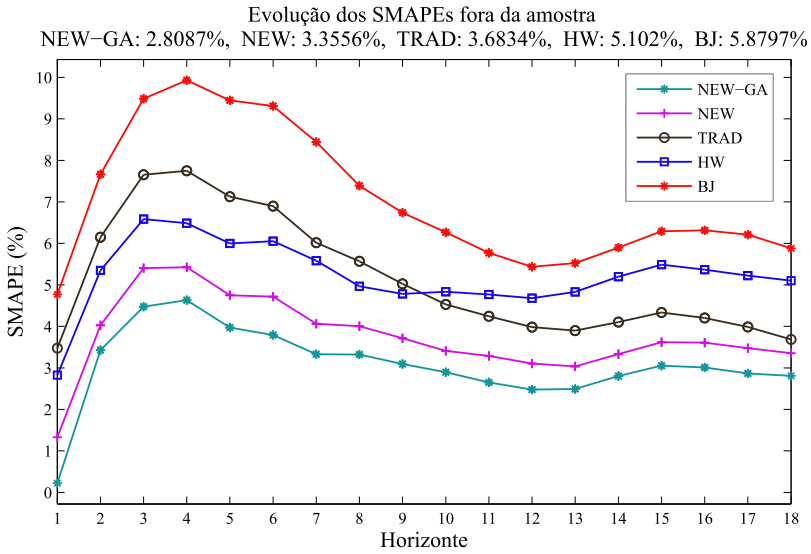
D.7(a): Evolução dos pesos de combinação



D.7(b): Previsões componentes e previsão combinada

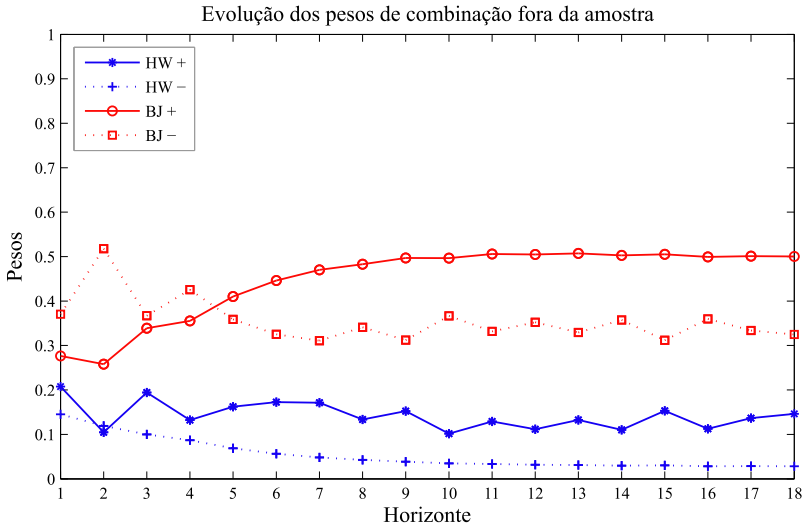


D.7(c): Previsões dos modelos de previsão avaliados

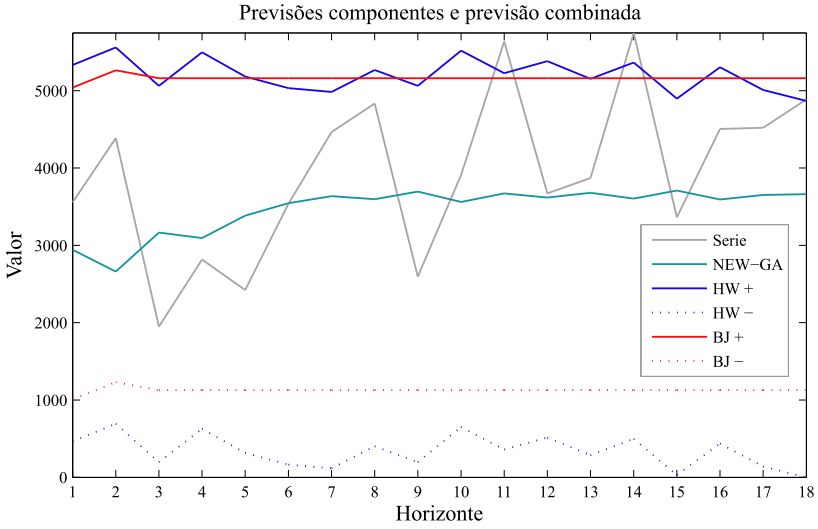


D.7(d): Evolução dos SMAPEs para cada modelo

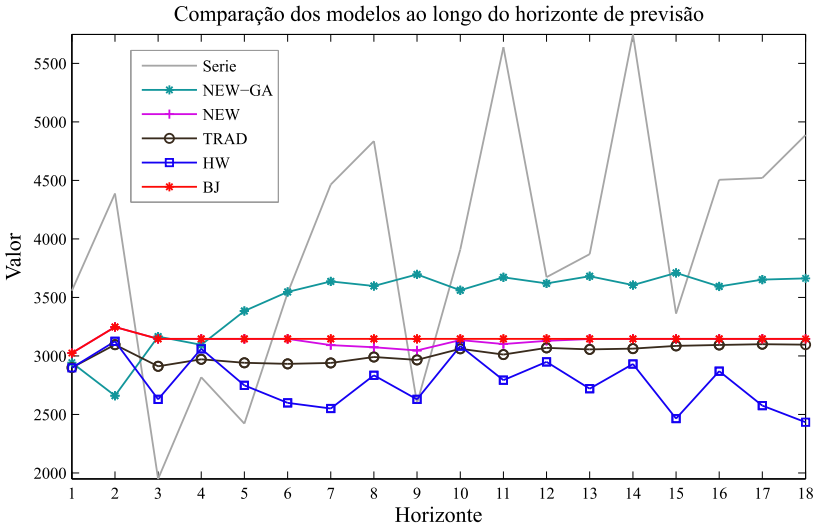
Figura D.7: Resumo dos resultados obtidos para a série NN3-107.



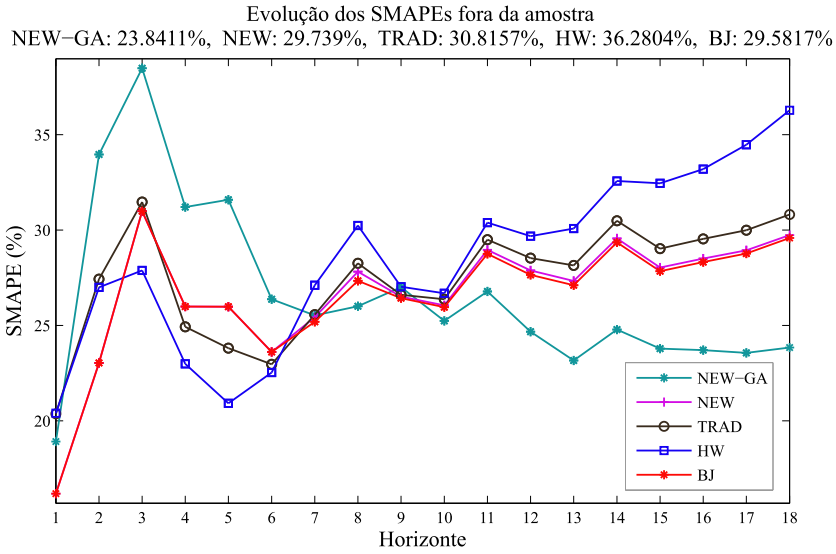
D.8(a): Evolução dos pesos de combinação



D.8(b): Previsões componentes e previsão combinada

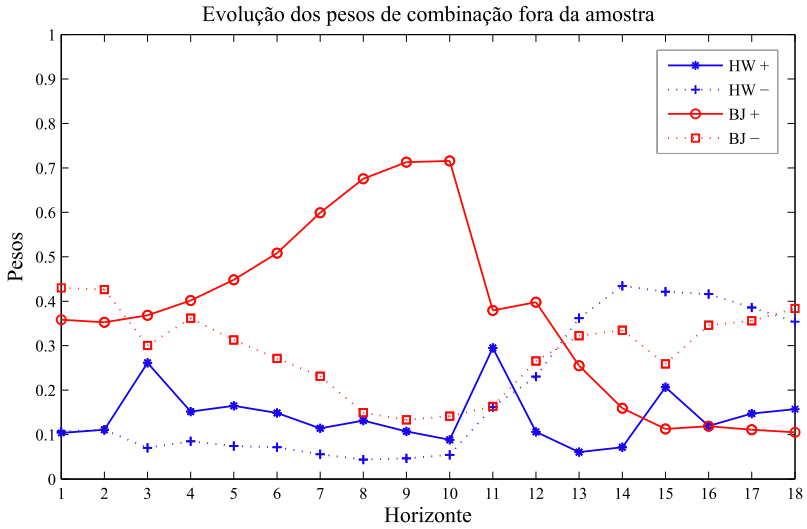


D.8(c): Previsões dos modelos de previsão avaliados

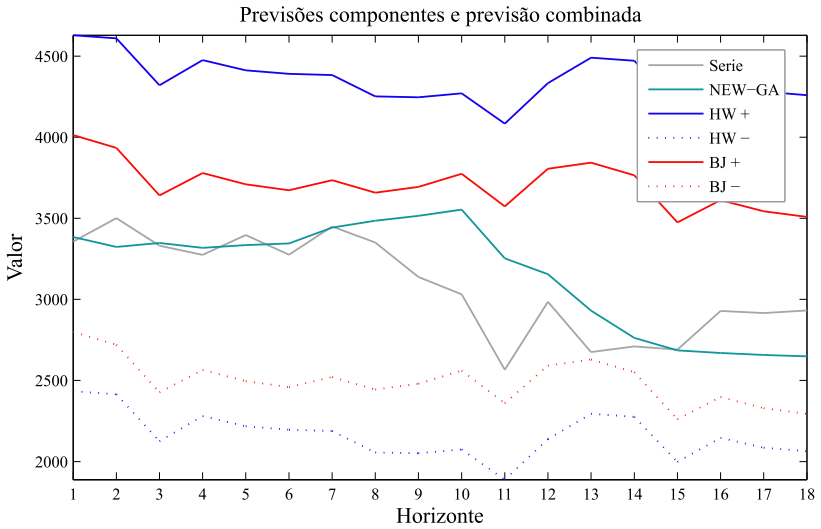


D.8(d): Evolução dos SMAPEs para cada modelo

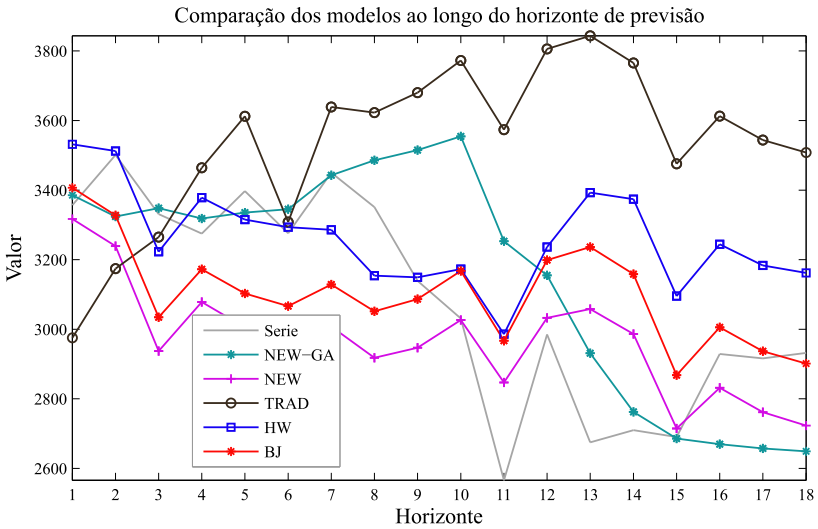
Figura D.8: Resumo dos resultados obtidos para a série NN3-108.



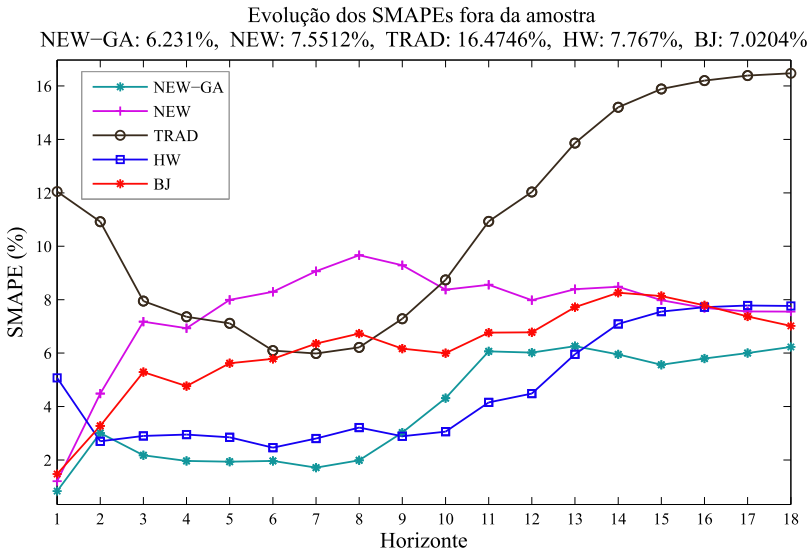
D.9(a): Evolução dos pesos de combinação



D.9(b): Previsões componentes e previsão combinada

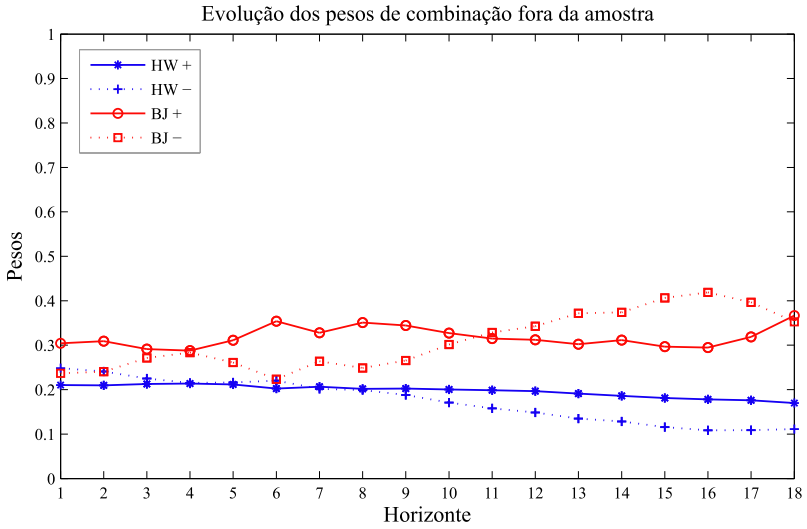


D.9(c): Previsões dos modelos de previsão avaliados

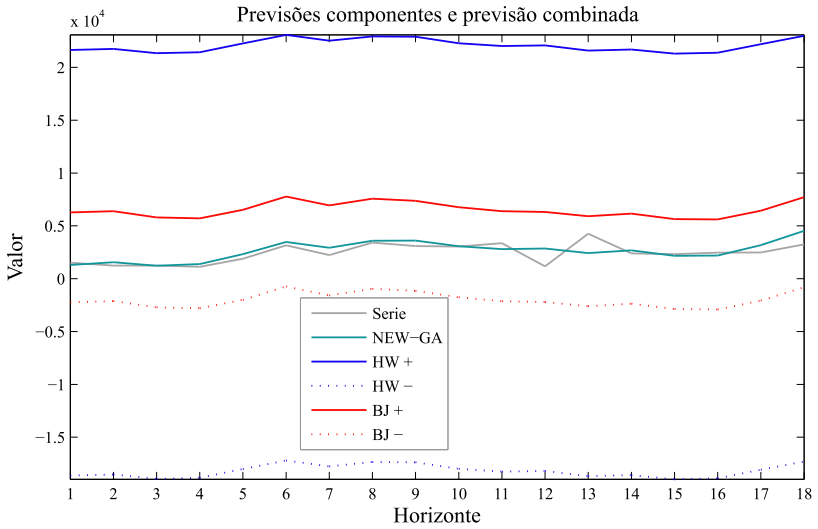


D.9(d): Evolução dos SMAPEs para cada modelo

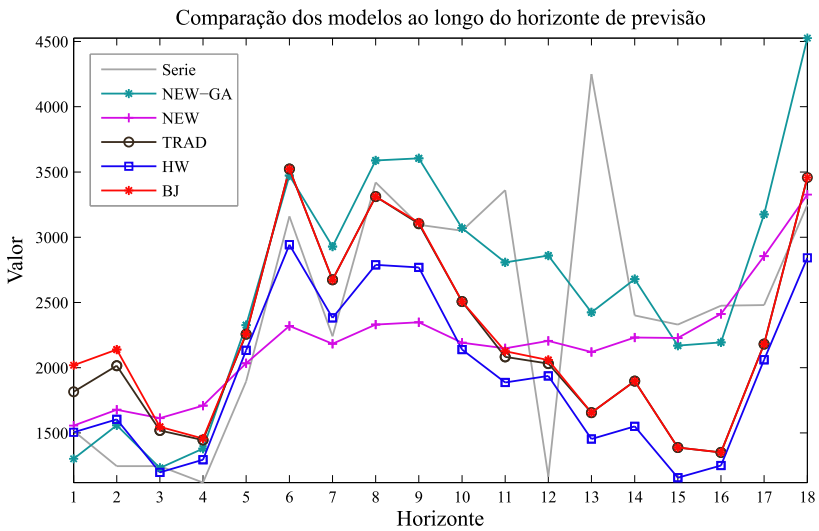
Figura D.9: Resumo dos resultados obtidos para a série NN3-109.



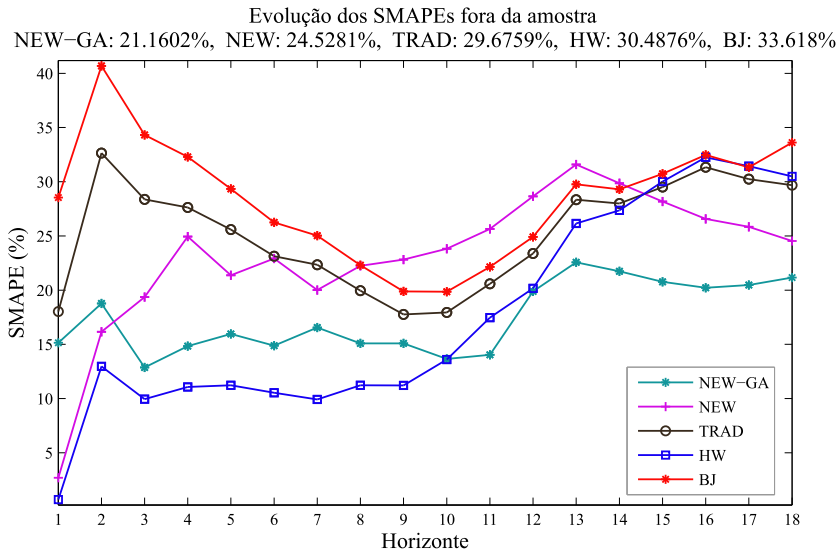
D.10(a): Evolução dos pesos de combinação



D.10(b): Previsões componentes e previsão combinada

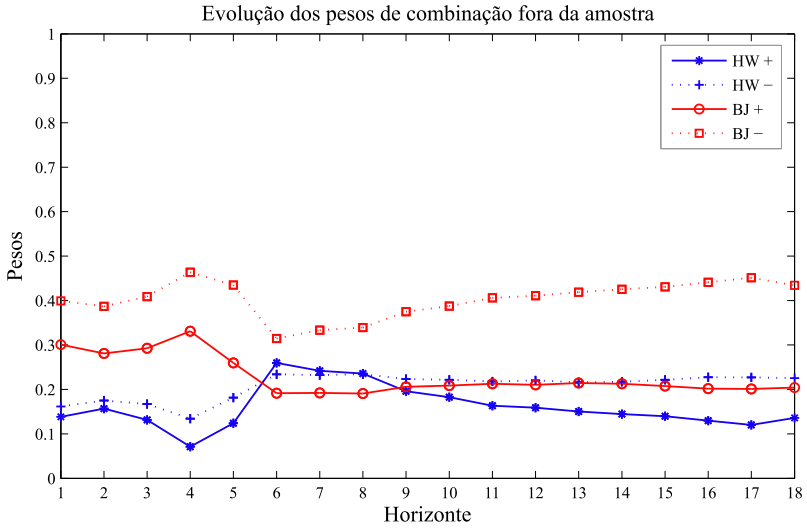


D.10(c): Previsões dos modelos de previsão avaliados

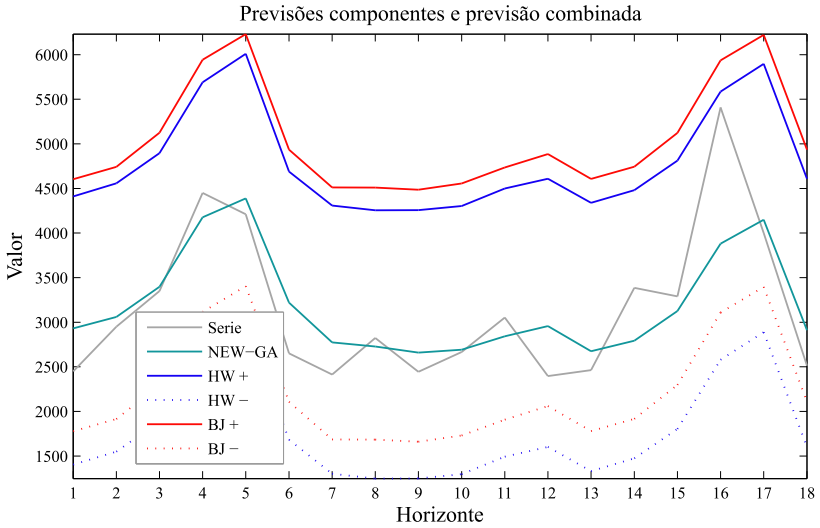


D.10(d): Evolução dos SMAPEs para cada modelo

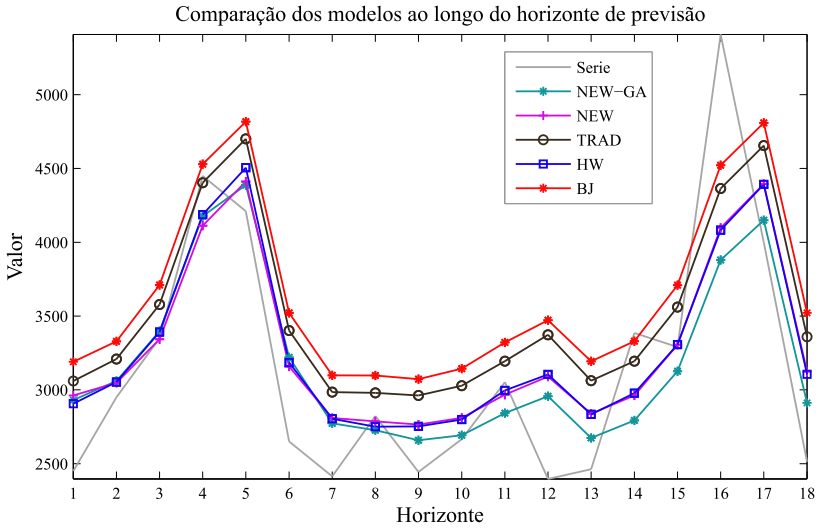
Figura D.10: Resumo dos resultados obtidos para a série NN3-110.



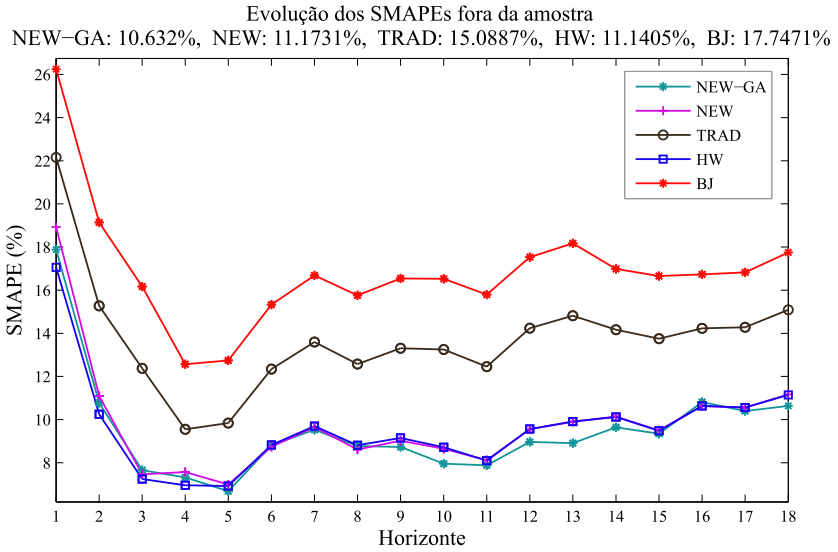
D.11(a): Evolução dos pesos de combinação



D.11(b): Previsões componentes e previsão combinada



D.11(c): Previsões dos modelos de previsão avaliados



D.11(d): Evolução dos SMAPEs para cada modelo

Figura D.11: Resumo dos resultados obtidos para a série NN3-111.

E
Ajuste dos Modelos nos Conjuntos de Treinamento e Validação - Séries Derivados do Petróleo

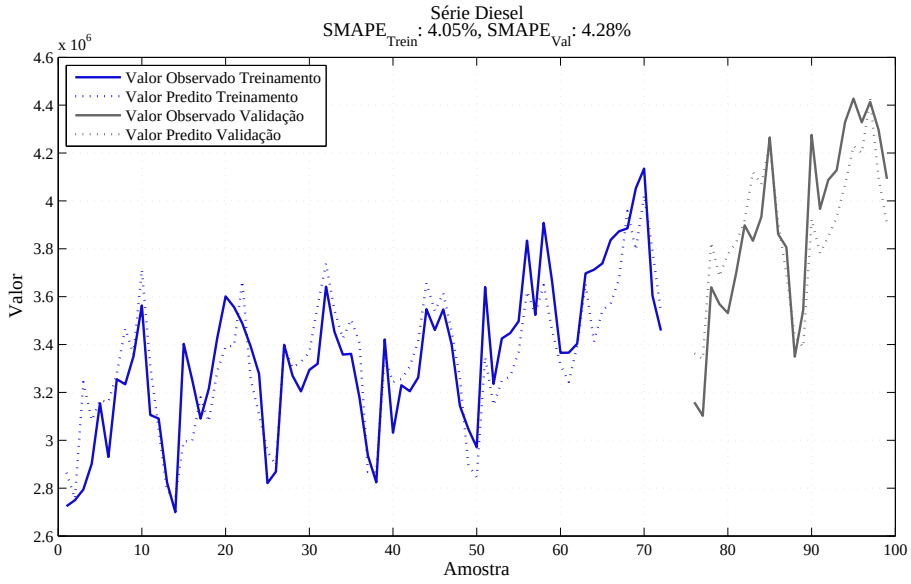


Figura E.1: Ajuste do modelo nos conjuntos de treinamento e validação para a série Diesel.

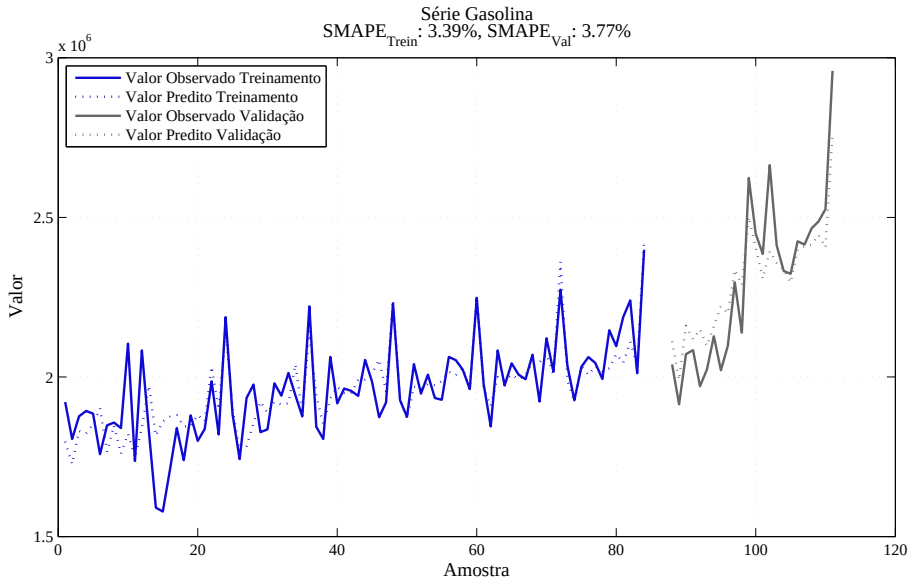


Figura E.2: Ajuste do modelo nos conjuntos de treinamento e validação para a série Gasolina.

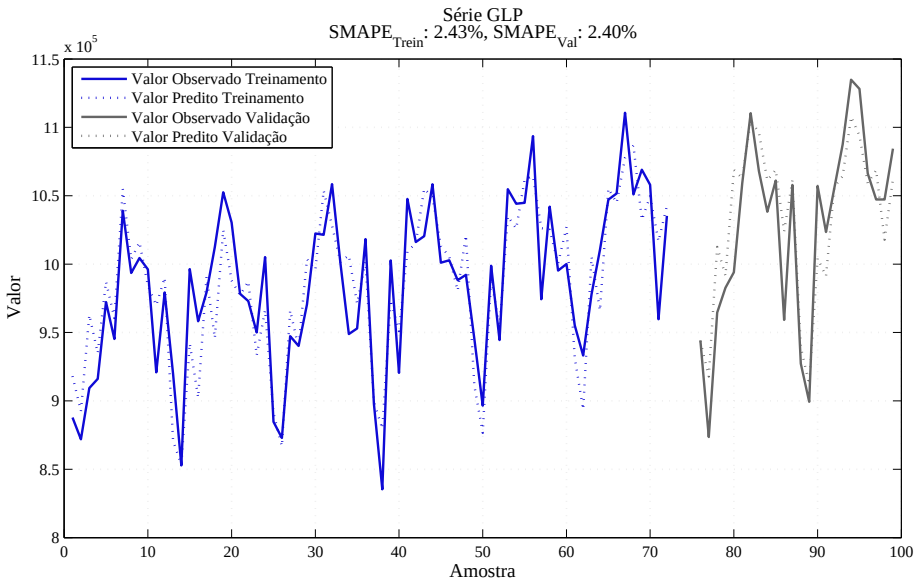


Figura E.3: Ajuste do modelo nos conjuntos de treinamento e validação para a série GLP.

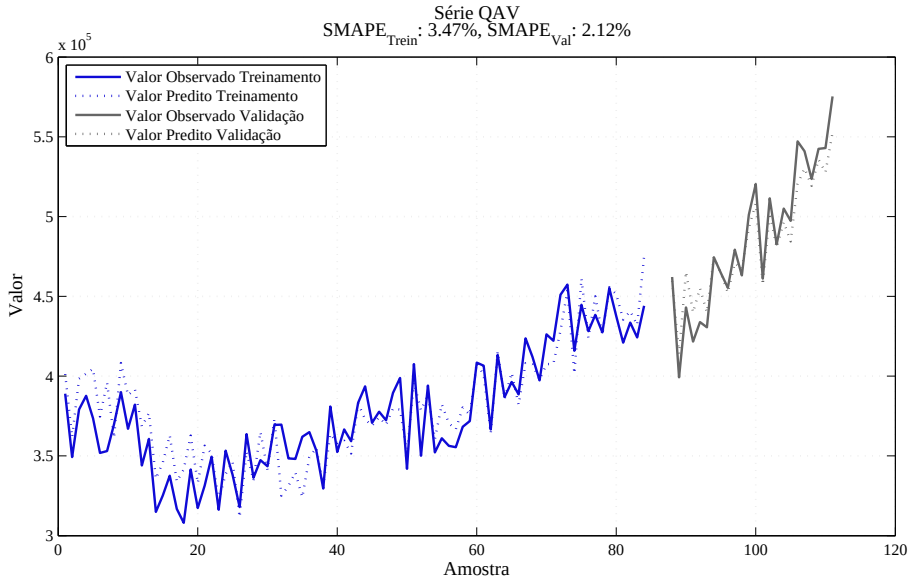


Figura E.4: Ajuste do modelo nos conjuntos de treinamento e validação para a série QAV.

F
Ajuste dos Modelos nos Conjuntos de Treinamento e Validação - Séries da Competição NN3

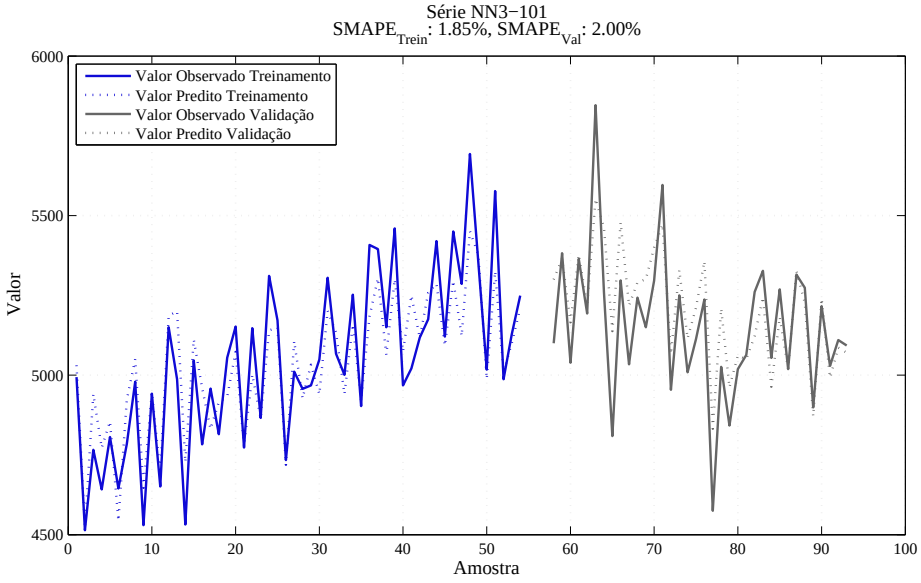


Figura F.1: Ajuste do modelo nos conjuntos de treinamento e validação para a série NN3-101.

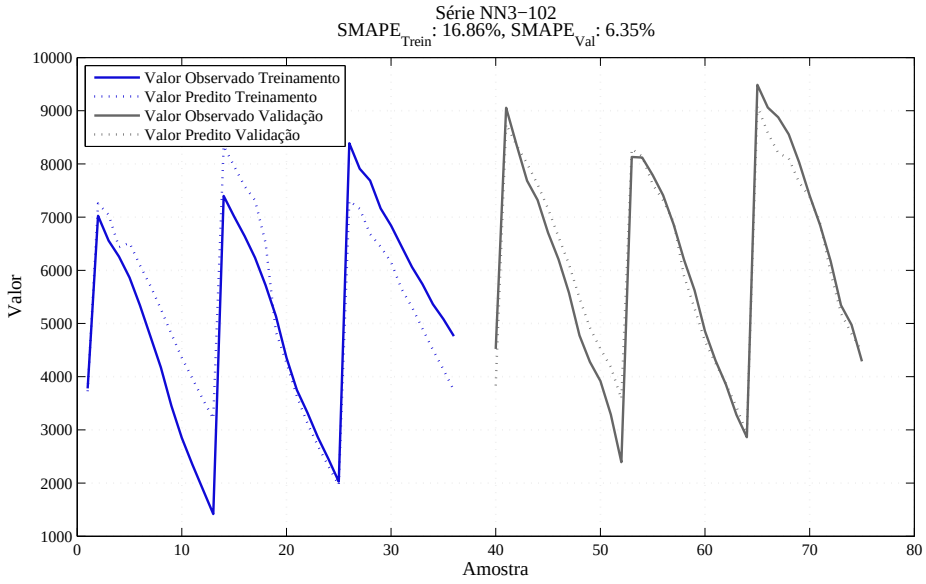


Figura F.2: Ajuste do modelo nos conjuntos de treinamento e validação para a série NN3-102.

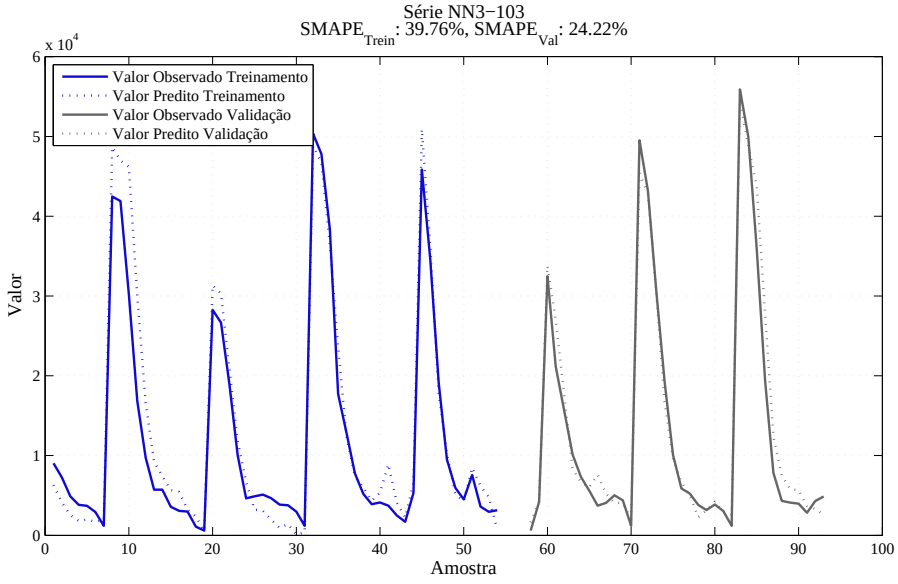


Figura F.3: Ajuste do modelo nos conjuntos de treinamento e validação para a série NN3-103.

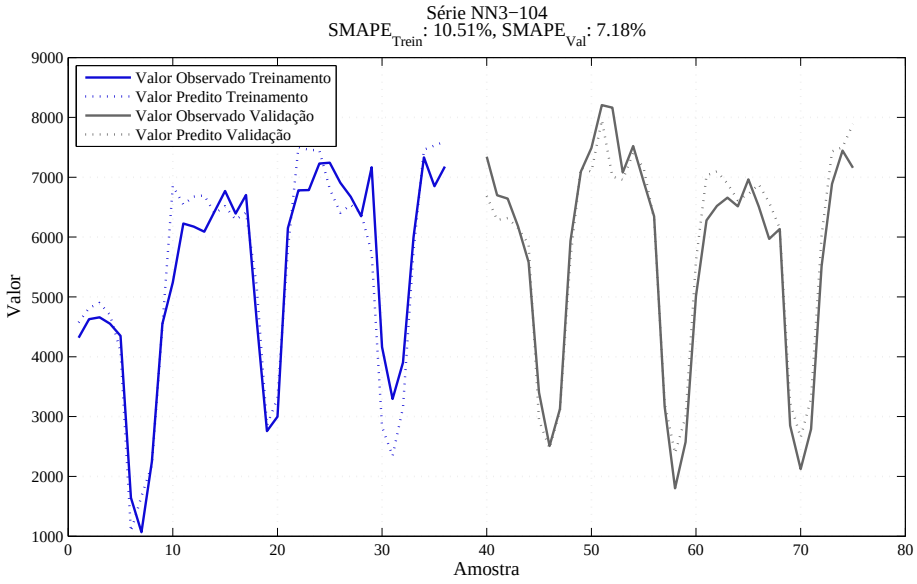


Figura F.4: Ajuste do modelo nos conjuntos de treinamento e validação para a série NN3-104.

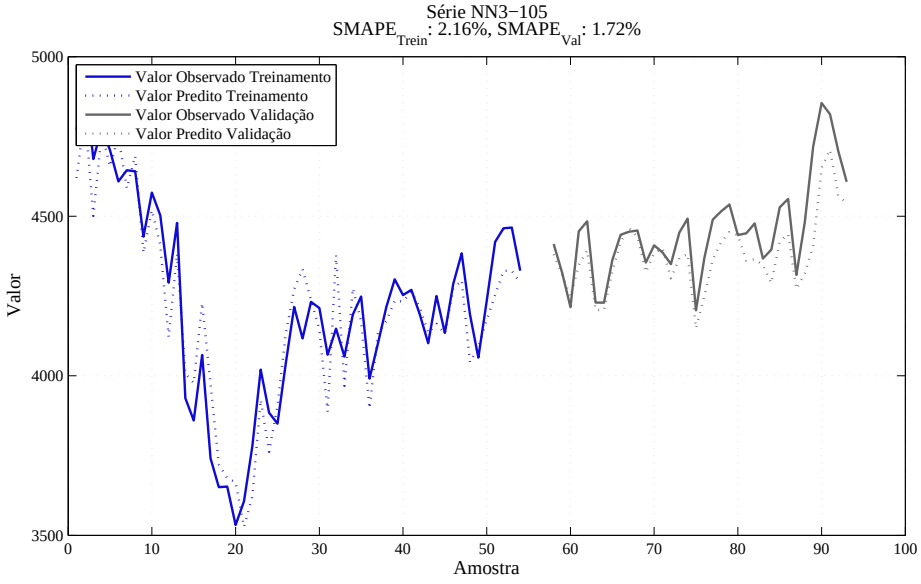


Figura F.5: Ajuste do modelo nos conjuntos de treinamento e validação para a série NN3-105.

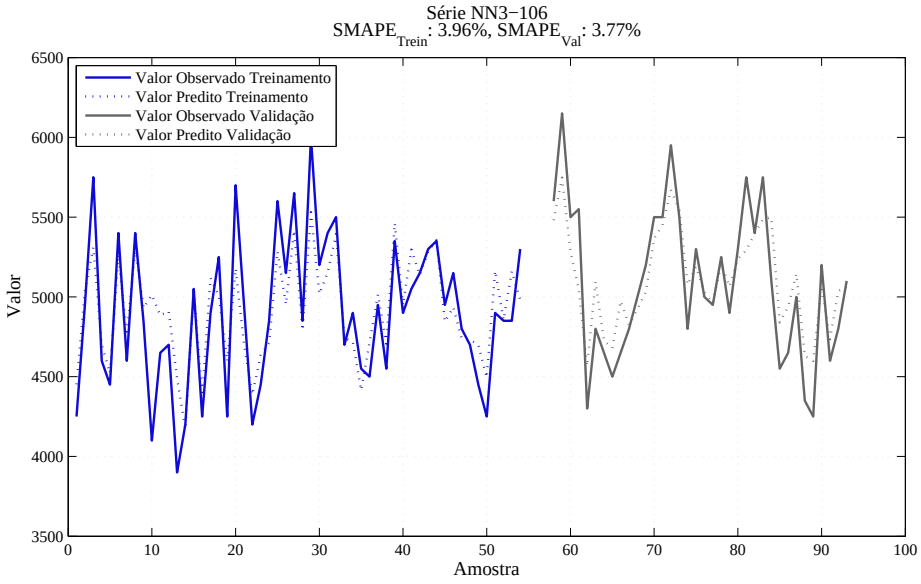


Figura F.6: Ajuste do modelo nos conjuntos de treinamento e validação para a série NN3-106.

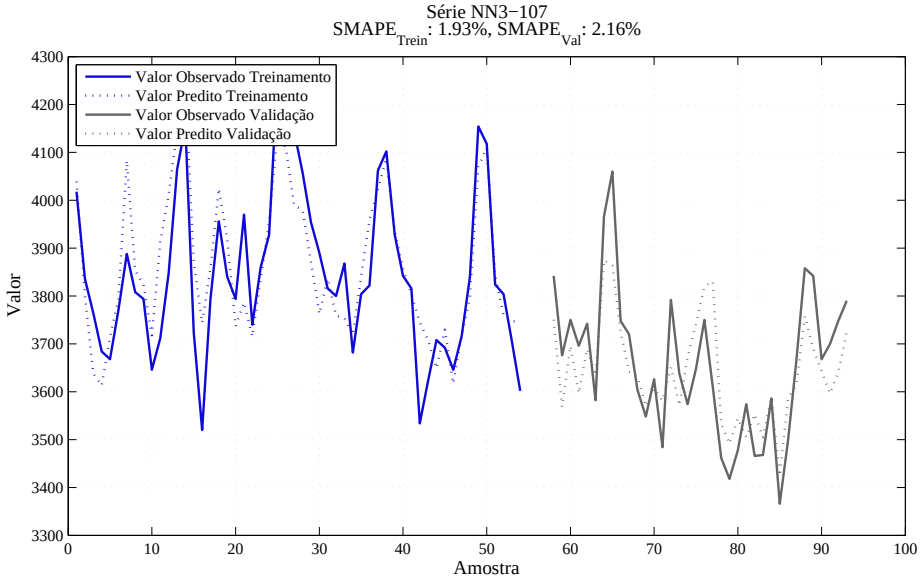


Figura F.7: Ajuste do modelo nos conjuntos de treinamento e validação para a série NN3-107.

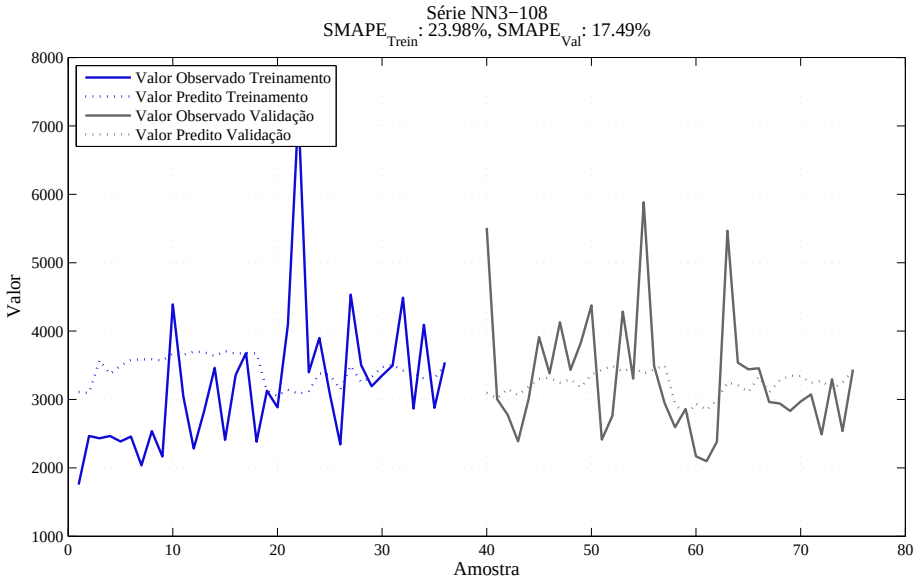


Figura F.8: Ajuste do modelo nos conjuntos de treinamento e validação para a série NN3-108.

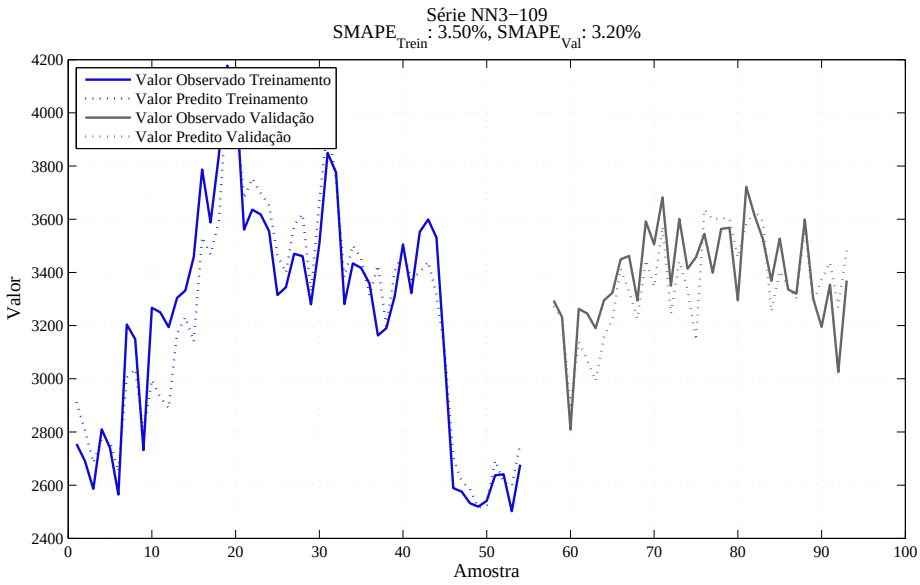


Figura F.9: Ajuste do modelo nos conjuntos de treinamento e validação para a série NN3-109.

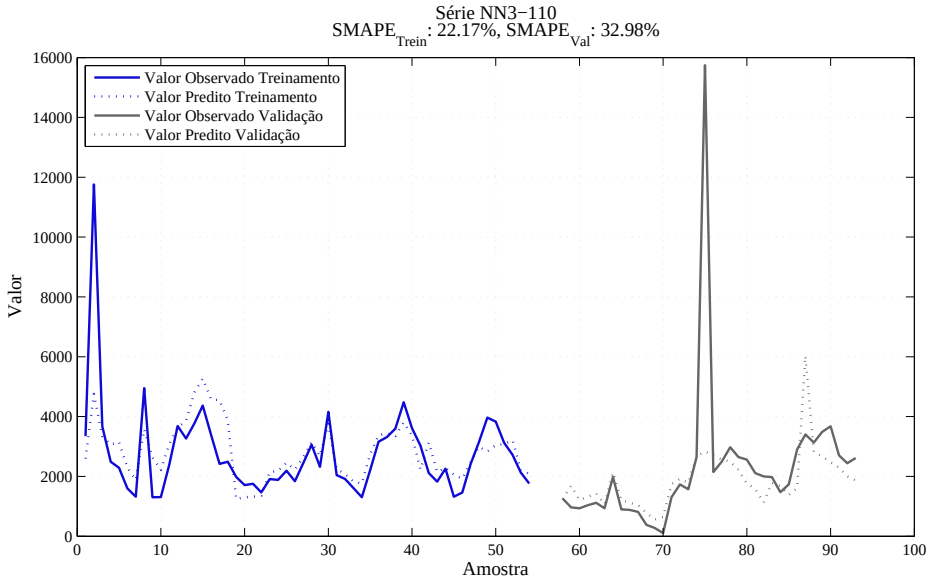


Figura F.10: Ajuste do modelo nos conjuntos de treinamento e validação para a série NN3-110.

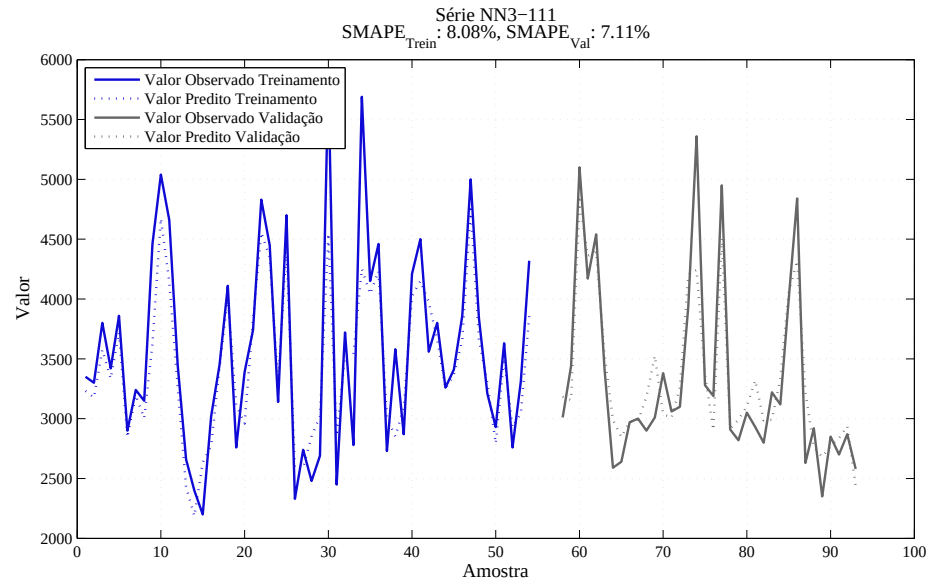


Figura F.11: Ajuste do modelo nos conjuntos de treinamento e validação para a série NN3-111.