Pontifícia Universidade Católica
do Rio de Janeiro

Diego de Lemos Brito da Silva

e-AutoMFIS2: Modelo interpretável para previsão de séries multivariadas baseado em comitês de Sistemas de Inferência Fuzzy e otimização multiobjetivo

Dissertação de Mestrado

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

Orientador : Prof. Marley Maria Bernardes Rebuzzi Vellasco
Coorientador: Prof. Ricardo Tanscheit

Rio de Janeiro
Abril de 2024

Pontifícia Universidade Católica
do Rio de Janeiro

Diego de Lemos Brito da Silva

e-AutoMFIS2: Modelo interpretável para previsão de séries multivariadas baseado em comitês de Sistemas de Inferência Fuzzy e otimização multiobjetivo

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:

Prof. Marley Maria Bernardes Rebuzzi Vellasco
Orientador
Departamento de Engenharia Elétrica – PUC-Rio

Prof. Ricardo Tanscheit
Coorientador
Departamento de Engenharia Elétrica – PUC-Rio

Prof. Karla Tereza Figueiredo Leite
Universidade do Estado do Rio de Janeiro – UERJ

Prof. José Franco Machado do Amaral
Universidade do Estado do Rio de Janeiro – UERJ

Rio de Janeiro, 29 de Abril de 2024

Diego de Lemos Brito da Silva

Graduado em Engenharia Elétrica com ênfase em Sistemas de Potência pela UERJ, teve como interesse durante a graduação o estudo de previsão de séries temporais, o que o levou a procurar um mestrado na área de métodos de apoio à decisão para especialização na área. Suas principais áreas de interesse são *Explainable AI*, *Generative AI* e técnicas de previsão multivariadas.

Ficha Catalográfica

Silva, Diego de Lemos Brito da

e-AutoMFIS2: Modelo interpretável para previsão de séries multivariadas baseado em comitês de Sistemas de Inferência Fuzzy e otimização multiobjetivo / Diego de Lemos Brito da Silva; orientador: Marley Maria Bernardes Rebuzzi Vellasco; coorientador: Ricardo Tanscheit. – 2024.

123 f: il. color. ; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Elétrica, 2024.

Inclui bibliografia

1. Engenharia Elétrica – Teses.
2. Previsão de séries multivariadas. 3. Sistema de Inferência Fuzzy. 4. Interpretabilidade. 5. Otimização multiobjetiva. 6. Algoritmos Genéticos. I. Vellasco, Marley Maria Bernardes Rebuzzi. II. Tanscheit, Ricardo. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Elétrica. IV. Título.

CDD: 621.3

Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Durante esta trajetória do mestrado, do início até a confecção dessa dissertação, algumas pessoas me apoiaram e incentivaram até o final, a elas agradeço.

À minha família, que me possibilitou chegar até aqui, com amor, educação, orientação e, principalmente, incentivo nos momentos de dúvida e dificuldade.

À minha noiva, Tatiana Pereira Correia, que mesmo com suas dificuldades, sempre esteve ao meu lado, dando suporte e carinho, para que eu continuasse nessa jornada até o fim.

À minha orientadora Marley Vellasco e coorientador Ricardo Tanscheit por me fornecerem o conhecimento e direcionamento durante a confecção desta dissertação e, principalmente, pela paciência com a demora que tive para terminar este projeto.

À Thiago Medeiros Carvalho, por toda a atenção e paciência ao me apresentar o modelo e-AutoMFIS, tirar dúvidas e dar dicas de como tocar este trabalho, facilitando a confecção desta dissertação.

Agradeço, também, a todos os meus professores que pavimentaram o caminho do conhecimento que segui até agora e que culminou neste momento.

A todos os amigos que me ajudaram tanto técnica quanto psicologicamente ouvindo desabafos e dúvidas até o final, vocês foram fundamentais para eu chegar nesse momento.

Por fim, gostaria de agradecer a Deus, pois sem Ele, nada disso seria possível.

Resumo

Silva, Diego de Lemos Brito da ; Vellasco, Marley Maria Bernardes Rebuzzi; Tanscheit, Ricardo. **e-AutoMFIS2: Modelo interpretável para previsão de séries multivariadas baseado em comitês de Sistemas de Inferência Fuzzy e otimização multiobjetivo**. Rio de Janeiro, 2024. 123p. Dissertação de Mestrado – Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

No estágio inicial da modelagem de previsão, o foco primordial residia na acurácia dos modelos, muitas vezes negligenciando a interpretabilidade dos resultados. No entanto, à medida que esses modelos ganharam popularidade em diversas áreas, surgiu uma demanda crescente por modelos que fossem tanto confiáveis em relação a previsões quanto interpretáveis. Como resposta a essa demanda, uma série de modelos foram desenvolvidos com o intuito de conciliar esses dois aspectos, que são essencialmente contraditórios. Um exemplo desses modelos desenvolvidos com esse propósito é o e-AutoMFIS, baseado em princípios de lógica fuzzy e na técnica ensemble, com o objetivo de reduzir a dimensionalidade dos problemas e realizar previsões multivariadas. O e-AutoMFIS demonstrou-se promissor em comparação com modelos mais convencionais, superando-os em métricas de acurácia em alguns casos. Sua interpretabilidade, aliada a previsões mais acuradas, o torna uma ferramenta valiosa para análises mais aprofundadas. Apesar dos avanços alcançados, há áreas que demandam aprimoramento, como a configuração dos parâmetros do modelo, ainda realizada por meio de busca exaustiva, e a subamostragem aleatória, que pode necessitar de intervenção manual para otimização dos resultados. Diante desses desafios, esta dissertação propõe a otimização do e-AutoMFIS por meio do desenvolvimento do e-AutoMFIS2. Este trabalho detalha a arquitetura do e-AutoMFIS2, descrevendo as modificações implementadas no processo de subamostragem e a aplicação de um algoritmo genético de otimização multiobjetiva para seleção de hiperparâmetros. Além disso, discute os possíveis benefícios dessas mudanças e avalia os resultados por meio de estudos de caso, comparando-os com seu antecessor e outros modelos tradicionais na literatura, tanto em termos de acurácia quanto de interpretabilidade.

Palavras-chave

Previsão de séries multivariadas; Sistema de Inferência Fuzzy; Interpretabilidade; Otimização multiobjetiva; Algoritmos Genéticos.

Abstract

Silva, Diego de Lemos Brito da ; Vellasco, Marley Maria Bernardes Rebuzzi (Advisor); Tanscheit, Ricardo (Co-Advisor). **e-AutoMFIS2: Interpretable Model for Multivariate Time Series Forecasting based on ensemble of Fuzzy Inference Systems and Multi-Objective Optimization.**. Rio de Janeiro, 2024. 123p. Dissertação de Mestrado – Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

In the initial stage of prediction modeling, the primary focus was on model accuracy, often overlooking the interpretability of results. However, as those models gained popularity in various fields, an increasing demand for accurate and interpretable models emerged. In response to this demand, a series of models was developed to reconcile those two inherently contradictory aspects. An example is the e-AutoMFIS, based on principles of fuzzy logic and ensemble technique, which aims to reduce the dimensionality of problems and make multivariate predictions. The e-AutoMFIS outperforms more conventional modes in accuracy metrics in some cases. Its interpretability, coupled with those gains in accuracy, makes it a valuable tool for deeper analyses. However, some aspects require improvement, such as the configuration of model parameters, still carried out through exhaustive search, and random subsampling, which may require manual intervention for results optimization. Faced with those challenges, this dissertation proposes the optimization of e-AutoMFIS through the development of e-AutoMFIS2. This work details the architecture of e-AutoMFIS2, describing the modifications implemented in the subsampling process and the application of a multiobjective optimization genetic algorithm for parameter selection. Additionally, the potential benefits of those changes are discussed and results are evaluated through case studies, comparing them with its predecessor and other traditional models in the literature regarding accuracy and interpretability.

Keywords

Multivariate time-series forecasting; Fuzzy Inference System; Interpretability; Multi-Objective Optimization; Genetic Algorithm.

Sumário

Lista de figuras

Lista de tabelas

Lista de Abreviaturas

XAI - *Explainable Artificial Intelligence*
SIF – Sistema de Inferência Fuzzy
AG - Algoritmos Genéticos
NSGAII - *Non-dominated sorting genetic algorithm II*
ML – *Machine Learning*
DL – *Deep Learning*
AR – Autoregressivo
MQR – Mínimos Quadrados Restrito
AutoMFIS – *Automatic Multivariate Fuzzy Inference System*
e-AutoMFIS – *Ensemble of Automatic Multivariate Fuzzy Inference System*

1
Introdução

O dinamismo intrínseco ao contexto contemporâneo requer a prontidão de todos os setores da sociedade. Um exemplo evidente disso reside na competição acirrada do mercado, onde a agilidade das empresas em se adaptar a tecnologias emergentes, às oscilações na demanda dos consumidores e às mutações do mercado é imperativa. Nesse contexto, a informação assume um papel fundamental como recurso primordial. É ela que capacita as organizações a tomar decisões embasadas e estrategicamente alinhadas às exigências do ambiente empresarial atual.

Atualmente, a obtenção de informações tornou-se significativamente simplificada, em grande parte devido ao avanço das técnicas de medição de diversas grandezas e, em especial, ao surgimento da internet. Esta última possibilita tanto a fácil localização quanto a disseminação de registros para uma variedade de propósitos. Todos esses aspectos têm contribuído para a criação de bases de dados cada vez mais volumosas e abrangentes, alimentando assim a esfera da informação e impulsionando os avanços na análise e utilização de dados.

Uma das abordagens fundamentais na extração de conhecimento a partir de dados é a análise de séries temporais, que consiste na investigação de dados coletados em intervalos regulares ao longo do tempo. Essa análise é de suma importância para revelar padrões subjacentes nos registros históricos, os quais podem ser empregados na elaboração de projeções futuras. Como resultado, o estudo dessas séries é altamente relevante em uma variedade de domínios, incluindo Economia [1], Finanças [2] e Saúde [3].

A extração de *insights* por meio de séries temporais é uma tarefa desafiadora, pois a maioria das séries exibe comportamentos estocásticos relacionados aos padrões e tendências dos dados. Isso torna a concepção de modelos matemáticos para representação desses dados uma atividade de grande complexidade, que vai além das capacidades do raciocínio humano isolado.

Para lidar com essas complexidades, diversos métodos foram desenvolvidos para modelar os comportamentos das séries temporais com acurácia. Nesse contexto, o campo da Inteligência Computacional emergiu como uma força significativa, fornecendo técnicas relevantes, tais como Redes Neurais, Árvores de

Decisão e Sistemas de Inferência *Fuzzy*. Cada um desses modelos apresenta vantagens e desvantagens distintas em relação à sua aplicabilidade. A escolha entre eles é determinada pelas necessidades e objetivos específicos do usuário, considerando-se a natureza dos dados e o contexto da aplicação em questão.

Nesta dissertação, o modelo proposto fundamenta-se em Sistemas de Inferência *Fuzzy*, amplamente conhecidos na literatura. Esse modelo se destaca por sua modelagem baseada na criação de regras semânticas e dicionários linguísticos e confere ao modelo uma característica distintiva: a interpretabilidade. A combinação de interpretabilidade com uma boa precisão torna os modelos *Fuzzy* altamente promissores para a exploração de novas técnicas e aplicações.

1.1 *Explainable Artificial Inteligence* (XAI)

A acurácia tem sido um dos principais alicerces no desenvolvimento do campo de inteligência computacional. Ao longo do crescimento deste campo, diversos modelos tradicionais foram concebidos com alta exatidão. Entretanto, devido à sua complexidade, tornou-se desafiador compreender como resultados foram obtidos. Por esta razão, esses modelos foram caracterizados como "caixa preta", indicando a impossibilidade de observação interna para um maior entendimento do processo de previsão [4]. Exemplos de tais modelos incluem LSTM (*Long Short-Term Memory*) e redes neurais convolucionais.

É importante destacar que o foco exclusivo na acurácia durante o processo de modelagem pode resultar em limitações significativas no uso prático dos modelos. Em muitos casos, a acurácia dos resultados por si só não é suficiente para embasar a tomada de decisão. Conforme mencionado em [5], a falta de interpretabilidade dos modelos pode dificultar a identificação e mitigação de possíveis riscos. No contexto da área de Saúde, modelos baseados em redes neurais têm demonstrado desempenho promissor na realização de prognósticos médicos [6]. No entanto, esses modelos ainda estão longe de serem infalíveis, sendo essencial que os especialistas compreendam o seu funcionamento para poderem formular suas próprias conclusões.

Em resposta a essas questões, o campo de *Explainable Artificial Intelligence* (XAI) tem ganhado destaque. Para esses tipos de modelos, a interpretabilidade - isto é, a capacidade de entender e explicar como o modelo realiza suas inferências - é tão crucial quanto a própria precisão da previsão. No entanto, é importante destacar que esses objetivos são considerados não complementares, resultando em um fenômeno conhecido como "*tradeoff*", onde a melhoria de um indicador ocorre em detrimento do outro.

A interpretabilidade não é uma característica intrínseca a todos os modelos matemáticos. Em XAI, existem diversas abordagens e aplicações distintas. Elas variam desde algoritmos baseados em árvores de decisão até a aplicação de técnicas em modelos com características de "caixa preta", para torná-los explicáveis. Um exemplo é a técnica LRP (Layer-wise Relevance Propagation), que utiliza a representação estrutural do modelo empregado, mapeando as predições feitas nas camadas mais internas da rede até alcançar a entrada do mesmo [7].

Outra estratégia amplamente reconhecida no contexto da XAI são os modelos baseados em regras. Essas técnicas são extensivamente empregadas devido à sua natureza intuitiva, especialmente as regras de tipo semântico, as quais requerem pouco esforço para serem compreendidas [8]. Na literatura atual, o Sistema de Inferência *Fuzzy* (SIF) destaca-se como um modelo de referência neste tipo de abordagem, em virtude da sua capacidade de simular o raciocínio humano. Ou seja, é possível representar conceitos imprecisos e vagos por meio de regras linguísticas, a fim de descrever o comportamento de forma interpretável do ponto de vista humano, facilitando assim a compreensão dos resultados obtidos.

Com base nesse tipo de técnica, foi desenvolvido um modelo denominado AutoMFIS [9], fundamentado em técnicas de estatística e heurísticas para ajustar os parâmetros do sistema *Fuzzy*. Esse modelo incorpora a capacidade de gerar regras de forma automática por meio de uma geração semi-exaustiva. Os resultados demonstraram uma relação favorável entre acurácia e interpretabilidade. No entanto, o modelo apresentou algumas limitações, principalmente relacionadas à dimensionalidade, tanto em relação ao número de variáveis quanto ao de registros.

Com o objetivo de tornar o modelo viável para conjuntos de dados com alta dimensionalidade, foi desenvolvido o e-AutoMFIS. A principal alteração consistiu na adoção de técnicas de subamostragem de dados e na implementação de *ensemble learning* durante a geração dos sistemas *Fuzzy*. A análise dos resultados apresentados em [10] evidencia o êxito da nova arquitetura em alcançar seus objetivos. No entanto, conforme destacado no próprio estudo, ainda persistem pontos de aprimoramento que podem ser explorados em pesquisas futuras.

Neste estudo, é apresentada a arquitetura e-AutoMFIS2, abordando os pontos de aprimoramento. As principais modificações incluem a revisão do método de subamostragem inicialmente proposto e o uso de uma abordagem de otimização multiobjetiva na seleção de hiperparâmetros utilizando algoritmos genéticos. Ao longo da exposição do trabalho realizado, será demonstrado que essas alterações tornaram o modelo mais eficaz em termos de precisão e interpretabilidade.

1.2 Objetivos

Conforme destacado anteriormente, o e-AutoMFIS demonstrou consistência em relação aos objetivos do trabalho, tornando-o aplicável em conjuntos de dados com alta dimensionalidade. No entanto, há áreas que necessitam de aprimoramento, o que impede que o modelo seja considerado plenamente otimizado.

Um ponto importante é a abordagem de *ensemble* adotada pelo modelo, que realiza a divisão temporal e dimensional de maneira aleatória. Embora essa estratégia possa ser adequada em certos cenários, em outros casos uma intervenção manual torna-se necessária para otimizar a separação. Contudo, essa intervenção manual é custosa e complexa, sugerindo a viabilidade de uma abordagem de divisão que incorpore alguma métrica específica, resultando em subamostras mais otimizadas.

Outro aspecto a ser considerado é a configuração dos hiperparâmetros do modelo, que atualmente emprega uma estratégia de busca exaustiva pela configuração ideal. Entretanto, dada a quantidade de parâmetros a ser ajustados, a busca por todas as combinações possíveis torna-se inviável. Além disso, surge a dificuldade adicional de otimizar métricas de precisão e interpretabilidade contrastantes entre si, o que aumenta a complexidade da seleção dos parâmetros. Nesse contexto, seria vantajoso para o modelo a utilização de técnicas de seleção de parâmetros guiadas por otimização multiobjetiva.

Tendo estas questões em vista, este trabalho propõe-se a mitigar as questões supracitadas via aprimoramentos nas etapas de execução do e-AutoMFIS, usando uma abordagem semiguiada pelas métricas de correlação cruzada e autocorrelação das séries temporais na análise e também na aplicação de otimização multiobjetiva via algoritmos genéticos para seleção de hiperparâmetros. O resultado é o modelo e-AutoMFIS2. As modificações são testadas em diversos estudos de caso, comparando-se com sua versão anterior.

Como objetivo secundário, serão apresentadas as bases teóricas do trabalho para auxiliar o entendimento do modelo e suas respectivas mudanças. Assim, esta dissertação propõe não só demonstrar o funcionamento do modelo, mas também de que modo as mudanças podem influenciar nos resultados de acurácia e interpretabilidade.

1.3 Contribuições

Este trabalho oferece uma contribuição conceitual ao expandir o modelo e-AutoMFIS, explorando técnicas tradicionais e bem estabelecidas para aprimorar sua capacidade de lidar com os problemas identificados em sua versão anterior (AutoMFIS). Introduz uma abordagem inovadora de subamostragem, baseada em correlação e autocorrelação, bem como uma estratégia de seleção de hiperparâmetros com base em algoritmos genéticos, permitindo a escolha da melhor configuração para otimizar o equilíbrio entre acurácia e interpretabilidade.

Outra contribuição, alinhada com trabalho anterior [10], é a busca por uma ponderação equitativa entre os conceitos de acurácia e interpretabilidade. Dada a escassez de análises de interpretabilidade na literatura, este estudo fornece uma análise comparativa abrangente entre os modelos, incluindo uma avaliação das principais métricas relacionadas ao sistema *Fuzzy*, enriquecendo assim a literatura existente.

Por último, destaca-se a contribuição prática do modelo, pois, em comparação com seu predecessor direto, apresentou melhorias em ambos os domínios de análise abordados nesse estudo. Apesar dos objetivos contraditórios, o modelo evidenciou a capacidade de aprimorar tanto a acurácia quanto a interpretabilidade.

1.4 Organização

Este trabalho está dividido em duas partes principais. A primeira parte concentra-se nos fundamentos teóricos essenciais para a compreensão do estudo proposto, abordando os seguintes temas: fundamentos de previsão de séries temporais multivariadas, algoritmos genéticos aplicados à otimização multiobjetiva e sistemas de inferência *Fuzzy* aplicados a séries temporais.

A segunda parte diz respeito ao modelo proposto, apresentando sua estrutura, operação e os resultados obtidos em estudos de caso para avaliação. Todos esses assuntos estão organizados da seguinte maneira:

- **Capítulo 2 - Previsão de Séries Temporais:** fornece um resumo dos conceitos fundamentais para a previsão de séries temporais, juntamente com técnicas de análise e processamento de dados relevantes para o trabalho.
- **Capítulo 3 - Algoritmos Genéticos voltados para otimização:** apresentam-se as bases teóricas para o uso de algoritmos genéticos na otimização de hiperparâmetros, incluindo a exploração da otimização de hiperparâmetros e da otimização multiobjetiva, juntamente com a introdução ao algoritmo NSGAII.
- **Capítulo 4 - Sistemas de Inferência *Fuzzy* aplicados a séries temporais:** aborda os conceitos de sistemas de inferência *Fuzzy* aplicados à previsão de séries temporais, incluindo a teoria relevante e os modelos que servem como base para o modelo proposto
- **Capítulo 5 - e-AutoMFIS2:** apresenta a arquitetura e o funcionamento do modelo proposto, com detalhes sobre cada módulo de execução e comparações com o modelo antecessor.
- **Capítulo 6 - Estudos de caso:** apresenta os problemas utilizados para avaliar o e-AutoMFIS2 e uma comparação com a versão anterior do modelo para avaliar o impacto das alterações.
- **Capítulo 7 - Conclusão:** discute as vantagens e desvantagens da nova abordagem, bem como possíveis modificações para futuros aprimoramentos.

2
Previsão de séries temporais

A previsão de séries temporais representa um desafio de alta complexidade na área de Ciência de Dados. No entanto, quando realizada com precisão, é de imensa importância em setores como economia, indústria, logística e outros. Portanto, um considerável esforço é dedicado a esse campo, visando garantir a acurácia das previsões. Isso se deve ao fato de que uma previsão incorreta pode ter consequências tão prejudiciais quanto uma decisão tomada sem embasamento adequado.

A literatura oferece uma diversidade de exemplos que ilustram a aplicabilidade das séries temporais em diversos domínios. Por exemplo, a previsão do consumo de energia em edifícios é de suma importância para pesquisas voltadas à eficiência e sustentabilidade energética, conforme destacado por [11]. Em outro contexto, na área da saúde, um exemplo de destaque é o esforço realizado para prever o pico de surtos de COVID-19 no início da pandemia, como documentado por [12]. Além disso, na esfera econômica, a previsão da taxa de desemprego no Brasil, conforme abordado por [13], também é extremamente relevante. Esses casos evidenciam a versatilidade e importância da previsão baseada em séries temporais, destacando seu papel significativo em uma ampla gama de áreas.

Atualmente, uma variedade de técnicas está disponível para extrair os padrões subjacentes às séries temporais, abrangendo desde métodos estatísticos até abordagens mais avançadas de Inteligência Computacional, como o *Deep Learning*. No entanto, a eficácia dessas técnicas depende da análise prévia das séries temporais para compreender os comportamentos das variáveis em estudo. Isso ressalta a importância da extração de informações preliminares antes da apresentação dos dados ao modelo, o que é crucial para alcançar um desempenho satisfatório.

Este capítulo tem como objetivo apresentar os fundamentos das séries temporais, fornecendo uma base essencial para orientar o processo de previsão. Além disso, serão abordadas técnicas que desempenham um papel crucial na realização dos estudos de caso neste trabalho.

2.1
Fundamentos de séries temporais e suas previsões

Do ponto de vista técnico, séries temporais podem ser definidas como sequências ordenadas de valores registrados em intervalos regulares de tempo [14]. São classificadas como discretas se o conjunto de instantes no tempo em que os dados são registrados for numerável; caso contrário, a série é considerada contínua. Um exemplo prático de séries temporais pode ser observado na figura 2.1, que representa as observações de manchas solares coletadas ao longo do século XVII.

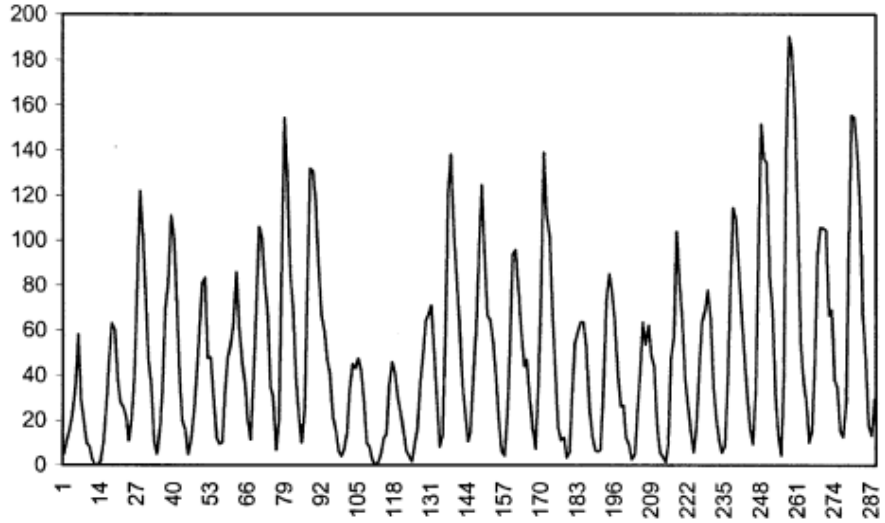

Figura 2.1: Exemplo de um série temporal [15]

Como mencionado anteriormente, compreender as séries temporais em questão é crucial para alcançar previsões confiáveis que possam ser utilizadas na tomada de decisões. Portanto, antes de iniciar o processo de previsão, é necessário realizar uma análise preliminar dos dados. Todos os conjuntos de dados têm suas peculiaridades, mas há informações intrínsecas comuns a todos eles [16]. Essas informações podem ser separadas em quatro componentes principais: tendência, ciclo, sazonalidade e ruído aleatório. A Figura 2.2 exemplifica a decomposição das séries temporais, destacando cada um desses elementos.

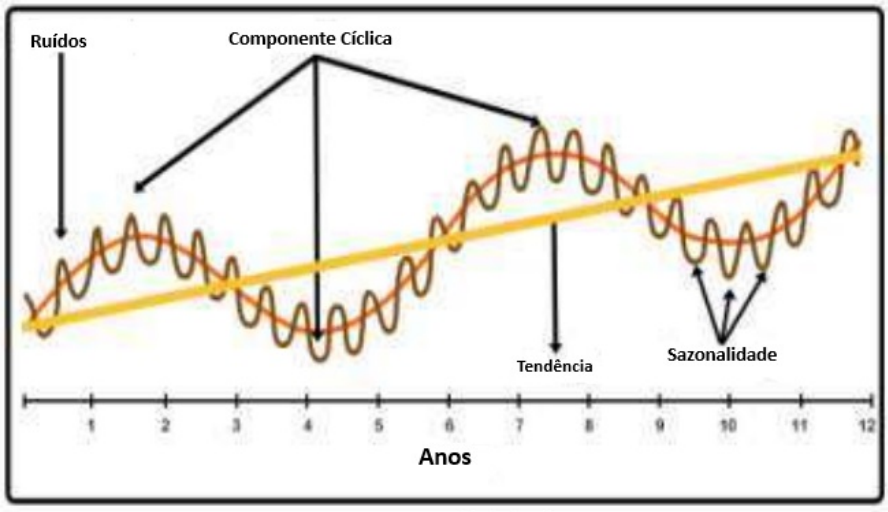

Figura 2.2: Apresentação das componentes de uma série temporal[17]

A tendência é um componente que caracteriza o comportamento de longo prazo da série temporal, indicando se há uma tendência de aumento, diminuição ou estabilidade geral na série. Além disso, a tendência pode assumir padrões mais complexos, manifestando-se de forma não linear. Existem vários exemplos que ilustram essas possibilidades: taxas de escolaridade e analfabetismo na população mundial representam casos de tendência positiva e negativa, respectivamente. Para casos não lineares, como o desempenho de ações no mercado financeiro, a tendência pode ser mais complexa. No contexto das previsões, a compreensão da tendência é fundamental, uma vez que essa informação pode ser empregada na projeção de eventos futuros e também na definição dos modelos a serem utilizados. Vale ressaltar que alguns modelos de previsão supõem a estacionariedade dos dados ao longo do tempo [18].

A sazonalidade refere-se a padrões identificados em períodos específicos, com duração inferior a um ano, que se repetem ciclicamente. Um exemplo clássico é o aumento nas vendas de produtos durante períodos festivos, como Natal e Páscoa. Essa componente é fundamental no processo de previsão de dados, pois fornece informações que podem ser extrapoladas para períodos semelhantes no futuro.

Além da sazonalidade, as flutuações cíclicas na série temporal também representam variações periódicas nos dados, mas com uma duração mais longa, geralmente superior a um ano. Um exemplo típico de uma componente cíclica são os fenômenos El Niño e La Niña, que podem influenciar o padrão de precipitação anual em determinadas regiões [19].

Por fim, há a componente de ruído aleatório, que consiste em variações não atribuíveis aos demais componentes. Essas flutuações são geralmente resultantes de eventos imprevisíveis ou aleatórios, o que as torna desafiadoras

de modelar. Ao contrário das outras componentes, não existem técnicas estatísticas específicas para analisar esses ruídos. No entanto, é importante mantê-los durante o processo de modelagem para que o modelo seja capaz de capturar essas variações e produzir resultados mais precisos.

Outra questão de extrema relevância para a extração de conhecimento em séries temporais é a forma como os elementos mencionados anteriormente podem ser combinados. Na literatura, são descritas duas abordagens de modelagem matemática para essa associação: os modelos aditivo e multiplicativo. O modelo aditivo parte da suposição de que os elementos da série são independentes entre si e podem ser facilmente desacoplados [20]. Por outro lado, o modelo multiplicativo pressupõe uma maior interdependência entre esses elementos. As formulações dos modelos aditivo e multiplicativo são apresentadas pelas Equações 2-1 e 2-2, respectivamente, onde $Y(t)$ representa a série temporal e $T(t)$, $S(t)$, $C(t)$ e $I(t)$ correspondem, respectivamente, à tendência, sazonalidade, componente cíclica e ruído aleatório no instante de tempo t.

$$Y(t) = T(t) + S(t) + C(t) + I(t) \tag{2-1}$$

$$Y(t) = T(t) * S(t) * C(t) * I(t) \tag{2-2}$$

Após a fase inicial de análise da série temporal para extração de conhecimento e padrões, avança-se para a etapa de previsão dos dados. Assim como na análise das séries temporais, há vários aspectos a serem considerados antes do desenvolvimento de um modelo para inferência de eventos futuros. Os problemas de previsão podem ser categorizados em três grupos distintos: número de variáveis, horizonte de previsão e características estatísticas. Esses aspectos devem ser avaliados inicialmente, pois auxiliam na seleção do modelo mais adequado para o problema em questão.

Em relação ao número de variáveis, as previsões podem ser classificadas como univariadas, multivariadas ou baseadas em julgamentos [21]. Essas categorizações são cruciais para a definição do problema e das relações que serão extraídas do modelo matemático [10].

– **Previsões Univariadas** - São realizadas por meio da projeção de dados passados de uma única variável. A equação 2-3 apresenta esta relação, onde $\hat{y}t$ e yt representam, respectivamente, os valores previstos e reais da série no instante t, h é a quantidade de passos previstos à frente e $f(.)$ é a função que define o modelo. Este tipo de modelagem é indicado quando não há nenhuma variável extrínseca para complementar a previsão. Outra vantagem é a maior variedade de métodos disponíveis para utilização, uma vez que problemas de natureza multivariada podem ser decompostos em vários problemas univariados.

$$\hat{y}_{t+h} = f(y_t, y_{t-1}, y_{t-2}, ...) \tag{2-3}$$

– **Previsões multivariadas** - São previsões em que a variável de interesse depende não apenas dela mesma, mas também de outras variáveis adicionais para projetar eventos futuros. A equação 2-4 ilustra essa relação, onde $y_2, ..., y_s$ representam as séries usadas na previsão da série alvo y_1, juntamente com suas defasagens próprias. A inclusão de informações adicionais visa a fornecer *insights* extras que podem contribuir para uma previsão mais precisa do modelo. No entanto, é necessário exercer cautela ao adicionar variáveis extras, pois isso pode aumentar a complexidade do problema a ser modelado e introduzir mais erros ao sistema. Além disso, muitas vezes não temos acesso às informações futuras dessas variáveis adicionais, o que exige que sejam previstas em conjunto com a série de interesse. Portanto, nem sempre as previsões multivariadas apresentarão desempenho superior às univariadas [22, 23].

$$\hat{y}_{i,t+h} = f(y_{1,t}, y_{2,t}, ...y_{S,t}, y_{1,t-1}, ...) \tag{2-4}$$

– **Julgamento** - É uma modalidade de previsão que se fundamenta no discernimento de um especialista em uma determinada área. Embora seja menos difundida na literatura por não se basear em dados concretos, atualmente tem sido empregada em conjunto com outros métodos de previsão para enriquecer os modelos desenvolvidos e, consequentemente, melhorar a precisão das previsões realizadas [24, 25].

Quanto ao horizonte de previsões das séries temporais, é possível identificar quatro grupos distintos: longo prazo, médio prazo, curto prazo e curtíssimo prazo. Essa classificação é fundamental para compreender o comportamento do problema e selecionar o modelo mais adequado [10].

- **Longo-prazo** - Refere-se a previsões que abrangem grandes períodos de tempo em relação ao intervalo de observação. Um exemplo são as projeções sobre os efeitos das mudanças climáticas globais, que visam a compreender o comportamento geral do fenômeno [26]. Modelos físicos ou estatísticos são comumente utilizados para esse tipo de previsão.
- **Médio-prazo** - Este tipo de previsão abrange um horizonte intermediário, geralmente de meses a anos. Embora menos abrangente do que as previsões de longo prazo, é crucial para antecipar tendências e comportamentos em um futuro próximo. Pode ser aplicado, por exemplo, para orientar o planejamento de produção em uma empresa com base na previsão de demanda.
- **Curto-prazo** - Refere-se a previsões realizadas em intervalos de alguns dias ou poucos meses. Um exemplo é o uso de previsões de curto prazo para o planejamento do controle de tráfego [27]. Nesses casos, uma variedade de metodologias pode ser aplicada, incluindo modelos de aprendizado de máquina, estatísticos ou físicos.
- **Curtíssimo-prazo** - São previsões que abrangem apenas algumas horas à frente dos dados coletados. Um exemplo é a previsão do consumo de energia elétrica, crucial para empresas do setor elétrico, especialmente para operações em tempo real, segurança e controle de frequência de carga [28]. Modelos baseados em aprendizado de máquina e estatísticos são comumente empregados nesses cenários devido à sua capacidade de capturar variações rápidas nas séries temporais [29].

Por último, as propriedades estatísticas da previsão podem ser divididas em dois grupos: estacionárias e não estacionárias. No entanto, antes de definilas, é importante apresentar alguns conceitos estatísticos que são usados como referência para determinar se uma série temporal é considerada, ou não, estacionária.

Em uma série temporal x com T observações, é possível extrair informações estatísticas importantes, como o valor esperado, variância e autocorrelação. Os dois primeiros são apresentados nas Equações 2-5 e 2-6, respectivamente, enquanto o último será apresentado posteriormente neste trabalho. Devido à probabilidade $p(x)$ em séries temporais ser uniforme, o valor esperado

($E[X]$) da série x é equivalente à sua média μ_x. Também é relevante mencionar que, para duas séries x e y, é possível calcular a covariância ($Cov(x, y)$), apresentada na Eq. 2-7. A partir destes conceitos, pode-se determinar o que é uma série estacionária.

$$E[x] = \sum_{t=1}^{T} x(t)p(x) \tag{2-5}$$

$$Var(x) = E[x - \mu_x] \tag{2-6}$$

$$Cov(x, y) = E[(x - \mu_x)(y - \mu_y)] \tag{2-7}$$

– **Séries Estacionárias** - Uma série é estacionária quando suas propriedades estatísticas permanecem constantes ao longo de toda sua extensão. No entanto, essa definição está sujeita a nuances, como no caso em que a série pode ser considerada fracamente estacionária, caso sua média e autocovariância não sejam constantes ao longo do tempo, mas sua variância seja finita [30]. Além disso, é possível encontrar na literatura séries estacionárias de até a enésima ordem [31].

– **Séries não-estacionárias** - são caracterizadas por propriedades estatísticas, como média e variância, que variam ao longo do tempo. A identificação dessas séries envolve a observação de se essas propriedades variam significativamente ao longo da extensão da série. A presença de componentes como tendência ou sazonalidade pode ser um indicativo de não estacionariedade das séries. No entanto, para confirmar se uma série é estacionária ou não, recomenda-se o uso de testes estatísticos, como o teste de Dickey Fuller [32], para complementar a inspeção visual.

2.2 Técnicas fundamentais

A exposição dos princípios fundamentais para previsão de séries temporais evidencia que várias características das séries podem ser prejudiciais para o processo de modelagem do problema. Por exemplo, a presença de sazonalidade pode introduzir padrões repetitivos que dificultam a identificação de tendências de longo prazo. Isso pode levar a modelos que capturam apenas a variação sazonal, negligenciando as tendências subjacentes, o que pode resultar em previsões de fraca acurácia. Por conseguinte, foram desenvolvidas técnicas para mitigar ou até mesmo eliminar esses efeitos indesejáveis.

Embora a literatura ofereça uma variedade de métodos para o tratamento de séries temporais, esta seção abordará exclusivamente as técnicas que serão empregadas na fase experimental deste estudo.

2.2.1 Médias móveis

A coleta de dados necessária para a construção de séries temporais está sujeita a diversos erros, sejam eles resultantes de falhas humanas ou não. Isso pode resultar em séries com um alto nível de ruído, o que por sua vez pode prejudicar o processo de previsão. Para mitigar esses ruídos, várias técnicas são apresentadas na literatura, sendo uma das mais clássicas o método de médias móveis.

A técnica de médias móveis é empregada para suavizar flutuações de curto prazo em séries temporais, permitindo evidenciar componentes de longo prazo, como a tendência e a componente cíclica [33]. Este método, resumido pela Eq. 2-8, consiste na definição de uma janela de tamanho M. Em seguida, o processo calcula a média dos primeiros M valores da série, deslocando então essa janela um passo à frente para calcular novamente a média dos próximos M valores. Esse procedimento se repete até alcançar a última observação da série. Ao final, uma nova série é gerada, composta pelas médias dos valores originais que compunham cada janela [34].

$$y_i = \frac{1}{M} \sum_{k=i}^{i+M} y_k \tag{2-8}$$

A suavização dos dados ruidosos por meio de médias móveis desempenha um papel crucial na etapa de modelagem, agindo como um mecanismo de regularização [35]. A redução dos ruídos aleatórios pode tornar os modelos menos sensíveis a variações locais ou pequenas irregularidades, diminuindo as chances de *overfitting* e, consequentemente, estabilizando o processo de previsão.

2.2.2 Remoção de tendências

Outro aspecto relevante é a presença de tendência nas componentes da séries. Embora tendências possam conter informações cruciais para o entendimento geral da série em análise, é importante observar que sua existência pode ser indesejável para muitos modelos. Por exemplo, alguns modelos estatísticos partem da suposição de que as séries possuem média e variância constantes ao longo do tempo, e a presença de tendência nos dados pode violar essa premissa. Portanto, a identificação e remoção da tendência são cruciais para garantir a adequada execução do processo de modelagem.

Na literatura, diversas técnicas de remoção de tendência são mencionadas, incluindo a remoção de tendência linear e não-linear [36], bem como o filtro Loess [37]. Neste trabalho, o foco será na remoção de tendência por meio da regressão linear, uma vez que a maioria dos casos pode ser representada por tendências lineares. Este método visa a se encontrar a melhor reta que represente a série temporal. Para isso, são ajustados os parâmetros de uma função de primeiro grau a e b, como apresentado na Equação 2-9. Determinada a melhor reta, esta é subtraída da série temporal original para remover a sua tendência.

$$y(t)_{trend} = ax(t) + b \tag{2-9}$$

Esse procedimento por si só não é suficiente para tornar a série estacionária. No entanto, ele é capaz de ajustar a série em torno de um valor médio, o que é um comportamento desejável para o modelo proposto neste trabalho.

2.2.3 Diferenciação de séries

Como mencionado anteriormente, a remoção da tendência serve para manter os valores das variáveis oscilando em torno de um intervalo constante ao longo do tempo [10]. No entanto, outras componentes, como sazonalidade e variações cíclicas, podem manter a série não estacionária. Em muitos cenários, a estacionariedade da série é um requisito fundamental para a obtenção de previsões mais precisas. Nesse contexto, a técnica de diferenciação emerge como uma alternativa viável para transformar a série, facilitando a obtenção da estacionariedade necessária para modelos de previsão mais robustos.

A diferenciação é realizada por meio de subtrações consecutivas entre as observações que compõem a série temporal. Essa técnica tem como objetivo eliminar as componentes das séries temporais que as tornam não estacionárias, como tendência, sazonalidade e comportamento cíclico. Embora existam diferenciações até a enésima ordem, a diferenciação de primeira ordem é suficiente para tornar a série estacionária na maioria dos casos.

Tendo isso em mente, o foco deste trabalho será na apresentação da diferenciação de primeira ordem, apresentada pela Eq. 2-10, onde o valor diferenciado $y^{'}(t)$ é obtido subtraindo-se o valor da série y na posição t pelo do seu antecedente, na posição $(t-1)$. Esse procedimento é repetido para cada ponto de dados da série até o final da sequência registrada, resultando em uma nova série diferenciada. Um aspecto destacado desta técnica é a facilidade com que pode ser revertida para obter os valores originais.

$$y^{'}(t) = y(t) - y(t-1) \tag{2-10}$$

2.2.4 Correlação Linear

Como mencionado anteriormente na seção sobre previsões multivariadas, a inclusão de variáveis extrínsecas não garante necessariamente um melhor desempenho na etapa de previsão dos dados. Portanto, é crucial analisar cuidadosamente as variáveis antes de incorporá-las ao processo de modelagem. Um método comumente utilizado na literatura para essa avaliação é a análise da correlação linear entre as variáveis, que ajuda a identificar possíveis relações lineares entre elas.

A correlação avalia o grau de dependência linear entre as variáveis. Diversos métodos estão disponíveis para realizar esse cálculo, sendo a coeficiente de correlação de Pearson um dos mais conhecidos. Representada pela Eq. 2-11, esta medida considera a covariância ($Cov(x_{t-i}, y_{t-j})$) entre as séries x_{t-i} e y_{t-j} e suas respectivas variâncias $Var(x_{t-i})$ e $Var(y_{t-j})$, juntamente com suas respectivas defasagens i e j. Seu valor varia entre -1 e 1, onde 1 indica uma correlação positiva máxima, -1 indica uma correlação negativa máxima e 0 indica ausência de correlação entre as séries.

$$\rho(x_{t-i}, y_{t-j}) = \frac{Cov(x_{t-i}, y_{t-j})}{\sqrt{Var(x_{t-i})Var(y_{t-j})}} \tag{2-11}$$

Essas informações são comumente empregadas para estabelecer um limiar de relevância para as séries que serão utilizadas na etapa de previsão, mantendo-se apenas as variáveis que realmente contribuem para a predição. É importante ressaltar que a avaliação da correlação por si só pode não ser suficiente para descartar uma variável, pois este método considera apenas a relação linear entre as variáveis, desconsiderando possíveis relações não-lineares. Portanto, é necessário complementar essa técnica com outras metodologias para uma análise mais abrangente e precisa.

2.2.5 Autocorrelação

Além da relevância da correlação linear entre variáveis, é crucial avaliar também a correlação entre as defasagens de uma série temporal, pois os valores passados podem exercer influência sobre os valores futuros. Essa análise permite identificar padrões de dependência temporal na série, fornecendo *insights* sobre como as observações passadas podem afetar as futuras. O exame da correlação entre as defasagens permite entender melhor a dinâmica temporal dos dados e capturar informações importantes para a modelagem e previsão da série temporal.

A autocorrelação ($\rho(k)$) é uma medida estatística que avalia a relação entre uma série temporal e suas defasagens, indicando a similaridade entre os valores em diferentes instantes no tempo. Essa técnica é comparável à correlação de Pearson, contudo, ao invés de medir a correlação entre duas séries distintas, ela analisa a relação entre uma série e suas próprias defasagens. A Eq. 2-12 exemplifica como o cálculo da autocorrelação segue uma lógica similar ao da correlação cruzada, calculando o quociente entre a covariância $Cov(y_t, y_{t-k})$ de y_t e a k-ésima defasagem y_{t-k} e a variância de y_t, destacando a dependência entre a série temporal e seus valores anteriores. Portanto, o método de avaliação da autocorrelação é similar ao apresentado anteriormente, com foco na análise das relações entre a série e seus valores passados.

$$\rho(k) = \frac{Cov(y_t, y_{t-k})}{Var(y_t)} \tag{2-12}$$

A função de autocorrelação desempenha um papel crucial na identificação das defasagens mais relevantes e na detecção de padrões sazonais em séries temporais. Essas informações são usadas de diversas maneiras, incluindo a seleção de modelos e a determinação das defasagens mais importantes para a previsão. A análise da função de autocorrelação permite identificar os atrasos com correlação significativa, auxiliando na escolha dos atrasos a serem incluídos nos modelos. Além disso, os padrões sazonais na função de autocorrelação oferecem *insights* sobre a natureza cíclica dos dados, permitindo ajustes precisos nos modelos para capturar essas variações temporais.

3
Algoritmos Genéticos voltados para otimização

Algoritmos metaheurísticos são técnicas de otimização que se fundamentam, em grande parte, em conceitos e teorias oriundos da biologia, como os processos evolutivos e a inteligência de enxames [38]. Essas abordagens podem ser categorizadas em dois grupos principais: algoritmos de solução única e aqueles baseados em população.

Nas metaheurísticas de solução única, os algoritmos operam com uma única solução em cada iteração, a qual é iterativamente modificada na busca pela solução ótima ou próxima a ela. Exemplos comuns incluem o recozimento simulado [39], busca tabu [40] e busca de vizinhança variável [41]. Esses métodos são adequados para espaços de busca menores ou restritos, pois podem ficar estagnados em ótimos locais em problemas mais complexos [42].

Os algoritmos baseados em população, ao contrário dos de solução única, operam com múltiplas soluções em potencial para a determinação da solução ótima. Estas metaheurísticas mantêm a diversidade da população para evitar que as soluções fiquem estagnadas em ótimos locais [43]. Portanto, tais algoritmos são mais adequados para problemas caracterizados por espaços de busca complexos e diversificados.

Dentre os algoritmos metaheurísticos baseados em população, destacam-se os algoritmos genéticos (AG), os quais se fundamentam em mecanismos de seleção natural e genética. A inspiração primordial reside na teoria da seleção natural, na qual os indivíduos mais adaptados possuem maior probabilidade de sobrevivência e, consequentemente, de reprodução, transmitindo seus traços genéticos para as próximas gerações [44]. O algoritmo busca simular esse processo ao avaliar possíveis soluções de uma função objetivo dentro de um espaço de busca predefinido e selecionar a melhor delas.

Os AGs apresentam ampla aplicabilidade em diversas áreas devido à sua capacidade de adaptação a diferentes problemas. Sua versatilidade advém da habilidade de se ajustarem a representações genéricas de soluções e de explorar o espaço de busca tanto local quanto globalmente em busca do ótimo. Na literatura, encontram-se exemplos variados de suas aplicações, abrangendo áreas como processamento de imagens e vídeos [45, 46], robótica [47] e problemas de escalonamento [48].

Neste capítulo, os Algoritmos Genéticos serão abordados de forma geral, com destaque para sua aplicação na otimização de hiperparâmetros. Em sequência, será discutida a otimização com múltiplos objetivos, seguida pela apresentação do NSGA-II, um algoritmo multiobjetivo utilizado no e-AutoMFIS2.

3.1 Processo de execução

Conforme mencionado anteriormente, os Algoritmos Genéticos constituem uma técnica de otimização fundamentada na teoria darwiniana da seleção natural, conforme destacado por [43]. A estrutura básica de um algoritmo genético é composta por cromossomos, também chamados de indivíduos, os quais representam pontos no espaço de busca. Cada cromossomo possui um valor de aptidão que indica sua qualidade em relação ao problema em questão. Com base nesses valores de aptidão, os cromossomos são submetidos a operadores inspirados em conceitos biológicos, tais como seleção, mutação e reprodução.

Descrevendo com mais detalhes a execução de um algoritmo genético de codificação binária, representado na Figura 3.1, o processo é realizado da seguinte maneira. Primeiramente, inicia-se com a geração aleatória de uma população composta por n cromossomos. Em seguida, é realizado o cálculo da aptidão para cada cromossomo, determinando a sua qualidade em relação ao problema em análise. Com os valores de aptidão definidos, os cromossomos mais promissores possuem maiores chances de serem selecionados para a etapa de reprodução. Durante essa etapa, ocorre a recombinação dos cromossomos selecionados, gerando novos descendentes. Estes novos indivíduos podem ser sujeitos a uma possível mutação, introduzindo variabilidade na população. O processo iterativo continua até que um critério de parada seja atingido ou até que uma solução próxima ao ótimo seja alcançada. Cada etapa mencionada será detalhada nos tópicos subsequentes.

3.1.1 Inicialização da População

Esta etapa é responsável pela inicialização da primeira população de soluções potenciais para o problema em questão. Essa fase inicial desempenha um papel crucial, uma vez que a qualidade e a diversidade da população inicial podem influenciar significativamente na eficácia do algoritmo.

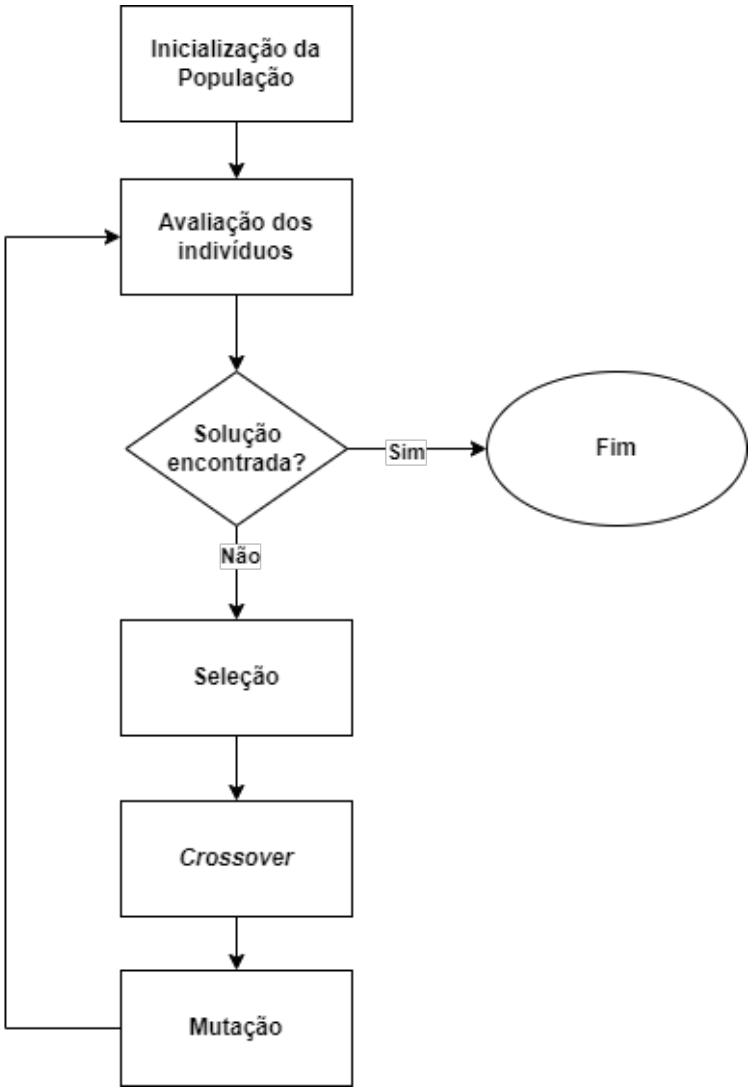

Figura 3.1: Estrutura básica de um algoritmo genético

A estratégia de inicialização da população geralmente implica na geração aleatória de cromossomos. Isso significa que cada cromossomo é criado com valores aleatórios para suas características, garantindo, ao mesmo tempo, que estejam dentro das restrições do problema. É crucial ressaltar que essa população inicial deve abranger uma ampla região do espaço de busca para explorar efetivamente as possíveis soluções.

A quantidade de indivíduos na população inicial também desempenha um papel significativo. Uma população muito pequena pode comprometer a diversidade genética e a capacidade de exploração do espaço de busca, enquanto uma população excessivamente grande pode aumentar o tempo de processamento do algoritmo e consumir recursos computacionais em demasia. Portanto, encontrar um equilíbrio entre esses extremos é fundamental para garantir um desempenho eficiente do AG.

Em problemas complexos, é possível empregar estratégias avançadas de inicialização, tais como a inclusão de heurísticas ou conhecimento prévio do problema, com o intuito de gerar uma população inicial mais direcionada ou diversificada [49, 50]. Essas abordagens podem melhorar a eficiência do algoritmo, permitindo uma exploração mais eficaz do espaço de busca desde o início do processo de otimização.

3.1.2 Cálculo de Aptidão

Nesta etapa do processo, é estabelecida a ligação direta entre o algoritmo e as características do problema que está sendo otimizado [51]. As soluções geradas pela população inicial são avaliadas por meio de uma função objetivo predefinida, alinhada com o objetivo almejado. Os resultados obtidos, representados pelos valores de *fitness*, indicam o grau de qualidade de cada indivíduo. Com base nesses valores, a população é então ordenada de acordo com sua aptidão, preparando-a para as próximas etapas.

A formulação da função de aptidão é adaptada de acordo com a natureza específica do problema. Em situações que envolvem a maximização ou minimização de uma função objetivo, a função de aptidão é projetada para atribuir valores mais elevados a soluções que se aproximam mais da solução ideal, e vice-versa. Por exemplo, em problemas de minimização, a função de aptidão pode ser concebida como o inverso da função objetivo que se deseja maximizar, refletindo assim a busca pela minimização dos valores da função objetivo.

A função de aptidão pode ser concebida para considerar uma variedade de métricas, como características específicas do problema, restrições e objetivos a serem alcançados. Uma função de aptidão bem elaborada desempenha um papel crucial no processo de seleção dos indivíduos mais adequados para reprodução, cruzamento e mutação. Essa seleção direcionada impulsiona a evolução da população de soluções ao longo das gerações, levando-a progressivamente a melhores resultados.

3.1.3 Seleção

Nesta etapa do processo, se a solução do problema ainda não foi satisfatória pela população avaliada no período anterior, os indivíduos passam por um processo de seleção, no qual os melhores avaliados têm maior probabilidade de serem escolhidos para a etapa de *crossover*. Na literatura, existem diversos métodos de seleção, sendo os mais conhecidos: seleção por roleta, rank e torneio.

A seleção por roleta, como sugere o nome, atribui a cada indivíduo uma porção de uma roleta proporcional ao seu grau de aptidão (vide Fig. 3.2). Em seguida, a roleta é "girada" e os indivíduos são selecionados para a etapa de reprodução com base nas porções atribuídas. Esse tipo de seleção, embora leve mais tempo para convergir, introduz uma maior diversidade nas próximas gerações.

A seleção por rank é uma versão modificada da seleção por roleta [43]. Neste método, os indivíduos são ordenados com base em sua aptidão, onde o pior indivíduo recebe o valor 1 e o melhor indivíduo recebe o valor igual ao número n de indivíduos na população. Esse tipo de seleção favorece os indivíduos com classificação mais alta, mas também oferece chances razoáveis para os indivíduos com menor aptidão. Essa abordagem pretende alcançar um equilíbrio entre a diversidade de soluções e o uso de soluções com alta aptidão.

Por fim, a seleção por torneio envolve a escolha aleatória de n indivíduos dentro de uma população, os quais competem entre si. O indivíduo com o maior valor de aptidão é selecionado para a próxima geração. Embora o número n de indivíduos participantes na competição seja frequentemente configurado como 2, esse valor pode variar, influenciando diretamente a convergência do algoritmo. Este método é amplamente utilizado devido à sua eficiência e facilidade de implementação [52].

	Indivíduo S_i	Aptidão $f(S_i)$	Aptidão Relativa
S_1	10110	2.23	0.14
S_2	11000	7.27	0.47
S_3	11110	1.05	0.07
S_4	01001	3.35	0.21
S_5	00110	1.69	0.11

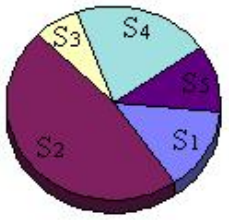

Figura 3.2: População e seus correspondentes valores na roleta de seleção

3.1.4 Reprodução

Na etapa de reprodução, conhecida como *crossover*, novos indivíduos são gerados de acordo com uma taxa percentual pre determinada através da combinação de genes de indivíduos das gerações anteriores. Os métodos mais comuns para aplicar o *crossover* incluem o *crossover* de um ponto, dois pontos e uniforme. Cada método tem sua própria maneira de combinar os genes dos pais para produzir descendentes. A Figura 3.3 fornece uma ilustração desses métodos em ação.

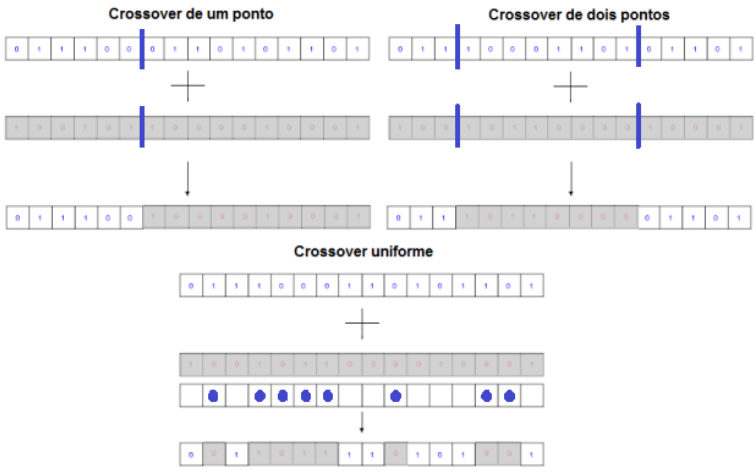

Figura 3.3: Exemplificação dos tipos de crossover [53]

No *crossover* de um ponto, um ponto de corte é escolhido aleatoriamente nos cromossomos dos pais, dividindo-os em duas partes. Os genes à direita do ponto de corte de um dos pais são trocados pelos genes à direita do ponto de corte do outro pai, criando dois novos descendentes. Este método é eficaz para trocar informações entre os pais em uma posição específica, permitindo a recombinação de características genéticas.

No *crossover* de dois pontos, dois pontos de corte são selecionados aleatoriamente nos cromossomos dos pais, dividindo-os em três partes. Os genes entre os dois pontos de corte de um dos pais são trocados pelos genes entre os mesmos pontos de corte do outro pai. Isso resulta em dois descendentes com seções genéticas misturadas dos pais. Esse método promove uma recombinação mais extensa das informações genéticas, potencialmente introduzindo maior diversidade na descendência.

No *crossover* uniforme, cada gene é escolhido individualmente para cada descendente. Uma máscara binária é gerada, onde cada bit indica qual pai contribui para o gene correspondente do descendente. Se o bit for 0, o gene é herdado do primeiro pai; se for 1, o gene é herdado do segundo pai. Esse método permite uma troca mais aleatória de informações genéticas, oferecendo uma combinação flexível de características dos pais em cada descendente.

3.1.5 Mutação

A mutação desempenha o papel de introduzir diversidade na população [54], reduzindo o risco de estagnação em ótimos locais e promovendo a exploração do espaço de busca [55]. Este operador introduz alterações aleatórias nos genes dos indivíduos com base em uma taxa de mutação predefinida, permitindo que características genéticas que podem não ter sido favorecidas durante a seleção e o *crossover* possam ressurgir na população, possibilitando sua avaliação em novos contextos populacionais [55].

Os operadores de mutação variam conforme a representação dos genes de cada indivíduo. Por exemplo, em codificações binárias, é comum empregar a mutação por bit, em que um ou mais genes são invertidos, conforme ilustrado na Figura 3.4. Em contextos onde os genes representam sequências ordenadas, como no problema do caixeiro viajante – um desafio clássico de otimização combinatória – é frequentemente utilizada a mutação por deslocamento. Nessa abordagem, uma parte dos genes é movida para uma nova posição no cromossomo.

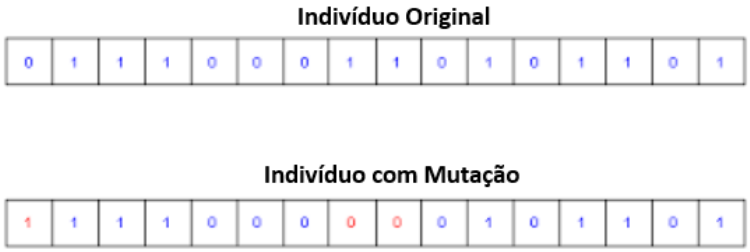

Figura 3.4: Exemplo de mutação (tipo binária) [53]

3.2 Otimização de hiperparâmetros

Os hiperparâmetros são elementos fundamentais em algoritmos de aprendizado de máquina, influenciando diretamente o desempenho dos modelos. São valores ajustáveis que não são aprendidos durante o treinamento do modelo, ao contrário dos parâmetros do modelo que são estimados a partir dos dados. Hiperparâmetros podem ter uma variedade de valores irrestritos e formam um espaço vetorial de n dimensões com valores reais [56]. Por isso a busca manual pela configuração ideal dos hiperparâmetros é inviável devido à vasta quantidade de combinações possíveis.

A dificuldade de se encontrar a configuração ótima, ou próxima a ela, muitas vezes transforma essa questão em um problema de otimização. O objetivo é obter a configuração que resultará em um desempenho ótimo ou quase ótimo do modelo, levando em consideração as restrições de cada hiperparâmetro. Nesse contexto, existem diversos métodos para auxiliar o processo de otimização, tais como a *Grid Search* [57, 58], otimização bayesiana [59, 60], e também os algoritmos genéticos [61]. Dado que o foco deste trabalho é o uso de algoritmos genéticos para lidar com esse tipo de problema, este tópico concentrará suas atenções nessa técnica, apresentando seu funcionamento, aplicações práticas, vantagens e desvantagens.

Em relação à aplicação de algoritmos genéticos (AGs) para otimização, a população é composta por cromossomos, cujo material genético representa os hiperparâmetros a serem configurados. Cada indivíduo constitui uma configuração específica a ser avaliada no espaço de busca. O cálculo da aptidão ocorre durante o processo de treinamento e teste do modelo a ser otimizado, e a seleção é realizada com base em uma ou mais métricas de avaliação escolhidas para o modelo. Os demais processos seguem conforme explicado nos tópicos anteriores.

Os AGs apresentam características distintas em comparação com outros métodos de otimização, com vantagens notáveis. Entre elas, destaca-se a inicialização aleatória da população, eliminando a necessidade de escolha de valores iniciais [62]. Além disso, os AGs são capazes de lidar com espaços de busca complexos devido à sua ampla e diversificada busca. Eles também se adaptam à resposta do modelo utilizado, explorando o espaço de busca de forma eficiente para convergir aos melhores resultados.

No entanto, algumas desvantagens devem ser consideradas. Os AGs executam suas etapas principais (inicialização de população, cálculo de aptidão, seleção, reprodução e mutação) de forma sequencial, o que pode resultar em uma convergência mais lenta [62]. No entanto, é importante destacar que o processo de avaliação dos indivíduos pode ser paralelizado para acelerar a convergência. Além disso, os AGs podem sofrer estagnação em ótimos locais, especialmente em problemas de grande complexidade. Outro aspecto crítico é a dependência dos próprios hiperparâmetros do AG, como o tamanho da população e a taxa de mutação, o que pode se configurar como um problema de otimização adicional.

3.3 Otimização multiobjetiva

Até agora, todas as discussões sobre otimização foram baseadas na premissa de que o problema possui apenas um objetivo a ser otimizado. No entanto, na prática, muitos problemas enfrentados apresentam uma natureza multiobjetiva, onde há a presença de diversos objetivos conflitantes. Por exemplo, problemas do mundo real discutidos em estudos como [63, 64] frequentemente envolvem a otimização de múltiplos critérios, tornando-os consideravelmente mais complexos. Nesses cenários, o desafio reside em encontrar soluções que equilibrem de maneira eficaz todos os objetivos relevantes. Esta seção tem como objetivo introduzir definições e conceitos fundamentais que são essenciais para abordar problemas de otimização multiobjetiva.

Matematicamente, um problema de otimização multiobjetiva pode ser formalizado pela Equação 3-1, onde k representa o número de objetivos e n denota o número de variáveis que compõem o vetor $x = x_1, ..., x_n$ no espaço de busca X. Esse espaço de busca é definido por diversas restrições, que podem ser lineares ou não-lineares, sobre as variáveis. O objetivo primordial é encontrar um vetor $\mathbf{x} \in X$ que minimize todos os k objetivos $f(x) = f_1(x), ..., f_k(x)$, representando uma solução que seja ótima em relação a todos os critérios estabelecidos.

$$\begin{aligned} \min f(x) &= \{f_1(x), \ldots, f_k(x)\} \\ \text{s.a. } \mathbf{x} &\in X \end{aligned} \tag{3-1}$$

Entretanto, no contexto de otimização multiobjetiva, o processo de minimização não é tão direto quanto em problemas de otimização com um único objetivo, onde os melhores resultados são determinados pela relação "menor ou igual"entre os valores da função objetivo [65]. Para problemas multiobjetivo, essa abordagem não é viável devido à impossibilidade de se estabelecer uma ordem no espaço de funções objetivas $\mathbb{R}^k$. Por isso, são utilizadas definições fracas para permitir a comparação entre vetores em $\mathbb{R}^k$. Estas definições incluem:

- **Definição 1:** Dominância de Pareto - Princípio fundamental em otimização multiobjetiva, onde se consideram duas soluções y e x. Neste conceito, é dito que y domina x se pelo menos um dos objetivos que a compõem é melhor, enquanto os demais não são piores (conforme exemplificado na Eq. 3-2).

$$\begin{cases} \forall_{i\in\{1,\ldots,k\}} : f_i(y) \leq f_i(x) \\ \exists_{i\in\{1,\ldots,k\}} : f_i(y) < f_i(x) \end{cases} \tag{3-2}$$

– **Definição 2:** Pareto-Ótimo - Uma solução é considerada Pareto-ótima, conforme definido na Eq. 3-3, quando não existe outra solução no espaço de busca X que a domine.

$$\begin{aligned} \nexists\ y \in X \text{ tal que } f_i(y) \leq f_i(x^*) \text{ para todo } i = 1, ..., k \\ \text{e } f_i(y) < f_i(x^*) \text{ para algum } j = 1, ..., k \end{aligned} \tag{3-3}$$

– **Definição 3:** Ótimo Fraco de Pareto - Uma solução y é considerada um ótimo fraco de Pareto quando nenhum de seus objetivos é inferior aos demais. Todas as soluções ótimas de Pareto no espaço de busca X são consideradas ótimos fracos, embora o contrário não seja necessariamente verdadeiro. Para ser classificado como um ótimo de Pareto, é necessário que pelo menos um objetivo seja superior aos das outras soluções.

– **Definição 4:** Conjunto Pareto-Ótimo - Em um espaço de busca X, pode haver k soluções consideradas Pareto-Ótimas, como ilustrado na Figura 3.5. O conjunto formado por essas soluções é denominado conjunto Pareto-Ótimo.

– **Definição 5:** Fronteira de Pareto - Refere-se à coleção de soluções Pareto-ótimas no espaço de busca das funções objetivas, derivadas do espaço de soluções X. Na Figura 3.6, a fronteira de Pareto é visualizada como uma espécie de ”barreira”, onde nenhum dos objetivos pode ser melhorado sem prejudicar os demais, destacando a noção de dominância em problemas multiobjetivos.

Formada a fronteira de Pareto, a seleção da melhor solução torna-se uma questão de extrema importância. Como a fronteira é composta pelas soluções não-dominadas identificadas durante o processo de otimização, a escolha de uma única solução é uma tarefa complexa. A literatura apresenta diversas metodologias para esse propósito, desde métodos mais simples, como o uso da distância euclidiana a um ponto ideal [66], até abordagens mais complexas, como o uso de *clusters* hierárquicos para a avaliação de soluções [67].

As definições acima são as bases para diversos algoritmos de otimização multiobjetiva como *Normal Boundary Intersection* (NBI) [68], algoritmo de colônia de formigas multiobjetiva [69] e algoritmo de enxame de partículas multiobjetiva [70] e GA. Dado que o foco deste capítulo é a apresentação específica dos Algoritmos Genéticos, o próximo tópico oferecerá uma breve descrição desse algoritmo no contexto da otimização multiobjetiva.

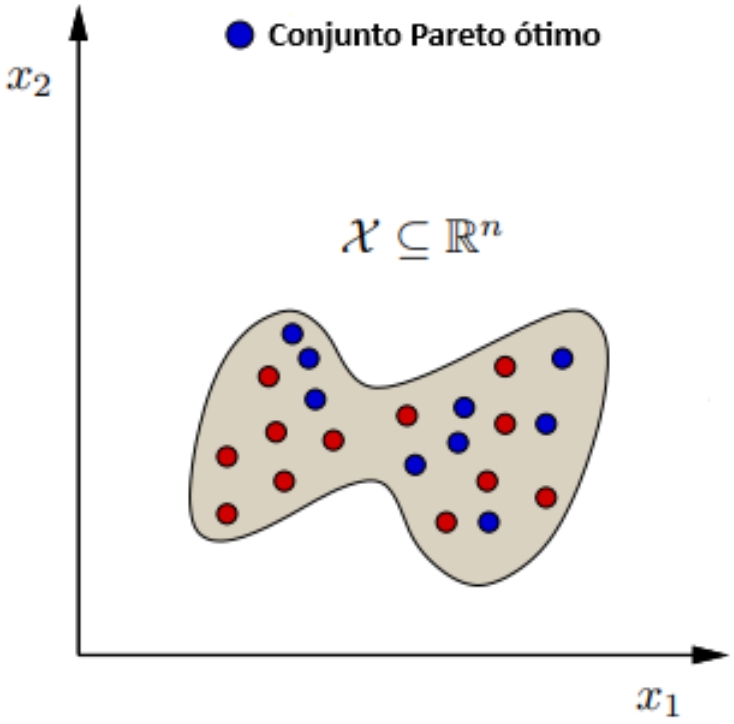

Figura 3.5: Exemplo de um espaço de busca X com seu conjunto de pareto-ótimos (em azul) [65]

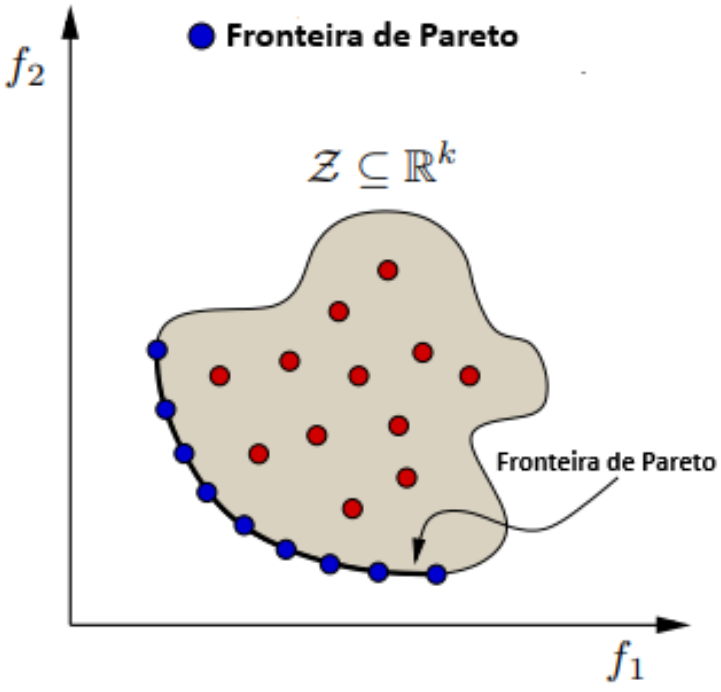

Figura 3.6: Fronteira de Pareto [65]

3.3.1
Algoritmos Genéticos para otimização multiobjetiva

Os Algoritmos Genéticos multiobjetivos compartilham muitas semelhanças com suas versões mais simples. A principal distinção reside na abordagem da função de aptidão, na qual a versão multiobjetiva busca gerar a fronteira de Pareto-ótima de tal forma que não seja possível realizar melhorias em qualquer objetivo sem prejudicar os demais [71]. Convergência, diversidade e cobertura são os principais objetivos desta variante do AG [43].

É importante destacar que o AG é altamente aplicável a problemas multiobjetivos, pois lida eficientemente com várias soluções simultaneamente. Isso possibilita a identificação de conjuntos de Pareto-ótimos em uma única população, além de sua capacidade de explorar globalmente o espaço de busca, o que é crucial para encontrar soluções diversificadas em problemas desafiadores com espaços de solução não convexas, descontínuas e multimodais [72]. Além disso, os operadores de *crossover* e mutação permitem a exploração de novas populações e, consequentemente, a descoberta de novos conjuntos de Pareto-ótimos em regiões não exploradas da fronteira de Pareto.

Existem diversas variantes de algoritmos genéticos multiobjetivos. A primeira versão foi proposta em [73], que introduziu o conceito de nicho e a tomada de decisão para tratar problemas multiobjetivos. Outro exemplo é o *Niched Pareto Genetic Algorithm* (NPGA) [74], que emprega técnicas de seleção por torneio e dominância de Pareto. Entre os algoritmos genéticos multiobjetivos, destacam-se também o *Weighted-Based Genetic Algorithm* (WBGA) [75], o *Random Weighted Genetic Algorithm* (RWGA) [76], e o *Non-dominated Sorting Genetic Algorithm* (NSGA) [77], que serviu de base para o desenvolvimento do *Fast Nondominated Sorting Genetic Algorithm* (NSGA-II). Este último é o algoritmo utilizado na etapa de experimentação deste trabalho. Portanto, a próxima seção deste capítulo é dedicada à apresentação detalhada do NSGA-II.

3.4 NSGA-II

O NSGA-II, proposto em [78], representa uma evolução do algoritmo NSGA original, visando superar suas limitações estruturais identificadas. Estas incluem alta complexidade computacional, falta de elitismo e a necessidade de configurar um parâmetro de compartilhamento para preservar a diversidade do algoritmo. Estas deficiências serviram como base para a concepção do NSGA-II, que introduziu uma metodologia para reduzir o tempo de ordenação das fronteiras não dominadas, implementou a técnica de elitismo e eliminou a dependência do parâmetro de compartilhamento ao introduzir um novo operador conhecido como *crowding-distance* ou distância de multidão. Os próximos tópicos desta seção têm como objetivo apresentar esses pontos principais e explicar como o NSGA-II opera durante a sua execução.

3.4.1 Princípios fundamentais do NSGA-II

Como mencionado anteriormente, o NSGA-II apresenta algumas características distintivas em sua estrutura básica. Estas incluem a ordenação das fronteiras não dominadas, a preservação de soluções por meio de elitismo, e a utilização de operadores de distância de multidão e seleção. A seguir, examinam-se esses elementos para compreender como eles contribuem para o desempenho do algoritmo.

No NSGA-II, a ordenação das fronteiras não-dominadas é otimizada em relação ao seu antecessor. Este processo baseia-se na contabilização de dois valores: a contagem de dominância n_p para uma solução de referência p, que indica o número de soluções que dominam p, e o conjunto de soluções dominadas S_p pela mesma referência.Inicialmente, as soluções com $n_p = 0$ formam a primeira fronteira de soluções não dominadas. Em seguida, cada solução em S_p é visitada e seu valor n_p é reduzido em 1. Se a contagem de dominância de uma solução se tornar zero durante esse processo, ela é movida para o conjunto Q, que constitui a segunda fronteira de soluções não dominadas. Os conjuntos S_p de cada solução em Q passam pelo mesmo procedimento até que todas as fronteiras sejam determinadas. Essa abordagem eficiente permite uma ordenação rápida e precisa das soluções com base em sua dominância de Pareto.

Outra diferença fundamental em relação à versão anterior do NSGA II é a inclusão do elitismo. Ao contrário de seu antecessor, este algoritmo incorpora uma estratégia que preserva as soluções não dominadas para as próximas gerações. Essa abordagem garante que as soluções de alta qualidade sejam mantidas ao longo das iterações, impedindo a perda precoce de soluções promissoras.

Mais um aspecto distintivo do NSGA-II é a utilização do operador de distância de multidão (*crowding-distance*), que desempenha um papel crucial ao eliminar a necessidade de configurar o parâmetro de compartilhamento presente em sua versão anterior. No NSGA original, esse parâmetro era crucial para controlar a influência de soluções próximas. No entanto, a distância de multidão calcula automaticamente o "espaço vazio" ao redor de cada solução, promovendo a diversidade sem a necessidade de ajustes manuais.

Matematicamente, o cálculo da distância de multidão é representado pela Eq. 3-4, onde f^ij é o valor do $j-$ésimo objetivo da $i-$ésima solução, $f^{max}j$ e f_j^{min} são, respectivamente, os valores máximos e mínimos do $j-$ésimo objetivo entre todos os indivíduos da população. Nesse contexto, preferem-se soluções com uma maior distância de multidão, pois isso indica que a solução está localizada em uma região menos densa, favorecendo a manutenção da diversidade do algoritmo.

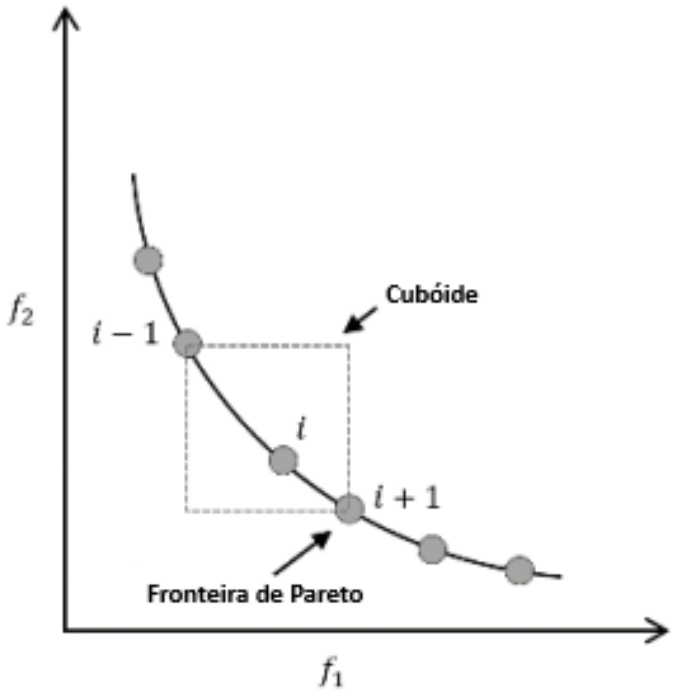

Figura 3.7: Exemplificação do cálculo de *crowding distance* [79]

$$cd(i) = \sum_{i=1}^{k} \frac{f_j^{i+1} - f_j^{i-1}}{f_j^{max} - f_j^{min}} \tag{3-4}$$

O último pilar do NSGA-II é o operador de seleção, baseado em seleção de torneio utilizando a distância de multidão como referência. Neste método, a ordem dos membros da população e suas respectivas distâncias de multidão são consideradas. A seleção entre duas soluções é realizada seguindo duas condições: se as soluções têm ordens diferentes, a solução de maior ordem é selecionada; se as soluções têm a mesma ordem, a escolha é feita com base na maior distância de multidão, favorecendo a diversidade na próxima geração.

3.4.2 Operação do NSGA-II

O processo de execução do NSGA-II se inicia com a geração da primeira população P_t com tamanho N. Esta população inicial é então ordenada com base na metodologia de não-dominância discutida anteriormente, atribuindo a cada solução um valor de aptidão correspondente ao seu nível de não-dominância. Em seguida, os operadores genéticos são aplicados para gerar uma nova população Q_t, também de tamanho N. Posteriormente, essas duas populações são combinadas para formar uma nova população R_t de tamanho $2N$. Para reduzir esta nova população ao tamanho predefinido N, as soluções devem ser selecionadas com base na hierarquia das fronteiras não-dominadas. No entanto, em relação à quantidade de soluções S na primeira fronteira, existem três possibilidades:

- $S > N$: as N soluções da área menos povoadas da fronteira são utilizados para formar a nova população P_{t+1}.
- $S = N$: todas as soluções desta fronteira são levados para formar a nova população P_{t+1}.
- $S < N$: todas as soluções da primeira fronteira são levadas para próxima população e, para completar a quantidade N necessária, as soluções das fronteira subsequentes são utilizadas, sempre respeitando a ordem das fronteiras de não-dominância.

Após formada a população P_{t+1}, o processo descrito acima é repetido para a criação de novas gerações, até que o critério de parada (e.g. Número máximo de gerações, convergência de soluções) seja atendido. A Figura 3.8 ilustra todo o processo apresentado.

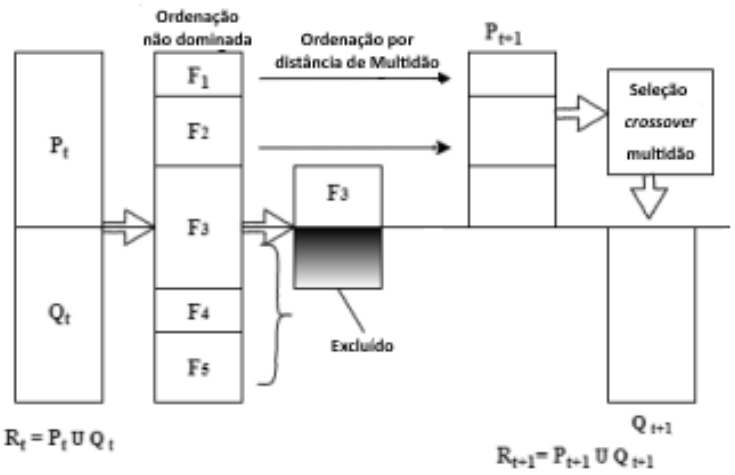

Figura 3.8: Exemplificação do processo de execução do algoritmo NSGA-II [79]

4
Sistemas de Inferência *Fuzzy* aplicados a séries temporais

A capacidade de prever eventos ou fenômenos futuros serve como ferramenta de apoio para o planejamento e a tomada de decisões em diversas áreas do conhecimento, como Engenharia, Medicina e Economia. Nesse contexto, a análise de séries temporais desempenha um papel crucial, e na literatura são apresentadas diversas técnicas, desde abordagens estatísticas até técnicas de *soft computing*. Entre estas, destaca-se a baseada em Lógica *Fuzzy*.

As técnicas baseadas em Lógica *Fuzzy* são extremamente úteis devido à sua capacidade de modelar problemas complexos em funções contínuas, lineares ou não-lineares. Isso têm impulsionado o uso de Lógica Fuzzy na previsão de séries temporais, pois seus modelos são simples, computacionalmente eficientes e facilmente auditáveis. Além disso, diferentemente de modelos estatísticos e de *(*Deep Learning), não exigem uma grande quantidade de dados para operar de forma eficiente [86].

Outra característica importante das técnicas baseadas em Lógica *Fuzzy* é a interpretabilidade, um atributo fundamental para compreender como o modelo realiza as previsões. Essa interpretabilidade advém da modelagem de situações complexas por meio de sentenças expressas em linguagem natural. Ou seja, as variáveis são linguísticas e o modelo faz uso de regras no formato "se-então" Essa representação simplificada facilita significativamente o processo de análise e interpretação por parte dos usuários.

Vários estudos exploram a aplicação da Lógica *Fuzzy* para lidar com questões complexas, tais como a compensação de zona morta em sistemas de controle de movimentação [80], classificação de imagens [81], previsão de cargas para gerenciamento de produção de energia [82] e previsão da probabilidade de câncer de próstata [83]. Isso evidencia a sua versatilidade e aplicabilidade.

Este capítulo se concentra na aplicação da Lógica *Fuzzy* na previsão de séries temporais, abordando desde a transformação dos dados até a etapa de previsão. Em seguida, será abordada a questão da interpretabilidade, um dos principais diferenciais dessa técnica, explorando os principais aspectos que tornam tais modelos explicáveis. Por fim, serão apresentados dois modelos essenciais para a construção do modelo proposto neste trabalho: o AutoMFIS [9] e o e-AutoMFIS [10].

4.1 SIF para previsão de séries temporais

Técnicas baseadas na Lógica *Fuzzy* são atraentes devido à sua capacidade de lidar com problemas por meio de termos linguísticos. Essa abordagem é mantida quando se trata da previsão de séries temporais, onde as séries são tratadas como uma coleção de termos linguísticos indexados no tempo [84].

Ao abordar a aplicação da Lógica *Fuzzy* na previsão de séries temporais, é essencial estabelecer o conceito de séries temporais *Fuzzy* (STF). Segundo a definição de [85], uma STF é uma série $Y(t) \in \mathbb{R}$, onde $(t = 0, 1, ..., T)$, com um universo de discurso U delimitado pelos valores máximo e mínimo de $Y(t)$. A partir de U, é formada uma função $F(t)$ de k conjuntos *Fuzzy* $f_i(t)$, para $i = 1, ..., k$. Assim, pode-se definir $F(t)$ como uma série temporal *Fuzzy* definida em $Y(t)$ para $t = 0, 1, ..., T$.

As STFs podem ser classificadas tanto em relação à variância no tempo quanto em relação à sua ordem. Em relação ao tempo, há duas classificações possíveis: invariante ou variante no tempo. No primeiro caso, a relação *Fuzzy* entre $F(t-1)$ e $F(t)$ é independente de t; caso contrário, a STF é considerada variante no tempo. Em relação à ordem, dois grupos são formados: os de primeira ordem ou de ordem superior. Uma STF é de primeira ordem quando a previsão $F(t)$ é dependente apenas de $F(t-1)$; em casos de ordem superior, $F(t)$ é dependente de $F(t-n), ..., F(t-2), F(t-1)$, onde n representa a ordem da STF.

A estrutura básica desses tipos de modelos está representada na Figura 4.1. Essa estrutura compreende os seguintes passos: definição e partição do Universo de Discurso, fuzzificação dos dados, extração de conhecimento e, se necessário, defuzzificação. Cada um desses elementos será detalhado a seguir.

4.1.1 Definição do universo de discurso e partição

As etapas de definição e partição do universo de discurso estão interligadas, como mostrado na Figura 4.2, pois ambas fornecem a base para a representação e manipulação das variáveis linguísticas. A definição do universo de discurso determina os limites e alcances dos valores considerados, enquanto a partição estabelece os intervalos linguísticos, essenciais para traduzir conceitos *Fuzzy* em termos claros e operacionais.

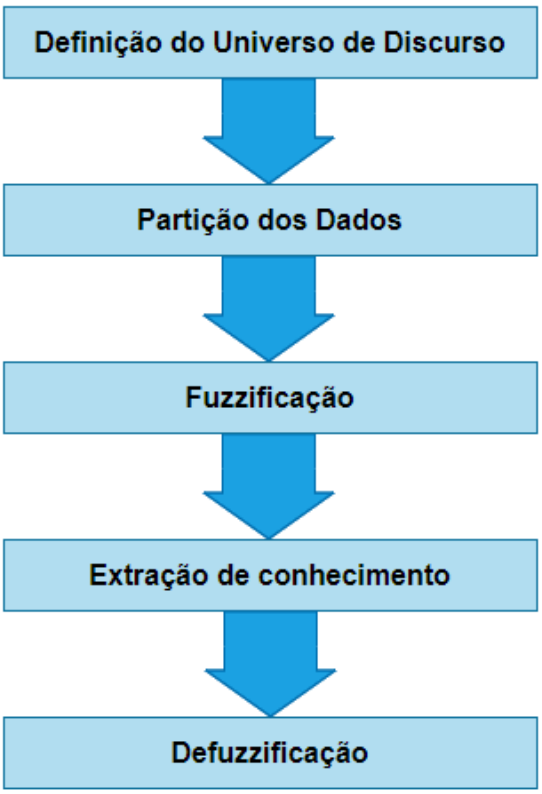

Figura 4.1: Estrutura básica de um modelo baseado na Lógica *Fuzzy*

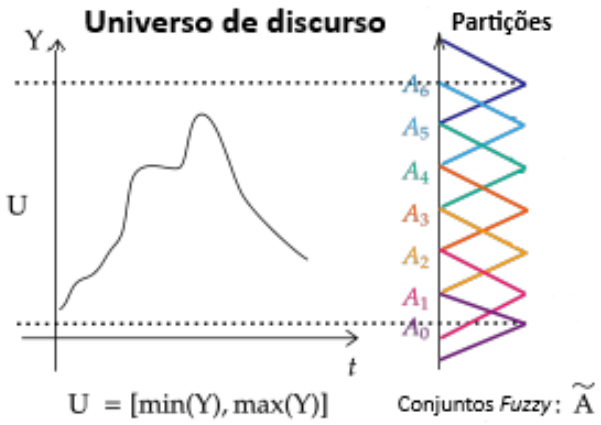

Figura 4.2: Definição do universo de discurso e sua partição [84]

O primeiro passo é a definição do universo de discurso U a ser utilizado. Embora a definição mais comum seja $U = [min(Y), max(Y)]$ para séries temporais Y, também é adotado o uso de valores de confiança D_1 e D_2 quando esses limites são excedidos. D_1 e D_2 são geralmente valores percentuais dos extremos da série Y, resultando em $U = [Min(Y) - D_1, Max(Y) + D_2]$, onde $D_1 = rMin(Y)$ e $D_1 = rMin(Y)$, com r como um termo multiplicativo entre 0 e 1, determinado com base no contexto do problema [109].

Na etapa de partição, são criados grupos sobrepostos, cada um atribuído a um termo linguístico. Estes grupos são definidos por três parâmetros principais: número de partições, técnica de partição e função de pertinência (μ).

Em relação ao número de partições, não há uma relação linear entre a quantidade de grupos e a acurácia do modelo [86], tornando a escolha da quantidade ideal uma análise caso a caso. Uma grande quantidade de partições (conjuntos *Fuzzy*) pode gerar modelos excessivamente complexos, dificultando a explicabilidade, enquanto poucos grupos podem resultar em modelos excessivamente simples e, portanto, incapazes de capturar padrões e relações nos dados.

Após definidas as quantidade de grupos, a divisão dos dados é realizada por meio de dois métodos de separação: uniforme e não uniforme. Conforme apresentado na Figura 4.3, as partições do tipo não uniforme podem ser baseadas em métodos matemáticos ou de *soft computing*.

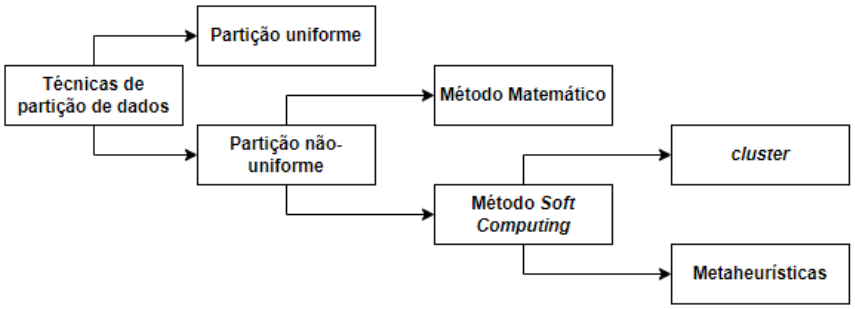

Figura 4.3: Classificação dos métodos de partição

No caso das partições uniformes, todos os grupos possuem tamanhos iguais, sendo determinados pelo número de grupos e pela extensão do universo de discurso. Esses grupos *Fuzzy* são sobrepostos, como exemplificado na Figura 4.4. Esse método é utilizado devido à sua simplicidade, porém, em casos em que os dados não são uniformes, geralmente apresenta baixo desempenho [87].

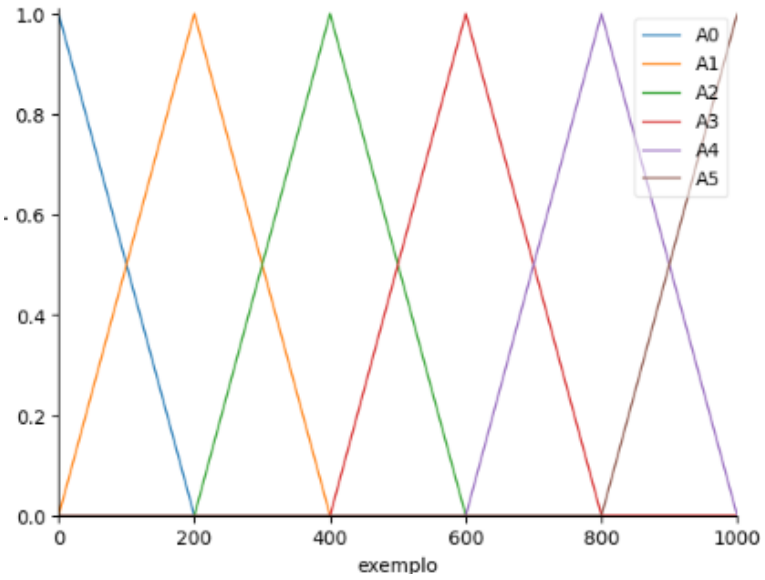

Figura 4.4: Partição uniforme do universo de discurso

Os métodos de partição não uniforme são mais complexos, podendo ser baseados em métodos matemáticos – como o uso de percentis (método Tukey) ou de partições baseadas em distribuição e média [88] – ou em técnicas de *soft computing* como clusterização, que procuram otimizar os grupos usando a distribuição dos dados como referência. Outra opção são as técnicas meta-heurísticas, como o algoritmo de enxame de partículas [90] ou algoritmos genéticos [89]. Embora mais lentos, esses métodos tendem a oferecer melhores resultados de acurácia [84].

Por fim, na última etapa da partição dos dados, são definidos as funções de pertinência, que irão representar o grau de relação, entre $[0, 1]$, dos valores reais (*crisp*) e seus respectivos grupos *Fuzzy*. Esses grupos podem ter diferentes formatos: triangulares, trapezoidais ou gaussianos. Embora não influenciem significativamente na acurácia dos modelos, afetam diretamente sua explicabilidade [84].

4.1.2 Fuzzificação

Fuzzificação é o processo de compatibilização de valores reais com valores *Fuzzy*: um valor preciso $Y(t)$ é mapeado para os domínios de conjuntos *Fuzzy*, representado por $f(t) = \{\mu_{A_0}(Y(t)), ..., \mu_{A_n}(Y(t)), ..., \mu_{A_{k-1}}(Y(t))\}$, onde μ_{A_n} é o grau de pertinência do valor $Y(t)$ ao n-ésimo conjunto *Fuzzy* A_n.

Um exemplo prático desse processo pode ser observado na Figura 4.5, onde cada valor pode pertencer a um ou mais grupos com diferentes valores de pertinência. Por exemplo, o valor 0.125 de uma variável é compatível com os conjuntos (*baixo* e *médio*), com graus de pertinência 0.75 e 0.25, respectivamente.

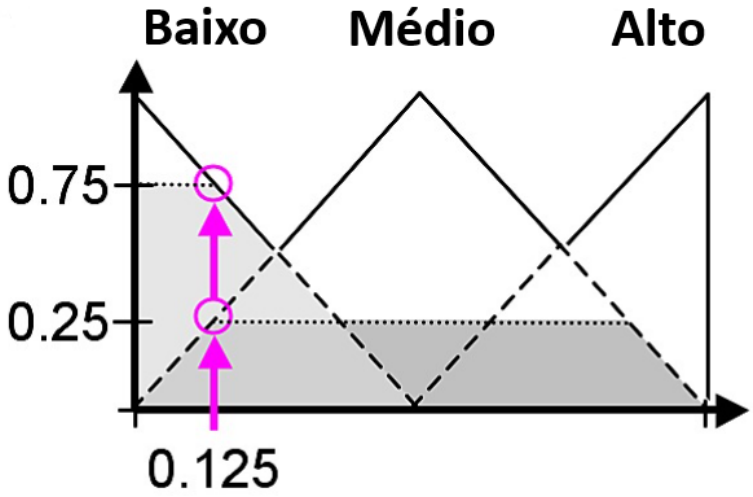

Figura 4.5: Exemplo do processo de fuzzificação [110]

4.1.3 Extração de Conhecimento

A extração de conhecimentos em Séries Temporais *Fuzzy* (STF) baseia-se na análise de dados históricos, relacionando informações passadas com dados presentes para a previsão. Diversos métodos são empregados para extrair essas informações, incluindo modelos baseados em regras (*Fuzzy*), redes neurais, métodos estatísticos e abordagens híbridas [91, 92, 93, 95, 94].

No contexto dos modelos baseados na Lógica *Fuzzy*, o conhecimento é representado por meio de uma base de regras com sentenças do tipo ”se-então”, que estabelecem uma relação entre os dados de entrada (antecedente) e a saída correspondente (consequente).

As regras (RLF) em STFs são geralmente expressas no formato $A_i \rightarrow A_j$, onde A_i é o valor fuzzificado no instante $t-1$, atuando como antecedente da regra, e A_j, representando o valor fuzzificado no instante t, é o consequente. Essas regras podem ser organizadas de duas maneiras: matricial [96, 97]ou por agrupamento por relação lógica (*Fuzzy*) (ARLF) [98].

No método matricial, as regras são organizadas em uma matriz $R_{ij} = (A_i^T) \cdot A_j$, onde R_{ij} representa a relação Lógica *Fuzzy* entre os conjuntos A_i e A_j. As inferências são realizadas utilizando a operação max-min ($\circ$), como $f(t) = f(t-1) \circ R(t, t-1)$, em que $R(t, t-1)$ é a matriz operacional formada por todas as regras para todos os grupos *Fuzzy*.

O método ARLF agrupa regras com antecedentes iguais e consequentes diferentes. Por exemplo, as regras de formato $A_1 \rightarrow A_2$, $A_1 \rightarrow A_3$ e $A_1 \rightarrow A_4$, são agrupadas como $A_1 \rightarrow A_2, A_3, A_4$ no ARLF. Este método aumenta a interpretabilidade e, geralmente, apresenta resultados superiores em termos de acurácia em comparação com o método matricial [84].

Na técnica de modelos com regras ponderadas, apresentada em [99], pesos são atribuídos às regras recorrentes e sua ordem cronológica. Isso permite a consideração de diferentes padrões nos dados, como componentes cíclicas ou sazonais, melhorando a capacidade de previsão, mesmo em modelos de ordem alta, como previsão com múltiplos passos à frente.

A escolha da técnica de extração de conhecimento é crucial, pois influencia tanto a precisão dos resultados previstos quanto a interpretabilidade do modelo gerado.

4.1.4 Defuzzificação

A defuzzificação é a etapa final do processo de previsão em modelos baseados em *Fuzzy*. Nesta fase, os consequentes das regras ativadas na etapa anterior são convertidos de volta para valores precisos, ou seja, o valor fuzzificado $f(t+1)$ é trazido de volta para o domínio real, representado pelo valor $\hat{y}(t+1)$.

Em [96, 97], foi proposta uma metodologia de defuzzificação dividida em três possibilidades distintas para garantir a cobertura máxima dos casos possíveis. No primeiro caso, quando há apenas um valor máximo de pertinência, o valor previsto é igual ao ponto médio do conjunto *Fuzzy* com a máxima pertinência. No segundo caso, quando há mais de um valor máximo de pertinência, o resultado da previsão é a média dos pontos médios dos conjuntos resultantes. Por fim, no terceiro caso, onde não há valores máximos de pertinência, o valor previsto é atribuído pela média dos pontos médios de cada conjunto.

É importante ressaltar que existem outras abordagens para a defuzzificação, como o método do centróide, amplamente utilizado na literatura, e o método da altura. Essas técnicas serão empregadas no modelo proposto e apresentadas posteriormente neste trabalho.

4.2 Interpretabilidade

Como mencionado anteriormente, os modelos baseados em *Fuzzy* possuem a capacidade de traduzir relações complexas em conceitos simples e facilmente compreensíveis, como termos linguísticos e regras de formato simples, tornando-os modelos considerados auditáveis. Para situações mais simples, a configuração desses modelos é geralmente realizada manualmente, com base no conhecimento de especialistas.

No entanto, para casos mais complexos, como os abordados neste trabalho, o processo é frequentemente automatizado, orientado pelos dados do problema em análise. Vale ressaltar também que essas técnicas de automatização, em sua maioria, priorizam os resultados relacionados à precisão, o que pode gerar modelos excessivamente complexos para serem compreendidos, perdendo assim a característica de interpretabilidade.

Há métodos para avaliar o quão auditável um modelo pode ser, por meio da avaliação de sua interpretabilidade. No entanto, por ser um tema amplo, não é fácil quantificá-lo. Assim, na literatura, são apresentados critérios e restrições que têm como objetivo controlar e orientar o desenvolvimento de modelos de forma a manter a característica de explicabilidade.

A Figura 4.6 mostra os diferentes níveis de um modelo baseado em *Fuzzy* e como eles podem ser avaliados em relação à sua interpretabilidade. Começando, no primeiro nível, pelos critérios e restrições dos conjuntos *Fuzzy*, existem três possíveis avaliações:

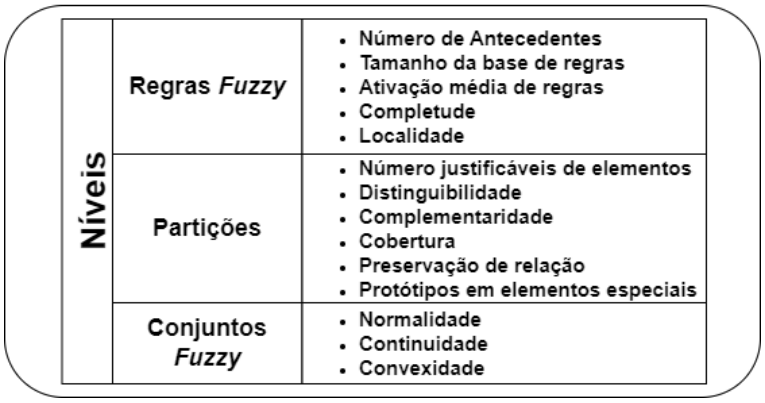

Figura 4.6: Níveis de um modelo baseados na Lógica *Fuzzy* e suas respectivas restrições

– **Normalidade**: Garante que pelo menos um elemento do universo de discurso seja totalmente representado por um conjunto específico, ou seja, seu grau de pertinência a esse conjunto deve ser igual a 1.

- **Continuidade**: Exige que os conjuntos sejam contínuos no universo de discurso, o que é crucial dada a natureza contínua da maioria dos eventos representados [100].
- **Convexidade**: Avalia se o grau de pertinência em um ponto médio entre três pontos possíveis no universo de discurso é maior ou igual ao grau de pertinência mínimo entre os outros dois pontos [101], garantindo coerência nas relações de pertinência.

O segundo nível de restrições avalia as partições do universo de discurso, que definem a semântica de uma variável linguística [102]. Para esses casos, existem seis restrições que devem ser consideradas:

- **Número justificável de elementos**: Avalia o número de conjuntos *Fuzzy* em um universo de discurso, considerando a limitação da interpretabilidade humana.
- **Distinguibilidade**: Avalia como os conjuntos *Fuzzy* que representam conceitos distintos se relacionam, em termos de sobreposição e delimitação.
- **Complementaridade**: Garante que a soma dos graus de pertinência para qualquer valor dentro do universo de discurso seja próxima ou igual a 1.

  Cobertura: Assegura que todos os elementos no universo de discurso sejam relacionáveis com pelo menos um conjunto *Fuzzy*, garantindo cobertura total do universo de discurso.

  Preservação da relação: Verifica se os conjuntos *Fuzzy* respeitam a semântica estabelecida, evitando sobreposições inadequadas. Na figura 4.7 apresenta um exemplo, onde é exposto que o conjunto maior se sobrepõe ao menor ferindo a ordem implícita proposta.
- **Protótipos em elementos especiais**: Avalia se valores específicos no universo de discurso são representados por conjuntos *Fuzzy* correspondentes.

O último nível de restrições refere-se às regras *Fuzzy*, que armazenam o conhecimento nos modelos de inferência *Fuzzy*. Para garantir a explicabilidade dessas regras, são consideradas cinco restrições:

- **Número de antecedentes**: Indica a média do número de antecedentes em todas as regras da base, preferencialmente mantido pequeno para facilitar a interpretação.

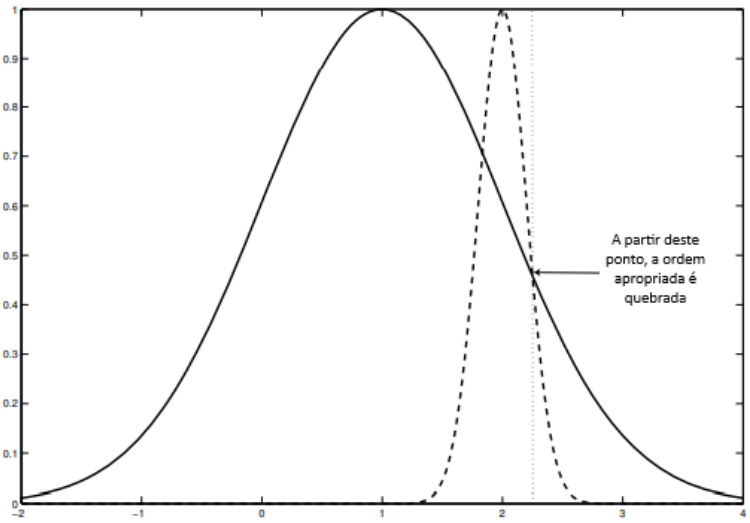

Figura 4.7: Exemplo de quando a preservação da relação não é atendida [102]

- **Tamanho da base de regras**: Quantifica o número de regras na base, sendo preferível uma base pequena para facilitar a análise.
- **Ativação média de regras**: Indica quantas regras são ativadas quando um valor de entrada é apresentado, buscando minimizar o número de regras ativadas para facilitar a inferência.
- **Completude**: Garante que a inferência para todos os valores do universo de discurso seja maior do que um limiar estabelecido. Esta restrição é usada devido a dificuldade de se justificar a ativação das regras para entradas cujo o grau de ativação é muito pequeno [102].
- **Localidade**: Dita que cada regra deve definir uma região dentro do universo de discurso, com sobreposições desejáveis para permitir uma transição suave entre as regiões e seus respectivos conceitos [102].

Para obter um modelo verdadeiramente interpretável, é importante que todas as restrições sejam atendidas. No entanto, alcançar esse objetivo pode ser desafiador, pois algumas restrições são subjetivas. Por exemplo, o tamanho da base de regras deve ser equilibrado entre a simplificação e a representação adequada do problema. Além disso, é importante considerar o compromisso entre acurácia e interpretabilidade. A otimização desses dois objetivos muitas vezes ocorre em duas etapas, onde inicialmente o modelo é desenvolvido e otimizado para acurácia e, em seguida, são aplicadas técnicas de filtragem para simplificar o modelo sem comprometer significativamente a acurácia. Este princípio é aplicado aos modelos que inspiraram este trabalho, o AutoMFIS e o e-AutoMFIS.

4.3 Modelo AutoMFIS

Como mencionado anteriormente, os modelos de inferência *Fuzzy* podem ser desenvolvidos manualmente com base no conhecimento de especialistas, mas isso apresenta limitações, especialmente em problemas complexos. Para esses casos, existem métodos que geram Sistemas de Inferência *Fuzzy* automaticamente. Esses métodos incluem técnicas meta-heurísticas, como algoritmos genéticos [103], inteligência de enxame [104], ou técnicas auto-adaptativas *Fuzzy* [105], entre outros.

Esta seção se dedica à apresentação do *Automatic Multivariate Fuzzy Inference System* (AutoMFIS), um modelo para previsão multivariada de séries temporais que combina acurácia e interpretabilidade. O AutoMFIS, em conjunto com o próximo modelo a ser apresentado, serviu como referência para o desenvolvimento deste trabalho.

O AutoMFIS, proposto em [9], oferece uma abordagem alternativa para a previsão de séries multivariadas, considerando tanto a acurácia quanto a interpretabilidade em sua arquitetura. Sua estrutura, mostrada na Figura 4.8, pode ser dividida em três partes: Definição de dicionário linguístico, Geração da base de regras e Geração dos valores *crisp* de saída.

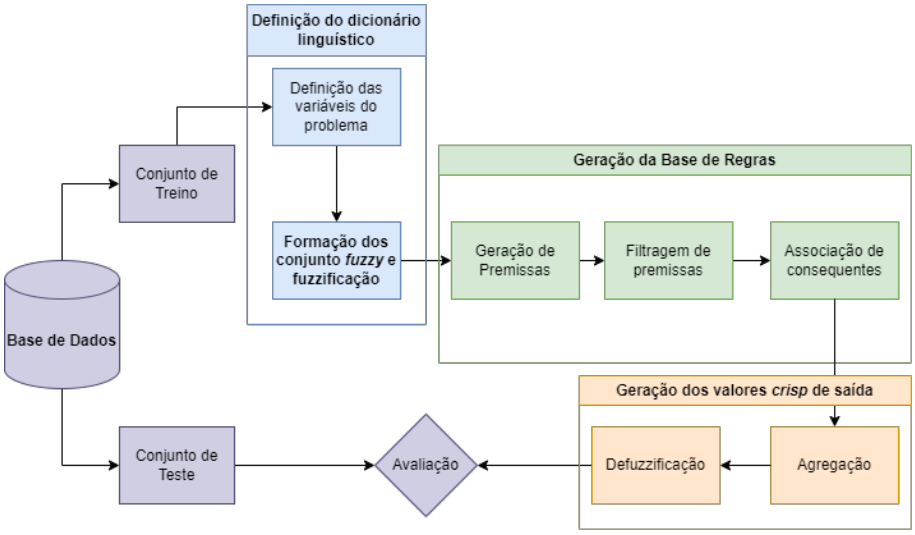

Figura 4.8: Estrutura AutoMFIS

A etapa de definição do dicionário linguístico forma a base para a fuzzificação dos dados. Ela busca definir as variáveis relevantes para o problema e os parâmetros dos conjuntos *Fuzzy*. O objetivo é restringir o número de variáveis apresentadas ao modelo para reduzir a complexidade do problema. Diferentes metodologias, como clusterização hierárquica [107, 108] ou ordenação por importância de variáveis [106], podem ser aplicadas para esse fim.

A geração das regras linguísticas é crucial, pois é onde todo o conhecimento do problema e explicabilidade do modelo são armazenados. Esta etapa consiste em três subtarefas: geração de antecedentes, filtragem de premissas e associação de consequentes. A geração de antecedentes é realizada de forma semi-exaustiva (Figura 4.9), combinando iterativamente todos os antecedentes possíveis, com remoção de combinações duplicadas. Em seguida, as regras são filtradas para remover aquelas pouco ou nunca ativadas, de forma a melhorar a interpretabilidade do modelo. Por fim, os consequentes são associados às premissas com base em uma métrica de afinidade.

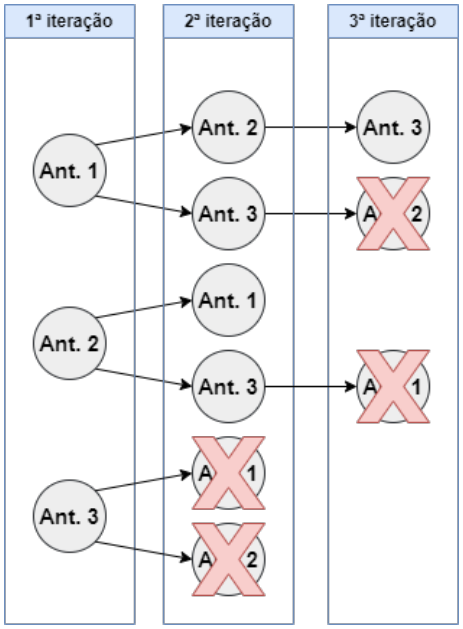

Figura 4.9: Processo de geração semi-exaustiva

A geração dos valores precisos da saída é dividida em agregação e defuzzificação. Na agregação, as regras com consequentes semelhantes, mas premissas diferentes, são unificadas para dar pesos diferentes às regras ativadas e evitar influências ambíguas. Na defuzzificação, os valores *Fuzzy* da saída são transformados de volta ao domínio real, utilizando-se o método da altura modificado para considerar os conjuntos *Fuzzy* de todas as variáveis.

4.4 Modelo e-AutoMFIS

Apesar de promissor, o AutoMFIS apresentou deficiências em sua execução, especialmente em problemas de alta dimensionalidade, seja por quantidade de observações ou variáveis. Além disso, é um modelo de aprendizagem global, onde as regras são escolhidas pela ativação do conjunto de treino em sua totalidade, o que dificulta a extração de conhecimento de forma local.

Para superar essas limitações, foi proposto o modelo e-AutoMFIS em [10], que utiliza a metodologia *ensemble* (Figura 4.10) para melhorar a execução. A premissa básica do *ensemble* é que a combinação dos resultados de vários modelos leva a uma previsão mais precisa do que cada modelo individualmente.

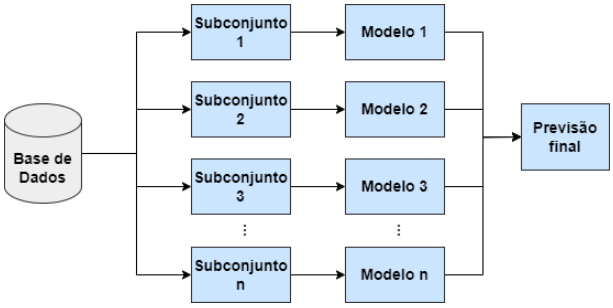

Figura 4.10: metodologia *ensemble*

O e-AutoMFIS beneficia-se desse conceito ao dividir problemas complexos em partes menores, reduzindo sua dimensionalidade e facilitando o processo de previsão. O modelo é estruturado em cinco módulos principais: Definição e organização do problema, Formação da base de regras, Filtragem e agregação de regras, Defuzzificação e Previsão *multistep*.

Na primeira etapa, os dados são pré-processados para seleção das variáveis mais correlatas, determinação da defasagem máxima e remoção de características não-estacionárias. Em seguida, é definido o dicionário linguístico para a fuzzificação dos dados, respeitando a interpretabilidade e utilizando funções de pertinência triangulares e trapezoidais.

O segundo módulo concentra-se na formação das bases de regras, seguindo a mesma metodologia do AutoMFIS, mas com a aplicação da metodologia *ensemble*. Diversas bases de regras são formadas com base em subamostras do conjunto de treinamento e posteriormente agregadas em um bloco final de regras.

O terceiro módulo trata da filtragem e agregação das regras para evitar redundância ou conflitos, o que pode prejudicar a acurácia e a interpretabilidade do modelo. Esse processo é realizado em duas etapas: interna e externa, para filtragem a nível local e da base final de regras, respectivamente.

Na fase de defuzzificação, o processo segue os mesmos moldes do AutoMFIS, mas com a possibilidade de fazer uso três técnicas diferentes: centróide, altura e altura modificada. Por fim, a previsão *multistep* é realizada, onde cada variável de interesse é prevista e os valores previstos são usados nos próximos passos de previsão, até atingir o número desejado de passos à frente.

5
Modelo proposto: e-AutoMFIS2

O e-AutoMFIS demonstrou sucesso em abordar as limitações de seu predecessor ao lidar com problemas de alta dimensionalidade. Ao decompor esses problemas em subproblemas de menor complexidade, o e-AutoMFIS captura efetivamente relações de curto e longo prazo, exibindo versatilidade em diversas aplicações e alcançando resultados competitivos em relação a modelos tradicionais, incluindo aqueles baseados em *deep learning*.

Como discutido em [10], o e-AutoMFIS representa um avanço na melhoria da robustez e eficiência computacional das técnicas de síntese automática de sistemas *Fuzzy*. No entanto, o modelo ainda carece de aprimoramentos.

Um dos pontos críticos identificados para melhoria é a otimização do modelo, dada a necessidade de se ajustarem diversos parâmetros de forma conjunta para encontrar a melhor configuração. A busca exaustiva nesse contexto é inviável, tornando essencial um estudo mais aprofundado dos problemas específicos para orientar a seleção das melhores configurações. Além disso, a otimização deve equilibrar acurácia e interpretabilidade, que são objetivos cruciais e, em essência, conflitantes.

Outro ponto de melhoria diz respeito à abordagem de subamostragem adotada pelo e-AutoMFIS, que atualmente é realizada de maneira aleatória. Embora essa abordagem gere diversidade nas amostras, ela pode resultar na seleção de informações pouco correlacionadas, levando à inclusão de regras pouco relevantes na base final. Portanto, o modelo pode se beneficiar da implementação de técnicas de amostragem guiada, visando selecionar variáveis e seus subconjuntos com maior dependência para formar regras mais eficazes.

Com base nessas considerações, o e-AutoMFIS2 foi desenvolvido utilizando-se técnicas estabelecidas na literatura. Para a otimização do modelo, foi empregado o algoritmo genético NSGA-II, especializado em problemas de otimização multiobjetivo, a fim de buscar soluções de configuração Pareto-ótimas e eliminar a necessidade de uma busca exaustiva. Quanto à subamostragem, o e-AutoMFIS2 adota uma abordagem semi-guiada, ponderando a seleção de amostras com base em valores normalizados de correlação cruzada e autocorrelação, de forma a aumentar a relevância das informações selecionadas.

Este capítulo apresenta a arquitetura do e-AutoMFIS2 e suas etapas, fornecendo uma compreensão abrangente da proposta deste trabalho.

5.1 Arquitetura do e-AutoMFIS2

O e-AutoMFIS2, assim como seu antecessor, mantém a estrutura de *ensemble*, porém incorpora uma etapa adicional para otimização da seleção de hiperparâmetros e modifica o método de subamostragem, utilizando uma abordagem semi-guiada baseada em correlação cruzada para a seleção de variáveis e autocorrelação para suas respectivas defasagens.

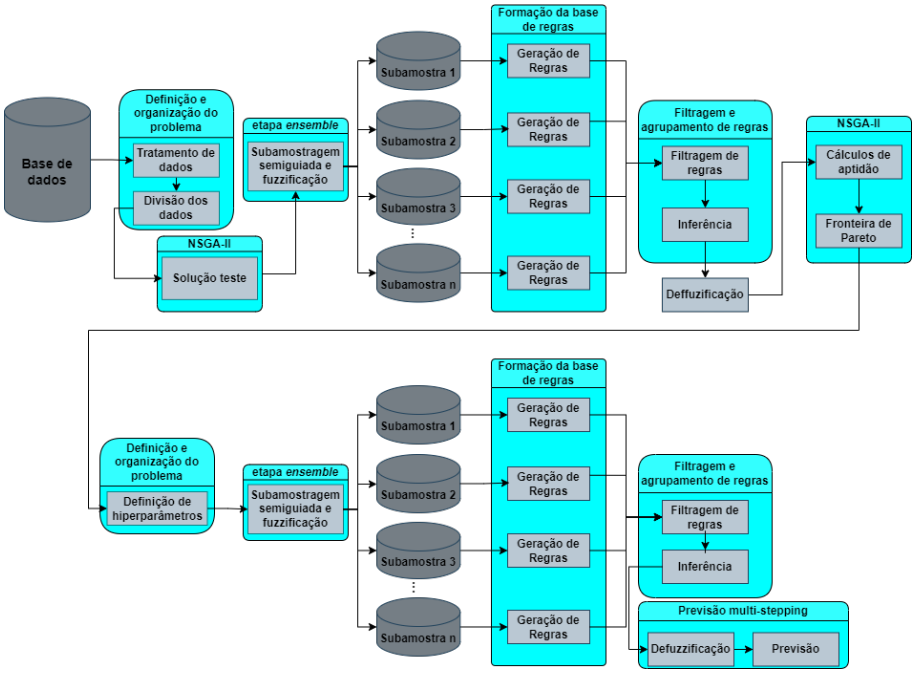

Figura 5.1: Arquitetura simplificada do modelo e-AutoMFIS2

Como ilustrado na Figura 5.1, a arquitetura do e-AutoMFIS2 é dividida em seis módulos que abrangem tanto a etapa de otimização de dados quanto a etapa de previsão *multistep*. É importante destacar que, embora os módulos relacionados à etapa de *ensemble*, formação da base de regras e filtragem e agregação de regras sejam duplicados para atender ambas as etapas, eles serão tratados como únicos, pois os procedimentos essenciais são os mesmos.

O primeiro módulo refere-se à definição e organização do problema, onde ocorre o pré-processamento da base de dados, incluindo a seleção de variáveis e a divisão dos dados para uso em diferentes etapas da modelagem, como treinamento, validação e teste.

O segundo módulo corresponde à etapa de *ensemble*, na qual é realizada a subamostragem semi-guiada, utilizando informações de correlação cruzada e autocorrelação para ponderar as escolhas potenciais, e a fuzzificação dos dados que serão apresentados aos previsores dentro do comitê.

O terceiro módulo refere-se ao NSGA-II, responsável pela etapa de otimização dos dados. Aqui, várias configurações são testadas para definir a fronteira de Pareto, de forma a selecionar aquela que melhor atende os objetivos estabelecidos, como a maximização da acurácia ou a busca por um equilíbrio entre acurácia e interpretabilidade.

O quarto módulo trata da formação da base de regras, onde ocorre a extração de conhecimento de cada subamostra para a criação das regras linguísticas intermediárias que irão compor a base final de regras. Esse procedimento é executado tanto na etapa de otimização de hiperparâmetros quanto na etapa de previsão.

O quinto módulo refere-se à filtragem e agregação de regras, onde são selecionadas as regras mais relevantes para compor a base final de regras. Esse processo é aplicado nas etapas de otimização de hiperparâmetros e previsão.

Por fim, o último módulo trata da previsão *multistep*, que engloba a defuzzificação para a obtenção de valores precisos, e a previsão multivariada do modelo para avaliação de sua acurácia.

Todos os módulos citados possuem subseções dedicadas a eles, com o objetivo de detalhar sua importância, objetivos e os diferentes métodos aplicados, fornecendo uma compreensão de seus funcionamentos e discutindo sua relevância.

5.1.1 Definição e organização do problema

Nesta etapa, o objetivo é conduzir o pré-processamento necessário para facilitar a extração de regras nas etapas subsequentes de formação da base de regras. Aqui, são realizadas operações como limpeza e organização dos dados, bem como a separação da base completa em conjuntos destinados a diferentes fases do processo de modelagem.

Na primeira parte deste módulo, o pré-processamento da base de dados é ma etapa fundamental para compreender o comportamento dos dados. Embora não esteja diretamente envolvida com o e-AutoMFIS2, essa fase é crucial para determinar a qualidade das previsões e a auditabilidade do modelo. Questões como seleção de variáveis, detecção e remoção de tendências, e determinação de defasagens são abordadas.

A seleção de variáveis tem o papel de reduzir complexidade do problema: descartam-se aquelas que não possuem correlação linear ou não-linear com a variável de saída. Diversas abordagens podem ser empregadas para avaliar a importância das variáveis. Em situações de relações lineares, métodos como o coeficiente de correlação de Pearson e a regressão Lasso são aplicáveis. Para análises de relações não-lineares, o coeficiente de correlação por postos de Spearman e o modelo Random Forest são algumas das opções viáveis.

A detecção e remoção de tendências é uma etapa importante em modelos baseados em sistemas *Fuzzy*, pois a presença de tendências nos dados pode prejudicar a qualidade das previsões, uma vez que os grupos predefinidos podem não mais refletir a realidade dos dados.

Em determinados cenários, a análise visual de uma série temporal pode sugerir a presença de tendência. No entanto, esse método é susceptível a interpretações subjetivas, o que limita sua confiabilidade. Portanto, é imperativo recorrer a técnicas mais robustas para uma avaliação precisa da existência de tendência na série temporal. Nesse contexto, abordagens como a decomposição de séries temporais ou o teste estatístico de Mann-Kendall são recomendadas. Essas técnicas proporcionam uma análise mais sólida e confiável, permitindo uma avaliação com maior grau de certeza quanto à presença de tendência.

Quanto à remoção da tendência de uma série, existem diversos métodos disponíveis, conforme discutido no Capítulo 2. O e-AutoMFIS2, assim como seus predecessores, adota a metodologia de remoção por meio da regressão linear, que é uma técnica bem estabelecida e eficaz.

Para o e-AutoMFIS2 e seus predecessores, a definição do valor máximo de defasagem é outro aspecto relevante. Esse valor determina a quantidade máxima de registros retroativos de cada variável que será utilizada na extração de conhecimento. Geralmente, esse valor é definido manualmente, frequentemente com base em informações de correlação cruzada e autocorrelação.

Após esses procedimentos, ocorre a divisão dos dados para a modelagem. Geralmente, essa divisão é feita em conjuntos de treinamento,validação e teste, permitindo avaliar a capacidade de generalização do modelo. No caso do e-AutoMFIS2, a divisão é feita em quatro partes (vide Figura 5.2) : treinamento (para extração de conhecimento), validação (para avaliação das regras formadas pelo comitê), teste NSGAII (otimização, usado para calcular a aptidão das soluções no algoritmo) e teste (previsão, usado para avaliar a acurácia das previsões do modelo). Essa separação em dois conjuntos de teste é fundamental para evitar viés nos resultados, já que os dados não são apresentados ao modelo mais de uma vez. Com essa abordagem, busca-se garantir a robustez e eficácia do modelo ao longo de todo o processo de modelagem.

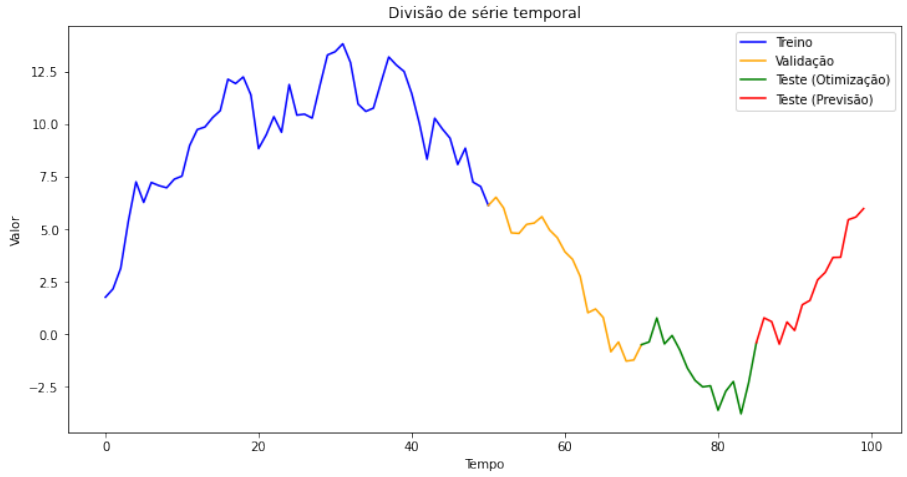

Figura 5.2: Exemplo de divisão de dados entre treinamento, validação e teste (otimização e previsão)em uma base de dados

5.1.2 Etapa *ensemble*

A etapa *ensemble* constitui o módulo que abrange as operações relacionadas à preparação da base de dados para apresentação ao comitê de previsores do modelo. Este módulo é subdividido em duas tarefas principais: subamostragem e fuzzificação dos dados, iniciando assim o processo de modelagem do e-AutoMFIS2.

A primeira parte deste módulo diz respeito à subamostragem dos dados, cujo propósito é reduzir a complexidade do problema, transformando-o em diversos problemas menores e, consequentemente, facilitando a extração de conhecimento pelos previsores. Esta metodologia permitiu ao e-AutoMFIS solucionar a principal limitação de seu antecessor em relação a bases de dados de alta dimensionalidade. Entretanto, a amostragem no e-AutoMFIS é realizada de forma aleatória, o que pode resultar em amostras pouco correlacionadas, dificultando o aprendizado do modelo.

No e-AutoMFIS2, esse processo foi aprimorado para adotar uma abordagem semi-guiada, na qual as subamostras são selecionadas aleatoriamente, mas com a aplicação de ponderações tanto para as variáveis quanto para suas respectivas defasagens. Como demonstrado na Figura 5.3 o processo de subamostragem dos dados é realizada em duas etapas: subamostragem dimensional e subamostragem temporal.

A etapa de subamostragem dimensional é responsável pela seleção das variáveis que formarão as subamostras a serem apresentadas ao modelo. Nesse processo, as variáveis previamente selecionadas na etapa de Definição e Organização do problema são ponderadas com base nos valores normalizados da correlação cruzada com um *target* previamente definido. Isso é exemplificado na Eq. 5-1, onde o peso da série x ($peso_x$) é calculado pelo quociente entre a correlação cruzada ($\rho_{x,target}$) de x com o *target* e a soma das correlações das demais séries com o *target*. Esse procedimento mantém um grau de aleatoriedade na seleção, pois o número de séries que formarão o subconjunto é escolhido aleatoriamente, embora a escolha das variáveis seja influenciada pela ponderação de cada uma.

$$peso_x = \frac{\rho_{x,target}}{\sum_{i=1}^{n} \rho_{n,target}} \tag{5-1}$$

Na etapa de subamostragem temporal, são selecionadas amostras em diferentes instantes de tempo para cada variável selecionada na etapa anterior, formando a subamostra a ser apresentada ao modelo. Cada valor $x(t)$ da série em um instante t recebe uma ponderação relacionada ao seu valor normalizado de autocorrelação com o dado mais recente disponível na série. Conforme demonstrado na Eq. 5-2, o peso de um valor defasado ϵ no instante $t-1$ é calculado como o valor de autocorrelação $\rho_{\epsilon_{t-1}}$ dividido pela soma dos coeficientes de autocorrelação de todas as n defasagens. Ressalta-se que n representa o número máximo de defasagens admitidas pelo modelo, previamente definido no módulo anterior. A quantidade de valores defasados escolhidos para cada variável é determinada aleatoriamente, enquanto as ponderações são usadas exclusivamente para selecionar os instantes que comporão o subconjunto.

$$W_{\epsilon_{t-1}} = \frac{\rho_{\epsilon_{t-1}}}{\sum_{i=1}^{n} \rho_n} \tag{5-2}$$

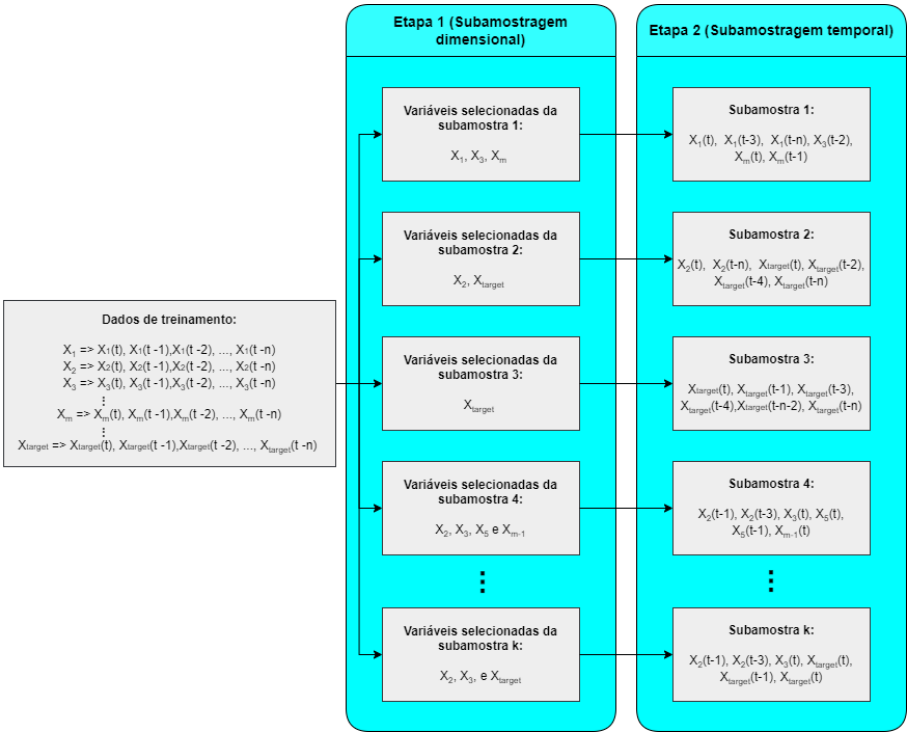

Figura 5.3: Esquema detalhado do processo de subamostragem do e-AutoMFIS2

Após a seleção da amostra a ser apresentada ao modelo, ocorre a fuzzificação dos dados para gerar as regras linguísticas. Esta etapa é definida pela formação do dicionário linguístico, incluindo a quantidade de conjuntos *Fuzzy* e suas funções de pertinência. No e-AutoMFIS2, a definição desses parâmetros é realizada durante a otimização multiobjetivo. No entanto, aspectos importantes, especialmente relacionados à interpretabilidade do modelo, devem ser considerados independentemente desses parâmetros.

Um exemplo disso é a escolha da estratégia de particionamento *Fuzzy*. Tanto no e-AutoMFIS2 quanto em seu antecessor, é utilizada a partição *Fuzzy* forte (*Strong Fuzzy Partition*), na qual a soma dos graus de pertinência de um valor pertencente ao universo de discurso é sempre igual a 1 (vide Equação 5-3).

$$\sum_{i=1}^{n} \mu_{A_n}(x) = 1, \forall x \in X \tag{5-3}$$

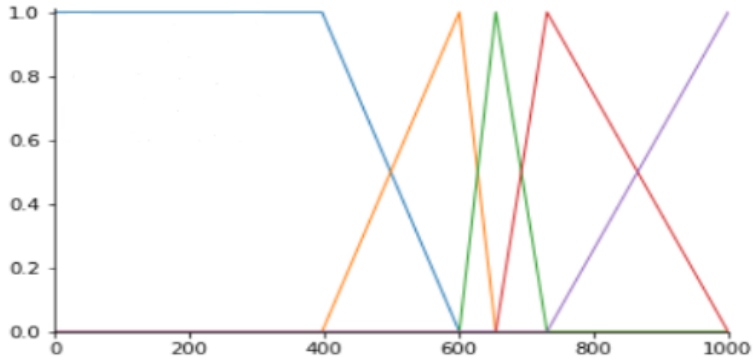

Figura 5.4: Representação da partição *Fuzzy* forte

Outro aspecto importante é a divisão dos conjuntos *Fuzzy*, para a qual são empregados três métodos distintos. O primeiro é a partição uniforme dos conjuntos, onde estes são espaçados de forma igualitária ao longo do universo de discurso, como mostrado na Figura 5.5 (b). Essa abordagem é eficaz apenas em casos de distribuição homogênea dos dados.

Para casos de distribuição não homogênea, é aplicado o método de Tukey, que se baseia nos percentis dos dados analisados (vide Figura 5.5 (c)) e é especialmente indicado para séries temporais com distribuição gaussiana ou normal.

Quando a variável em questão não se enquadra nos comportamentos mencionados anteriormente, tanto o método homogêneo quanto o de Tukey podem não ser os mais adequados para particionar os conjuntos *Fuzzy*. Nesses casos, o e-AutoMFIS2 oferece a opção do método baseado no agrupamento dos dados.

Essa abordagem consiste na divisão dos conjuntos com base nos n *clusters* criados para a série temporal, sendo n o número de conjuntos preestabelecidos no modelo. Os centros desses *clusters* são adotados como os picos das respectivas funções de pertinência. Essa metodologia é particularmente útil, uma vez que este método busca agrupar os dados de acordo com a similaridade, enquanto minimiza a variância entre os grupos, garantindo assim que a distribuição dos conjuntos *Fuzzy* represente os dados de forma precisa e fidedigna (ver Figura 5.5 (d)).

É importante ressaltar que a seleção do método mais adequado pode ser realizada por meio da otimização de hiperparâmetros. No entanto, dado que a escolha ideal depende principalmente da distribuição dos dados na série temporal, é recomendável que a definição seja baseada na análise das variáveis realizada no módulo anterior. Esse enfoque não apenas reduz a complexidade do espaço de busca do algoritmo genético, simplificando o processo de otimização multiobjetiva, mas também destaca a relevância do conhecimento especializado em relação ao problema em questão.

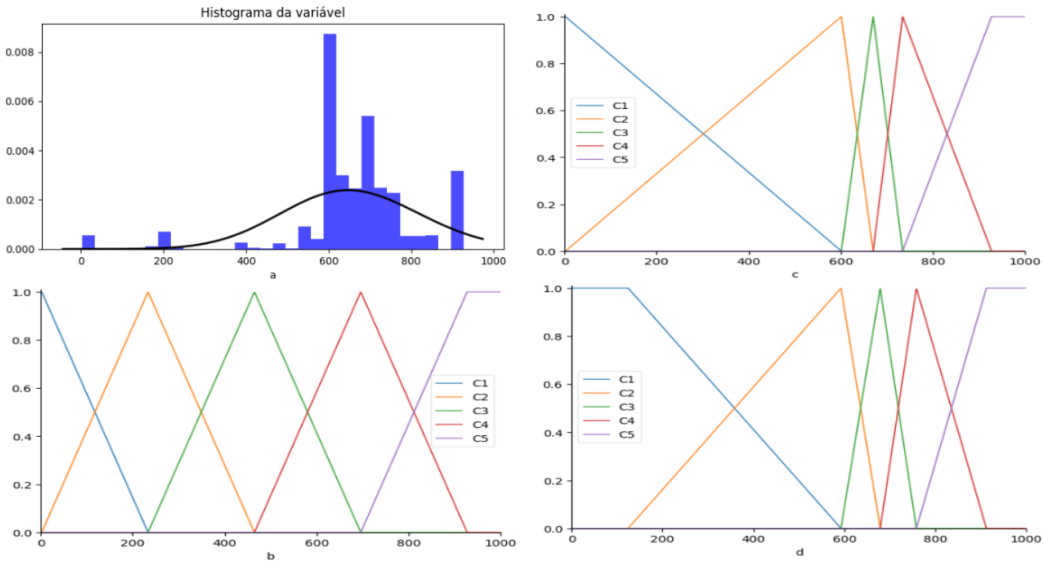

Figura 5.5: Comparação de geração dos conjuntos *Fuzzy* usando as metodologias disponíveis para o e-AutoMFIS2

Outro aspecto relevante na etapa de fuzzificação do e-AutoMFIS2 é a escolha do formato das funções de pertinência. O modelo utiliza predominantemente funções triangulares devido à sua simplicidade e facilidade de análise dos resultados. No entanto, uma abordagem distinta é adotada para as extremidades do universo de discurso, onde são empregadas funções trapezoidais. Isso é feito com o intuito de assegurar que todos os valores possíveis durante a etapa de previsão tenham uma representação adequada no domínio *Fuzzy*.

Todos os aspectos mencionados acima visam preservar a interpretabilidade do modelo, garantindo que as restrições necessárias sejam atendidas, como a complementaridade, distinguibilidade e cobertura das funções de pertinência, conceitos abordados anteriormente no capítulo 4.

5.1.3 NSGA-II

Conforme apontado por [10], uma das principais dificuldades no uso do e-AutoMFIS é a determinação da configuração ótima do modelo. Isso se deve à presença de diversos parâmetros que devem ser considerados durante o ajuste, como indicado na Tabela 5.1, que lista todos os parâmetros do modelo e suas descrições. A busca exaustiva por todas as combinações possíveis de parâmetros torna-se inviável devido à sua quantidade.

Além disso, surge a questão de aprimorar a acurácia do modelo sem comprometer significativamente sua interpretabilidade, o que representa um desafio adicional. Essas razões motivaram a aplicação de um sistema mais sofisticado para a otimização do modelo, em lugar de uma busca exaustiva. Essa abordagem busca equilibrar a melhoria da acurácia com a preservação da interpretabilidade do modelo.

Para essa tarefa, foi selecionado o NSGAII, conforme apresentado no capítulo 3, um algoritmo de otimização amplamente reconhecido na literatura por sua eficácia na resolução de problemas de múltiplos objetivos.

Tabela 5.1: Lista de parâmetros do modelo e-AutoMFIS

Nome da variável	**Descrição**
Lag	Valor da defasagem máxima da série
HPrev	Horizonte de previsão
NumInput	Quantidade de séries para compor o subconjunto
FuzzySets	Quantidade de conjuntos *Fuzzy* por variável
NumPred	Número de subsistemas para o comitê de previsores
MinAct	Ativação mínima das premissas
MaxRule	Quantidade máxima de premissas em uma regra

O esquema de execução do algoritmo de otimização do modelo pode ser observado na Figura 5.6, onde a população é composta pelos parâmetros do e-AutoMFIS2. É importante destacar que também é possível selecionar subconjuntos dos parâmetros para facilitar o processo de convergência do modelo. Outro ponto relevante sobre a população é a determinação da quantidade de indivíduos que a compõem. Esse valor deve ser escolhido de modo a garantir que o espaço de busca contenha uma quantidade suficiente de soluções, sem que isso tenha um grande impacto negativo no custo computacional.

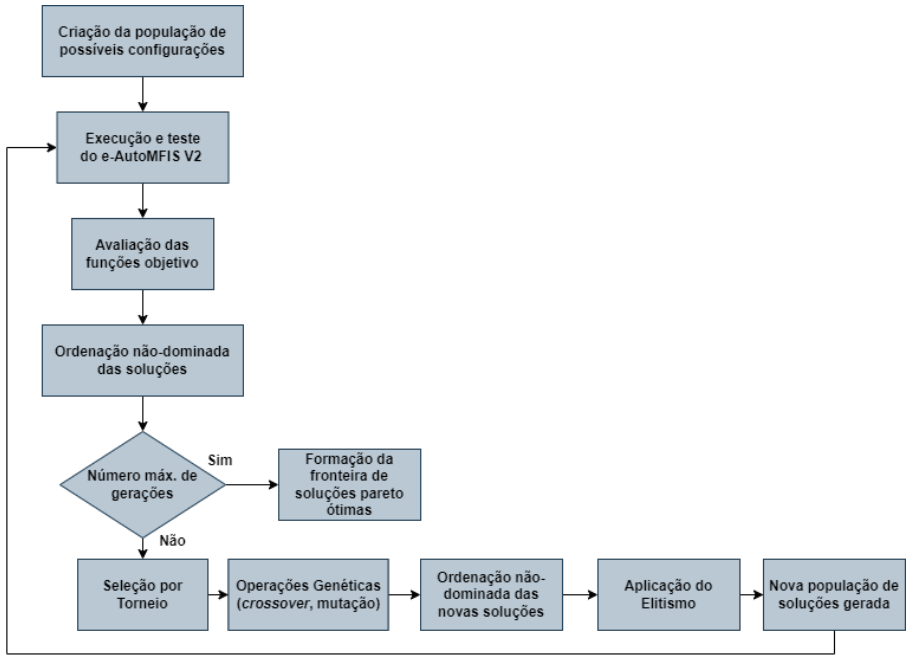

Figura 5.6: Processamento do NSGA-II aplicado ao e-AutoMFIS2

O segundo aspecto do NSGA-II é a execução do modelo e-AutoMFIS utilizando as configurações que compõem o espaço de busca do algoritmo e o seu teste, que servirá como referência para a próxima etapa. É válido ressaltar que os dados de teste utilizados nesta etapa são um conjunto separado especialmente para o uso na etapa de otimização, conforme explicado no módulo de Definição e Organização do Problema.

A terceira etapa da operação do NSGA-II refere-se à avaliação das funções objetivo, que no caso do e-AutoMFIS2, são as métricas de avaliação utilizadas para o modelo, tanto em relação à acurácia das previsões multivariadas quanto à interpretabilidade do modelo gerado.

A quarta etapa refere-se à ordenação não-dominada das configurações do modelo. Nesta fase, todas as soluções são comparadas pelo método do NSGA-II para elaboração das fronteiras não-dominadas, começando pelas soluções que não são dominadas por alguma outra, até que todas as soluções pertençam a algum nível.

Depois das configurações serem ordenadas, é realizado um processo de seleção por meio do método de torneio. As soluções são comparadas, inicialmente, com base nos níveis estabelecidos na etapa anterior. No caso de uma comparação entre soluções pertencentes ao mesmo nível, elas são então avaliadas com base em seu coeficiente de multidão. As soluções com os maiores valores de coeficiente de multidão são selecionadas, garantindo assim a diversidade da população.

Uma vez selecionados os indivíduos mais aptos pela etapa de seleção, eles passam pelos operadores genéticos, como *crossover* e mutação, para gerar novos indivíduos (configurações) a serem testados no modelo.

O processo de elitismo é responsável por preservar os indivíduos com melhor classificação da geração anterior para se juntarem aos novos indivíduos gerados pela etapa anterior. O elitismo é aplicado para evitar a perda de possíveis soluções pareto-ótimas. Caso as soluções da geração anterior sejam superiores, elas são mantidas na nova população.

Devido ao processo de elitismo, o número de configurações possíveis pode exceder o tamanho da população pré-estabelecida. Portanto, alguns indivíduos devem ser eliminados antes que o processo continue. O NSGA-II seleciona as melhores soluções, de acordo com suas posições nas camadas não-dominadas, até que o número de indivíduos predefinido seja atingido, formando assim a nova população.

É fundamental salientar que esse processo continua até que a quantidade predefinida de gerações seja alcançada para gerar a fronteira de soluções Pareto-ótimas final. A partir dessa fronteira, são selecionadas as melhores configurações que atendem de maneira mais satisfatória aos objetivos estabelecidos, seja com um modelo mais equilibrado, onde ambas as métricas de acurácia e interpretabilidade são atendidas, um modelo mais auditável, com prioridade para a interpretabilidade, ou um modelo de alta acurácia, com menos foco na interpretabilidade. É importante ressaltar que no e-AutoMFIS2 não se aplica alguma técnica heurística ou de aprendizado de máquina para selecionar a melhor solução não-dominada. Portanto, o julgamento e a seleção da melhor configuração para o problema tratado são de responsabilidade do usuário.

5.1.4 Formação da base de regras

NO módulo de formação da base de regras, as informações extraídas da base de dados são armazenadas para utilização na etapa de previsão. As regras também são o meio pelo qual o modelo pode transmitir o conhecimento extraído, tornando-as o principal aspecto a ser analisado em termos de interpretabilidade.

A síntese automática de regras do e-AutoMFIS2 segue a mesma metodologia proposta por seus antecessores, dividida em duas sub-tarefas: formulação de premissas e associação de consequentes. A única diferença está na aplicação, já que ela é usada tanto para otimização de hiperparâmetros, com o dicionário linguístico incluso, quanto na etapa de previsão do modelo, após a otimização dos parâmetros.

Na formulação de premissas, os termos linguísticos são combinados por meio de suas funções de pertinência para formar os antecedentes das regras. A Eq. 5-4 apresenta a estrutura de uma premissa do ponto de vista matemático, onde μ_{A_p} representa a função de pertinência da premissa p, calculada utilizando o operador t-norma $(*)$, neste trabalho representado pelo produto algébrico. Isso é realizado a partir do grau de ativação $\mu_{A_{j,1}}(x_1)$ do termo linguístico ligado ao conjunto *Fuzzy* k da i-ésima variável e da série $j \in 1, 2, ..., s$ a ser prevista.

$$\mu_{A_p} = \mu_{A_{j,1}}(x_1) * \mu_{A_{j,2}}(x_2) * ... * \mu_{A_{j,3}}(x_i) \tag{5-4}$$

A formulação dessas premissas é conduzida por meio de uma técnica de busca semi-exaustiva, conforme descrito por [9, 10]. Essa abordagem busca iterativamente formar as premissas, começando com um único termo linguístico por regra e aumentando gradualmente esse número até atingir o máximo de antecedentes por regra, um parâmetro configurável pelo usuário ou determinado durante a otimização do modelo.

O processo inicia-se com a avaliação de todos os termos linguísticos das variáveis selecionadas pelo subconjunto, resultando em regras contendo apenas um antecedente. Em seguida, essas regras são submetidas a um limiar de grau de ativação, também configurável pelo usuário ou definido durante a otimização de hiperparâmetros. As premissas que não alcançam o limiar são descartadas.

Nas iterações subsequentes, as premissas formadas são combinadas com os termos linguísticos de variáveis diferentes das premissas já existentes, evitando assim que as primeiras possuam antecedentes contraditórios. Um exemplo disso seria se uma série, que reflete a altura de uma população, possuísse na mesma premissa os termos linguísticos ligados aos conceitos de "alto"e "baixo"ao mesmo tempo. As novas premissas são reavaliadas em relação ao limiar de ativação estabelecido. Aquelas que ficam abaixo do limiar preestabelecido têm seu crescimento interrompido, enquanto aquelas que ficam acima desse valor mínimo são inseridas no conjunto de premissas selecionadas e passam novamente pelo processo de acréscimo de novos antecedentes. Esse procedimento é repetido até que o número máximo de antecedentes por premissa, um parâmetro ajustado pelo usuário, seja atingido.

Em relação ao grau de ativação mínimo, o e-AutoMFIS2 utiliza os mesmos métodos de seu antecessor: a Cardinalidade e a Cardinalidade não-nula. Ambas calculam a média do grau de ativação das premissas nos dados de treinamento, diferenciando-se apenas no uso de registros com ativações nulas [10].

O objetivo de ambas as métricas é selecionar as premissas mais relevantes em relação aos dados de treinamento, porém cada uma delas enfatiza um aspecto diferente na formação das regras. No caso da Cardinalidade, há uma preferência por premissas mais gerais, ou seja, com frequência alta e grau relativamente baixo de ativação. Isso ocorre porque a Cardinalidade supõe que as premissas possuem uma representatividade tanto em frequência quanto em grau de ativação.

A Cardinalidade não-nula pode ser considerada uma contraparte da primeira métrica, pois prioriza regras com altos graus de ativação que são ativadas com menos frequência. No entanto, é necessário cautela ao utilizar essa métrica, pois ela pode favorecer regras muito extensas e com frequências extremamente baixas, desde que apresentem um grau de pertinência razoável. Para lidar com esses casos, o modelo também emprega um parâmetro de frequência mínima, garantindo que a etapa de formulação aceite regras menos frequentes, mas não aquelas que foram ativadas pouquíssimas vezes.

Após a definição das premissas, inicia-se o processo de sua associação aos consequentes. Para tal, calcula-se o grau de pertinência $\mu_{A_{j,i}}(y)$ da saída y para cada conjunto *Fuzzy* i de cada variável j a ser prevista. Em seguida, as métricas de associação são calculadas entre os dados de entrada e saída, utilizando seus graus de ativação, e o conjunto *Fuzzy* que apresenta a maior compatibilidade é associado à premissa da regra.

Essa etapa requer cautela, pois a forma como é calculado o grau de similaridade pode gerar regras com consequentes inadequados, afetando a precisão do modelo. Por isso, o e-AutoMFIS2 utiliza duas métricas: o Grau de Confiança *Fuzzy* (GCF) e a Compatibilidade Média (CM).

$$GCF = \frac{\mu_{A_p}(X) \cdot \mu_{s,n}(Y_s)}{||\mu_{A_p}(X)|| \times ||\mu_{s,n}(Y_s)||} \tag{5-5}$$

O GCF avalia a semelhança entre dois vetores de forma análoga ao conceito de ortogonalidade, indicando compatibilidade total para valores iguais a 1 e nenhuma compatibilidade para valores iguais a 0. Essa metodologia calcula o cosseno dos dois vetores, como mostrado na Eq. 5-5, onde $\mu_{A_p}(X)$ representa os graus de pertinência da premissa p para o conjunto de treino X, e $\mu_{s,n}(Y_s)$ representa os graus de pertinência para os n conjuntos *Fuzzy* que representam a variável de saída s.

Por sua vez, a CM avalia cada valor de entrada do conjunto de treino em relação às premissas formadas, gerando um grau de ativação $\mu_{A_p}(X)$ e o grau de pertinência de cada conjunto *Fuzzy* $\mu_{s,n}(Y_s)$ para os dados de saída. A Compatibilidade Média [10] é calculada como a média do produto dos graus de ativação não-nulos e os valores de pertinência de cada conjunto *Fuzzy* da variável de saída, conforme a Eq. 5-6, associando cada premissa ao conjunto que apresenta a maior compatibilidade média.

$$CM = \frac{1}{N}\sum_{i=1}^{n}\mu_{A_p}(x_i)\mu_{s,n}(y_{i,s}), \tag{5-6}$$

A escolha da métrica de associação deve ser avaliada caso a caso, porém, em muitos problemas, o cálculo da compatibilidade média é mais adequado, pois prioriza a seleção do consequente mais apropriado quando a premissa é ativada. Além disso, a CM é vantajosa para regras importantes, mas com ativação esporádica, pois considera apenas os registros que geram ativação nas premissas.

5.1.5 Filtragem e agregação de regras

O e-AutoMFIS2, assim como seu antecessor, gera n conjuntos de regras correspondentes às n subamostras criadas durante a etapa de *ensemble*, simplificando o processo de formação de regras e reduzindo a dimensionalidade do problema. No entanto, essa abordagem pode resultar em regras redundantes, conflitantes ou excessivamente específicas, o que prejudica tanto a interpretabilidade quanto a acurácia. Por isso, foi desenvolvida a etapa de filtragem de regras.

O processo de filtragem no e-AutoMFIS2, seguindo a abordagem do seu antecessor [10], ocorre em duas fases distintas: a filtragem interna, realizada dentro de cada conjunto de regras das subamostras, e a filtragem externa, aplicada globalmente a todas as regras. A Figura 5.7 ilustra esse processo de filtragem.

A filtragem interna inicia-se com a remoção de regras semelhantes ou conflitantes dentro de cada conjunto de regras, eliminando aquelas de menor importância. Para isso, utiliza-se a métrica de similaridade entre os graus de pertinência das premissas, conforme a Eq. 5-7. Nessa equação, os graus de ativação $\mu_{A_P}(x_i)$ e μ_{A_Q} são calculados para cada par de premissas P e Q em relação aos dados de treinamento x_i. Se a similaridade $S_{P,Q}$ entre as premissas exceder um limite S_{max}, as premissas são consideradas similares e a de menor ativação média é descartada.

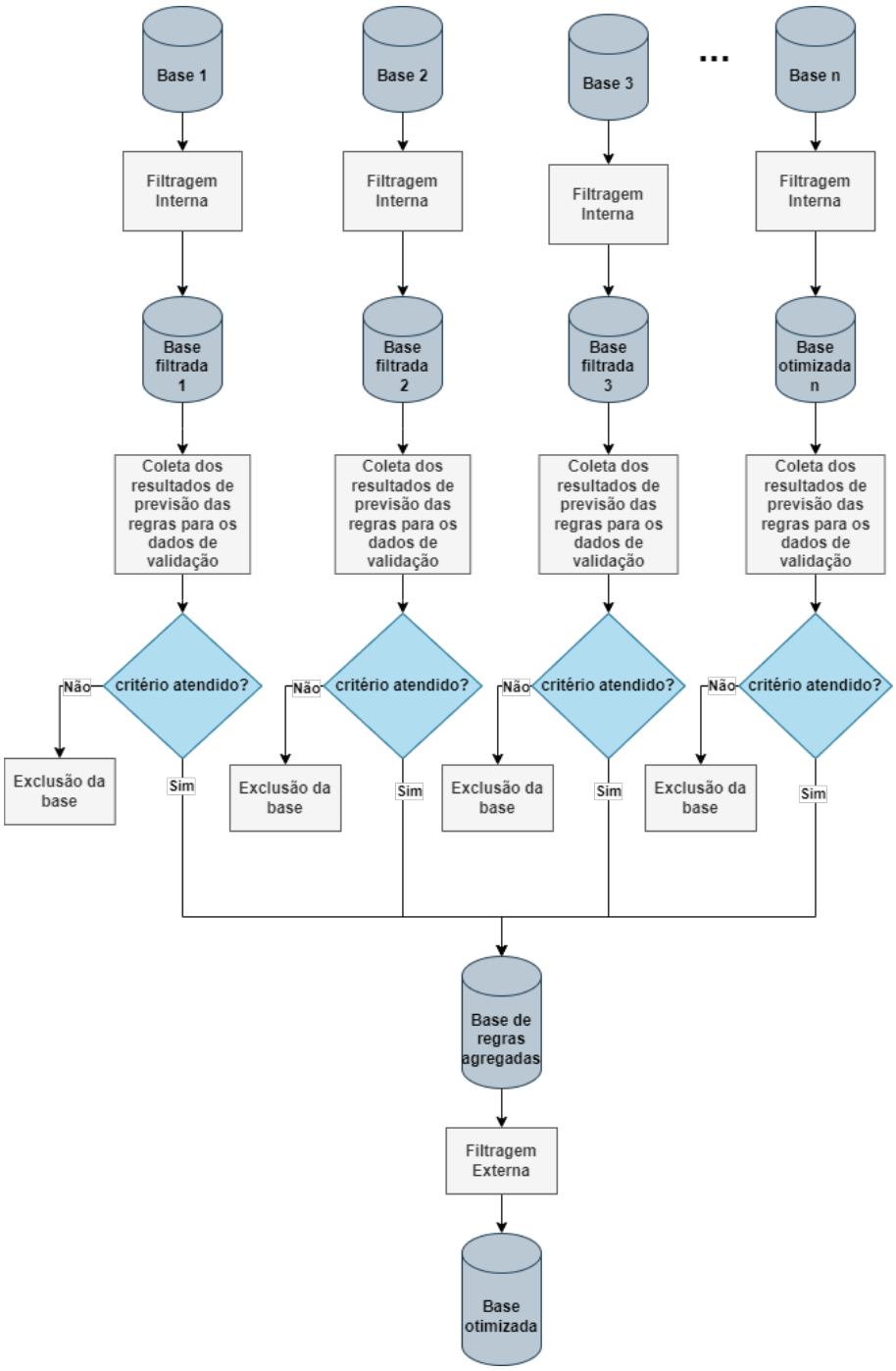

Figura 5.7: Diagrama de blocos da etapa de filtragem

Para distinguir entre regras conflitantes e similares, podem ser definidos dois limiares: S_{cmax} para conflitos e S_{smax} para similaridades. No e-AutoMFIS2, conforme a configuração da versão anterior [10], ambos os limiares recebem o mesmo valor, devido à dificuldade de se determinarem empiricamente os valores ideais.

$$S_{P,Q} = \sum_{i=1}^{n} \frac{min[\mu_{A_P}(x_i), \mu_{A_Q}(x_i)]}{max[\mu_{A_P}(x_i), \mu_{A_Q}(x_i)]} \tag{5-7}$$

O segundo objetivo da filtragem interna é selecionar as regras mais importantes para serem testadas com um conjunto de validação separado. Utiliza-se o método dos Mínimos Quadrados Restritos (MQR), já aplicado no AutoMFIS [9] e no e-AutoMFIS [10]. A Eq. 5-8 representa esse método, onde $\mu_{C_{s,n}}(y_{s,t})$ é o grau de pertinência do consequente C da série s ao conjunto *Fuzzy* n, e $\mu_{A_p}(x_t)$ é o grau de pertinência da premissa p gerada pelo valor de entrada x_t. O parâmetro w_p representa o peso atribuído a essa entrada (variando de 0 a 1), ajustado conforme o grau de ativação das premissas. A seleção das regras baseia-se nos valores calculados, descartando-se aquelas com pesos próximos de zero.

$$\begin{aligned} min(\sum_{t=1}^{T}(\mu_{C_{s,n}}(y_{s,t}) - \sum_{p=1}^{P} w_p \mu_{A_p}(x_t)) \\ s.a. \sum_{p=1}^{P} w_p = 1 \end{aligned} \tag{5-8}$$

O terceiro e último objetivo da filtragem interna é selecionar as regras que irão compor a base de regras otimizada, considerando sua capacidade de generalização. Avalia-se se as regras, após as etapas anteriores, são capazes de generalizar o problema modelado. Para isso, utiliza-se um conjunto de dados de validação separado na etapa de Definição e Organização do Problema. As regras são selecionadas com base em métricas definidas para avaliação do modelo, comparando-se os resultados da previsão com o conjunto de validação. Aquelas que obtêm um erro inferior a um limiar ϵ_{min} são selecionadas para compor a base final.

Após a etapa de filtragem interna, a base de regras final é composta pela união de todas as regras que satisfizeram os critérios estabelecidos na primeira parte da filtragem. O processo de filtragem externa segue uma abordagem semelhante à interna, pois existe a possibilidade de surgimento de regras similares, conflitantes ou duplicadas devido à combinação das bases da etapa anterior. Assim, a base final de regras é submetida novamente ao filtro de similaridade, de forma eliminar aquelas consideradas menos importantes segundo essa métrica.

Posteriormente, a base final passa por um último filtro em relação à importância das regras. Assim como na filtragem interna, aqui também é empregado o método MQR, onde os pesos determinados por essa técnica são utilizados como referência para a eliminação de regras com pesos pequenos, indicativos de baixa importância. No entanto, há uma pequena distinção em relação à mesma etapa na filtragem interna: as regras selecionadas mantêm seus respectivos pesos. Essa ponderação é preservada, pois é empregada para identificar as regras mais relevantes para o próximo passo deste módulo, que é a agregação.

A etapa de agregação é fundamental para determinar como as regras ativadas podem ser combinadas para formar o conjunto *Fuzzy* de saída. Nessa fase, todas as regras ativadas com o mesmo consequente são unidas para a formação do conjunto de saída. A equação 5-9 exemplifica o processo de agregação, onde $\mu_{B^*_{i,j}}(y)$ representa o conjunto de saída, $\mu_{A_P}(x)$ e $\mu_{A_Q}(x)$ são os graus das regras ativadas que compartilham o mesmo consequente da série i e conjunto *Fuzzy* j ($B_{i,j}$).

A etapa de agregação é uma etapa que auxilia a determinar como as regras ativadas podem ser combinadas para formar o conjunto *Fuzzy* de saída. Como o nome sugere, todas as regras ativadas com um mesmo consequente são unidas para a formação do conjunto de saída nessa etapa. A Eq. 5-9 exemplifica o processo de agregação onde $\mu_{B^*_{i,j}}(y)$ é o conjunto de saída, $\mu_{A_P}(x)$ e $\mu_{A_Q}(x)$ são os graus das regras ativadas que possuem o mesmo consequente da série i e conjunto *Fuzzy* j $B_{i,j}$.

$$\mu_{B^*_{i,j}}(y) = f(\mu_{A_P}(x), \mu_{A_Q}(x), \mu_{B_{i,j}}(y)) \tag{5-9}$$

A função utilizada para realizar a agregação no e-AutoMFIS2 é o Modus Ponens Generalizado (MPG), o mesmo método empregado na versão anterior. Nessa abordagem, o operador máximo é utilizado para a agregação de regras, conforme demonstrado na equação 5-10. Vale ressaltar uma pequena modificação aplicada ao MPG para sua utilização no modelo atual: além dos graus de ativação das regras, os pesos (w_p) determinados na etapa de filtragem externa são considerados para ponderar esses graus ($\mu_{A_P}(x)$), resultando no conjunto de saída ($\mu_{B_{s,n}}(y)$). Essa inclusão dos pesos adiciona uma camada adicional de confiabilidade às regras ativadas, teoricamente melhorando a acurácia do modelo sem comprometer significativamente sua interpretabilidade.

$$\mu_{B^*_{s,n}}(y) = min\{max\{w_P \cdot \mu_{A_P}(x), ...\}, \mu_{B_{s,n}}(y)\} \tag{5-10}$$

5.1.6 Previsão *multistep*

Após a formação e filtragem das regras do modelo, é necessário testá-las para avaliar sua capacidade de generalização ao lidar com dados novos durante a etapa de previsão multistep. Nesse módulo, existem duas etapas distintas: a defuzzificação e a coleta das previsões do modelo. Foram selecionadas três para serem utilizadas no e-AutoMFIS2: centróide, altura e altura modificada.

O método do centróide determina o valor de saída como sendo o centróide da figura formada pelos conjuntos *Fuzzy* de saída ativados, conforme demonstrado pela Eq. 5-11, onde o valor de saída defuzzificado y_c é obtido pelo quociente entre a integral do produto da variável y_i, que representa os valores possíveis no universo de discurso do conjunto *Fuzzy* ativado e o seu respectivo grau de pertinência $\mu_B(y_i)$ e a integral dos graus de pertinência.

$$y_c = \frac{\int y_i \mu_B(y_i)\, dy}{\int \mu_B(y_i)\, dy} \tag{5-11}$$

O método da altura adota uma abordagem mais simples, definindo o valor preciso como o ponto de máximo grau de pertinência apresentado pelos conjuntos *Fuzzy* de saída ativados. A Eq. 5-12 ilustra esse processo, onde o valor é obtido pela razão entre a soma ponderada dos valores y_i e seus respectivos graus de pertinência máxima $\mu_B(y_i)$ e a soma total dos graus de pertinência máxima.

$$y_c = \frac{\sum_{i \in S} y_i \mu_B(y_i)}{\sum_{i \in S} \mu_B(y_i)} \tag{5-12}$$

O método da altura modificada foi desenvolvido para reduzir o viés causado pelos conjuntos com suporte maiores. A Eq. 5-13 representa matematicamente esse método, que é semelhante ao método da altura apresentado anteriormente. No entanto, ambas as componentes são ponderadas por δ, que representa o tamanho do suporte dos conjuntos *Fuzzy* de saída ativados.

$$y_c = \frac{\sum_{i \in S} y_i \mu_B(y_i)/\delta}{\sum_{i \in S} \mu_B(y_i)/\delta} \tag{5-13}$$

Um exemplo prático é ilustrado na Figura 5.8, onde o método do centróide é deslocado para a esquerda devido à área de suporte maior do conjunto 1 (vermelho), mesmo que o segundo conjunto tenha um grau de ativação maior. Enquanto isso, o método de altura modificada pondera as áreas de suporte com seus respectivos graus de pertinência, minimizando esse efeito.

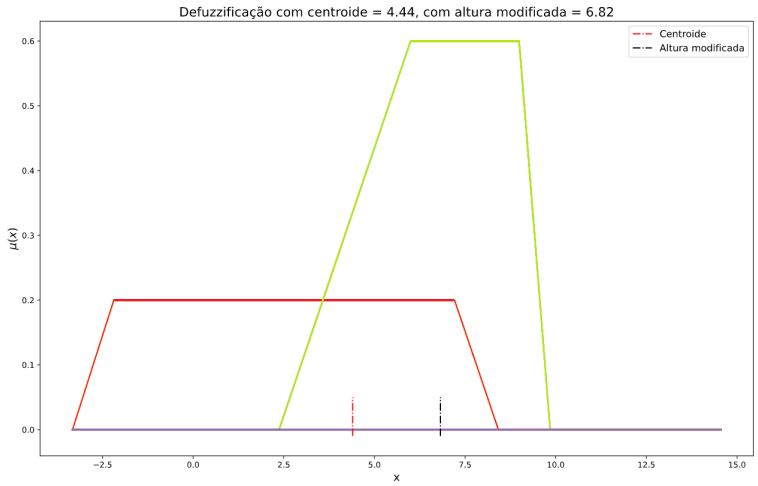

Figura 5.8: Comparação entre os métodos de defuzzificação do modelo [10]

Após a defuzzificação, ocorre a última etapa da previsão *multistep*, que não apresenta diferenças em relação aos outros modelos da literatura. Tratando-se de uma previsão multivariada para vários passos à frente, todas as variáveis precisam ser previstas e usadas como entrada no modelo, de forma iterativa, para inferir os n passos futuros predefinidos.

É importante ressaltar que o uso dos próprios dados de previsão, que já possuem um certo grau de erro, como entrada no modelo para prever passos futuros, introduz uma quantidade adicional de erros acumulados ao sistema. Portanto, é crucial ter um cuidado especial durante o módulo de Definição e Organização do Problema, principalmente na seleção de variáveis. Um aumento no número de séries apresentadas ao modelo não apenas aumenta a complexidade do problema, mas pode também resultar em uma maior quantidade de erros no sistema, afetando a interpretabilidade e a acurácia.

6
Estudos de casos

O objetivo deste estudo foi implementar modificações no e-AutoMFIS para avaliar se as alterações propostas resultariam em melhorias significativas, tanto em termos de acurácia quanto de interpretabilidade. Para fins de comparação, foram utilizada as mesmas bases de dados de referência (benchmark) empregadas em [10]: concentração de gases no ar [111], a competição M3 [112], taxas de tráfego terrestre e câmbio [113]. A Tabela 6.1 apresenta a dimensionalidade de cada base.

Tabela 6.1: Quantidade de séries e registros das bases usados nos estudos de caso

Base	**Registros**	**Variáveis**
Concentração de gases no ar	9357	12
Taxa de Tráfego Terrestre	17544	862
Competição M3	134	5
Taxa de Câmbio	7588	8

A experimentação realizada neste estudo é dividida em três etapas. Primeiramente, o modelo é otimizado pelo algoritmo NSGA-II, utilizando as métricas de acurácia e interpretabilidade como objetivos, para a seleção dos hiperparâmetros que melhor se adequem aos interesses do usuário, seja para obter um modelo mais preciso ou mais interpretável. Após a seleção da configuração, caso as condições de teste sejam as mesmas aplicadas em [10], os resultados da nova versão do modelo são comparados com sua versão anterior e com os demais modelos *benchmark* utilizados nos estudos de casos. Por fim, ambas as versões do modelo são avaliadas, fazendo-se uso das mesmas configurações de hiperparâmetros.

Os estudos de casos são apresentados em diferentes seções, detalhando o passo a passo do processo, fornecendo configurações, parâmetros e outras informações relevantes para a reprodução do modelo. A avaliação dos resultados considera a acurácia do modelo, utilizando as métricas pré-selecionadas pelos modelos *benchmark* e [10], e a interpretabilidade, utilizando as métricas já apresentadas anteriormente.

6.1 Concentração de Gases no Ar

Os dados apresentados foram coletados em [114], onde o objetivo era investigar a concentração de certos gases poluentes no ar, como CO, NO_2 e NO_X, utilizando dispositivos sensores de baixo custo para compará-los com as concentrações reais, obtidas por estações tradicionais de monitoramento de poluição do ar.

Como visto na Tabela 6.2, a base de dados consiste em um total de 12 variáveis, que incluem a concentração real dos gases medidas pelas estações de monitoramento, as correspondentes medidas pelos dispositivos de baixo custo, e outras variáveis de interesse, como temperatura e umidade. Os registros abrangem o período de 03/10/2004 a 04/04/2005, totalizando seis meses de dados, com medições a cada hora e um total de 9357 registros.

Tabela 6.2: Variáveis da base de dados Qualidade do ar

Variável	**Descrição**	**Unidade**
CO (GT)	Concentração média de CO	mg/m^3
PT08.S1 (CO)	Resposta horária média do sensor de CO	Óx. Est.
NMHC (GT)	Concentração média de hidrocarbonetos não-metais	$\mu g/m^3$
C6H6 (GT)	Concentração média de Benzeno	$\mu g/m^3$
PT08.S2 (NMHC)	Resposta horária média do sensor de NMHC	Titania
NOx (GT)	Concentração média de compostos nitrogenados	ppb.
PT08.S3 (NOx)	Resposta horária média do sensor de NOx	Óx. Tungs.
NO_2 (GT)	Concentração média de NO_2	$\mu g/m^3$
PT08.S4 (NO_2)	Resposta horária média do sensor NO_2	Óx. Tungs.
PT08.S5 (O_3)	Resposta horária média do sensor O_3	Óx. Índio
T	Temperatura	ºC
RH	Umidade relativa	%
AH	Umidade absoluta	-

A utilização dessa base de dados é fundamental para a avaliação do modelo devido à sua natureza multivariada e ao grande volume de dados disponíveis. A versão anterior do e-AutoMFIS demonstrou resultados satisfatórios, proporcionando previsões robustas. Portanto, é pertinente analisar o potencial de melhoria que a nova versão do modelo pode oferecer.

Os resultados apresentados neste estudo de caso, assim como no e-AutoMFIS, são focados na variável *target* CO (GT). As demais variáveis selecionadas também serão previstas, pois a previsão é realizada em múltiplos passos. Contudo, seus resultados não serão considerados na avaliação do modelo.

6.1.1 Avaliação do Modelo

Para efeitos de comparação entre os modelos em termos de acurácia, foram selecionadas as métricas de Erro Médio Absoluto (MAE) e Erro Médio Quadrático (RMSE), conforme apresentado nas Equações 6-1 e 6-2:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{6-1}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{6-2}$$

Aqui, $\hat{y}i$ representa o i-ésimo valor previsto pelo modelo, yi denota o i-ésimo valor real e N é a quantidade total de elementos. Conforme observado por [115], essas métricas são complementares, visto que o MAE atribui o mesmo peso a todos os erros, sem penalizar fortemente erros individuais muito grandes, enquanto o RMSE é mais sensível a erros extremos, penalizando-os de forma mais significativa.

Quanto às métricas de interpretabilidade para avaliação, serão utilizadas as mais tradicionais referentes a SIFs, conforme utilizado em [10]: quantidade de regras que compõe a base, a média de antecedentes por regra e a ativação média.

6.1.2 Processamento dos dados

Inicialmente, realizou-se uma análise geral das variáveis para verificar a possibilidade de exclusão de algumas delas antes da aplicação de técnicas de pré-processamento. Observou-se que algumas variáveis descreviam o mesmo tipo de informação, diferenciando-se apenas na forma como os dados foram capturados. Com o objetivo de evitar redundância e reduzir a dimensionalidade do problema, as variáveis redundantes (PT08.S1(CO), PT08.S2(NMHC), PT08.S3(NOx), PT08.S4(NO_2), PT08.S5(O_3)) foram removidas.

Após a pré-seleção das variáveis, foi realizada uma etapa de pré-processamento dos dados antes da seleção final das *features* a serem apresentadas ao modelo. Verificou-se uma quantidade significativa de valores nulos, conforme apresentado na Tabela 6.3. Para lidar com esses valores faltantes, optou-se pela técnica conhecida como *Backwards*, que preenche os valores faltantes com dados do passo seguinte. A escolha dessa técnica foi motivada pelo fato de os dados serem registrados de hora em hora, o que implica em uma variação suave entre as leituras consecutivas. Após o preenchimento dos dados, apenas a variável NMHC(GT) ainda apresentava casos de dados ausentes, totalizando 8126 valores faltantes. Portanto, essa variável também foi excluída da base final de treinamento.

Tabela 6.3: Quantidade dados faltantes por variável

Variável	**Qtd. de dados faltantes**
CO (GT)	1683
PT08.S1 (CO)	366
NMHC (GT)	8443
C6H6 (GT)	366
PT08.S2 (NMHC)	366
NOx (GT)	1639
PT08.S3 (NOx)	366
NO_2 (GT)	1642
PT08.S4 (NO_2)	366
PT08.S5 (O_3)	366
T	366
RH	366
AH	366

Posteriormente, deu-se início à seleção final das *features* que seriam apresentadas ao modelo, utilizando a correlação de Pearson das variáveis pré-selecionadas. O objetivo foi verificar o nível de relação linear entre as variáveis. Conforme observado no *Heatmap* de correlação apresentado na Figura 6.1, as variáveis que representam a concentração de gases na atmosfera exibem uma alta correlação entre si, enquanto as variáveis de temperatura e umidade têm uma correlação fraca. No entanto, isso não foi suficiente para excluí-las da seleção final, pois ainda poderia haver uma relação não linear entre elas.

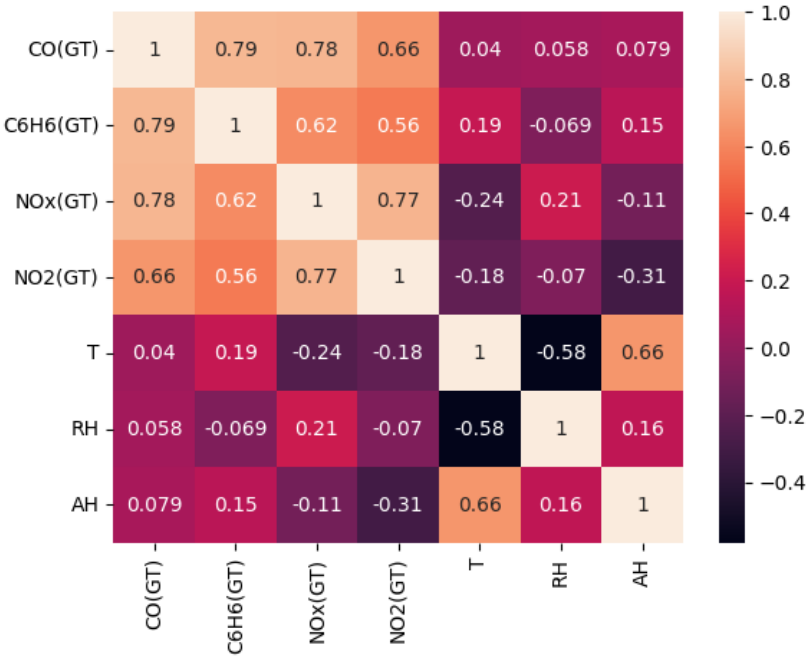

Figura 6.1: Gráfico *Heatmap* de correlação das variáveis pre-selecionadas do *dataset* de Qualidade do Ar

Nesta etapa também foi utilizada a técnica de seleção conhecida como *Recursive Feature Elimination* (RFE). Essa técnica opera removendo iterativamente as variáveis menos importantes de acordo com a importância atribuída por um modelo base. O modelo escolhido como base para o RFE foi o *Random Forest*, tendo como *target* a variável CO(GT). A Tabela 6.4 mostra o *ranking* de importância das variáveis, destacando mais uma vez que as variáveis de temperatura e umidade foram classificadas como menos importantes, reforçando o que já havia sido apresentado na análise dos coeficientes da correlação de Pearson.

Tabela 6.4: Ranking das variáveis segundo a metodologia RFE

Variáveis
NOx (GT)
C6H6 (GT)
NO_2 (GT)
T
RH
AH

Para permitir uma comparação direta entre as versões dos modelos avaliados, as mesmas variáveis selecionadas em [10] foram incluídas aqui. Assim, o grupo final de variáveis consistiu em: CO, NOx, C6H6, NO_2 e T. Outra etapa crucial do pré-processamento foi a separação dos dados para treinamento, validação, teste NSGAII e teste. A divisão dos dados e suas proporções podem ser visualizadas na Tabela 6.5.

Tabela 6.5: Separação dos dados de Concentração de Gases para treinamento e teste do modelo

Grupos	%
Treinamento	50
Validação	10
Teste (algoritmo)	20
Teste (modelo final)	20

6.1.3 Otimização de Hiperparâmetros

Na etapa de otimização dos hiperparâmetros, foram escolhidos 10 indivíduos por 50 gerações, totalizando uma amostragem máxima de 500 soluções. Para o parâmetro de cruzamento, foi empregada uma função personalizada da biblioteca *pymoo*, adequada para o tratamento de genes de indivíduos formados por variáveis mistas.

Outro aspecto importante na otimização do e-AutoMFIS2 é que nem todos os parâmetros foram ajustados utilizando o NSGAII. Testes preliminares indicaram que nem todos os exercem uma influência significativa nos resultados finais. Portanto, para facilitar a convergência do algoritmo e reduzir o custo computacional, foram otimizados apenas os parâmetros listados na Tabela 6.6, que apresentaram maior relevância nos resultados durante os testes preliminares.

Tabela 6.6: Parâmetros escolhidos para otimização

Parâmetros
N° máximo de antecedentes
Ativação mínima de premissas
N° de funções de Pertinência

A seleção dos melhores indivíduos foi realizada considerando o critério de equilíbrio entre interpretabilidade e acurácia, por meio da fronteira de Pareto formada após a execução do algoritmo genético. A Figura 6.2 ilustra o comportamento esperado de objetivos conflitantes, onde um menor número de regras resulta em modelos menos acurados, enquanto modelos mais precisos são associados a uma menor auditabilidade.

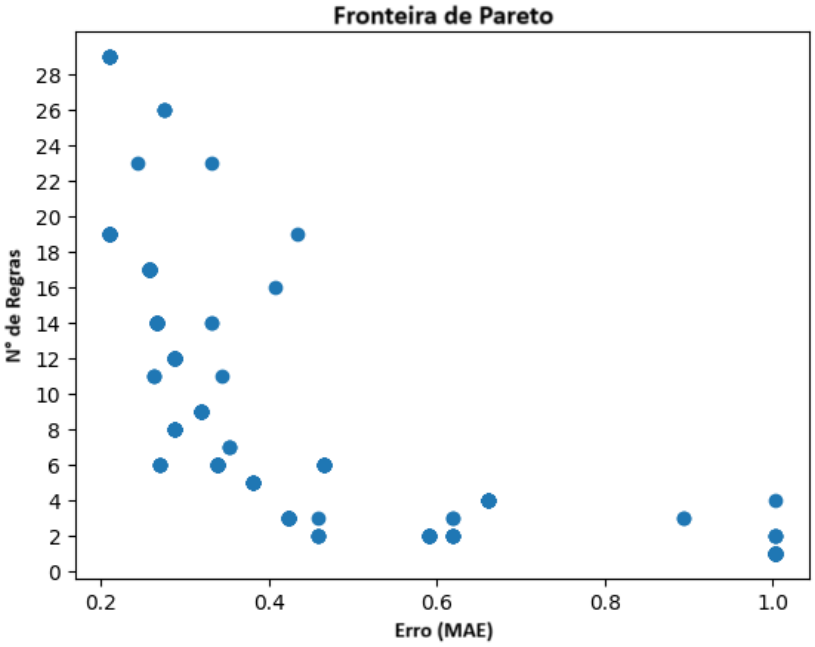

Figura 6.2: Fronteira de Pareto no estudo de caso de Concentração de Gases no Ar

Para o estudo de comparação entre as versões dos modelos, foi escolhida uma configuração que busca o equilíbrio entre acurácia e interpretabilidade, concentrando-se mais no centro do espaço de soluções. A Tabela 6.7 apresenta a solução selecionada aptidão.

Ao analisar os resultados, observa-se que o valor de aptidão demonstra um bom equilíbrio entre interpretabilidade e acurácia. Neste caso, apenas uma solução do espaço de amostragem foi selecionada para a etapa de comparações.

Tabela 6.7: Indivíduo escolhido do espaço de amostragem e seu valor *fitness* para o estudo de concentração de gases no ar

Indivíduo selecionado		Valor de aptidão	
Nº Máximo de Antecedentes	3	Acurácia (MAE)	0,209
Mínima Ativação de Premissas	0,595	Número de Regras	19
Número de Funções de Pertinência	3	Número de Antecedentes	1

6.1.4 Análise de Resultados

Após a seleção do melhor indivíduo do espaço de amostragem que melhor se adequa ao objetivo pretendido, o e-AutoMFIS2 é executado com os mesmos conjuntos de treino e validação, porém com um conjunto de teste novo para a avaliação final. Conforme mencionado anteriormente, a primeira comparação, quando aplicável, é realizada com os resultados da primeira versão do e-AutoMFIS e dos demais modelos *benchmark* apresentados em [10]. Nesse contexto, o e-AutoMFIS foi comparado com os modelos tradicionais citados no trabalho de [116], que incluem *Decision Tree*, *Random Forest*, *Adaboost*, *Support Vector Machine* (SVM), *Gated Recurrent Unit* (GRU) e *Long Short-term Memory* (LSTM).

O hiperparâmetros que pouco influem nos resultados finaisforam mantidos iguais aos melhores resultados apresentados em testes preliminares. A Tabela 6.8 apresenta os parâmetros e seus respectivos valores.

Tabela 6.8: Parâmetros testados no estudo de caso Qualidade do Ar

Parâmetro	**Configuração**
Lag	**24**
NumInput	**14**
NumPred	**60**
FuzzyMethod	**Agrupamento**
ActivationMethod	**Ativação média não-nula**
DefuzzMethod	**Altura ponderada**

As Tabelas 6.9 e 6.10 mostram, respectivamente, os resultados de acurácia e interpretabilidade do e-AutoMFIS2. Em ambos os casos, observa-se que a segunda versão do e-AutoMFIS continua sendo competitiva em comparação à sua versão original e aos demais modelos tradicionais em termos de precisão, ao mesmo tempo que apresenta uma interpretabilidade melhorada em relação ao seu antecessor.

Em um segundo teste, a versão 1 do eAutoMFIS foi executada com as mesmas configurações da versão 2 (vide Tabela 6.11). Nesse novo experimento, observa-se que as métricas de precisão são muito próximas, enquanto as métricas de interpretabilidade da versão 2 mostram-se superiores à sua antecessora. Isso era esperado, já que a aplicação da subamostragem semi-guiada faz com que as amostras selecionadas sejam mais correlacionadas, criando, assim, regras mais eficientes para a descrição e previsão do problema tratado.

Tabela 6.9: Resultado do estudo de caso Qualidade de Ar

Método	**Desempenho (MAE)**	**Desempenho (RMSE)**
Decision Tree	0,68	1,00
Random Forest	0,65	0,94
Adaboost	0,53	0,70
SVM	1,37	1,60
GRU	0,42	0,636
LSTM	0,42	0,635
e-AutoMFIS V1	0,50 ± 0,04	0,62 ± 0,2
e-AutoMFIS2	0,28 ± 0.02	0,34 ± 0,02

Tabela 6.10: Resultados das métricas de interpretabilidade das versões do e-AutoMfis no estudo de caso Concentração de Gases no Ar

e-AutoMFIS2		e-AutoMFIS V1	
Base de Regras	12,5 regras/série	Base de Regras	23,5 regras/série
N° de Antecedentes	1	N° de Antecedentes	2,1
Ativação Média	10,32 regras	Ativação Média	5,2 regras

Tabela 6.11: Resultados das versão usando os mesmos parâmetros no estudo de caso de concentração de gases

Métricas	e-AutoMFIS2	e-AutoMFIS V1
Acurácia (MAE)	0,28	0,31
Acurácia (RMSE)	0,34	0,38
Base de Regras	12,8	41,8
N° de Antecedentes	1 regras/série	1 regras/série
Ativação média	10,32 regras	36,78 regras

6.2 Taxa de Tráfego

O segundo estudo de caso diz respeito à taxa de ocupação de ruas na cidade de San Diego. As medições foram realizadas por sensores posicionados pelo Departamento de Transportes da Califórnia em vários locais. Os registros abrangem o período entre 2015 e 2016, totalizando 17544 registros e 862 séries, que representam os pontos das ruas analisadas. A base de dados utilizada foi disponibilizada publicamente por [117].

Como cada série da base de dados representa a taxa de ocupação de uma rua específica, a previsão de todas elas foi levada em conta. Portanto, em consonância com estudos anteriores realizados nesta base de dados, a performance do e-AutoMFIS2 é avaliada com base nas previsões de todas as séries que compõem a base, utilizando um horizonte de previsão de 24 passos à frente.

No experimento conduzido na primeira versão do e-AutoMFIS, a motivação para usar essa base de dados foi analisar como o modelo se comportava em relação à sua grande dimensionalidade e que apresentava padrões de curto prazo. O modelo obteve um desempenho satisfatório em termos de acurácia, gerando uma base de regras considerável.

6.2.1 Avaliação do Modelo

Em relação à precisão do modelo, a métrica escolhida será a RRSE (*Root Relative Squared Error*), utilizada tanto por [10] quanto por [117], definida pela Equação 6-3:

$$RRSE = \sqrt{\sum_{(i,t)\in\Omega_{test}} \frac{y_{i,t} - \hat{y_{i,t}}}{y_{i,t} - \bar{Y}}} \tag{6-3}$$

onde $\hat{y}i$ representa o i-ésimo valor previsto pelo modelo, yi é o i-ésimo valor atual e $\bar{Y}$ é a média dos valores reais. Esta métrica é interessante para avaliação, pois seus resultados são independentes da escala dos dados analisados, facilitando a comparação entre diferentes modelos. Quanto à interpretabilidade, serão usadas as mesmas métricas empregadas no primeiro estudo de caso.

6.2.2 Processamento dos Dados

Inicialmente, para manter a fidelidade com a versão original do experimento e o teste de comparação realizado com a primeira versão do e-AutoMFIS, os últimos 20% dos dados foram reservados para o teste do modelo. O restante dos dados foi dividido em três partes: treinamento (60%), validação (20%) e teste NSGAII (20%).

Após essa divisão, foi necessário utilizar um método para reduzir a dimensionalidade dos dados, visto que o uso completo das séries seria inviável devido ao custo computacional e provavelmente não geraria bons resultados. Para contornar essa questão, foi realizado um processo de agrupamento recursivo das séries com maior similaridade. O processo ocorre da seguinte maneira: a primeira série é selecionada, juntamente com as 5 outras séries que possuem maior similaridade com ela. Esse processo é repetido até que todas as séries estejam inseridas em um dos grupos formados. O uso desses grupos na modelagem do problema facilita a aprendizagem do modelo, pois favorece a captura de padrões e relações entre as séries.

Durante o processamento dos dados, foi identificado um padrão cíclico, conforme exemplificado na Figura 6.3, que mostra um recorte de duas semanas de uma das séries. Observam-se padrões em períodos diários (24 horas) e semanais (168 horas). Para explorar esses padrões, os dados foram segmentados em sete grupos distintos, cada um representando um dia da semana. Esta segmentação visa à formação de um conjunto de regras individualizado para cada dia, permitindo capturar de maneira mais precisa as particularidades dos padrões cíclicos diários e semanais presentes nos dados.

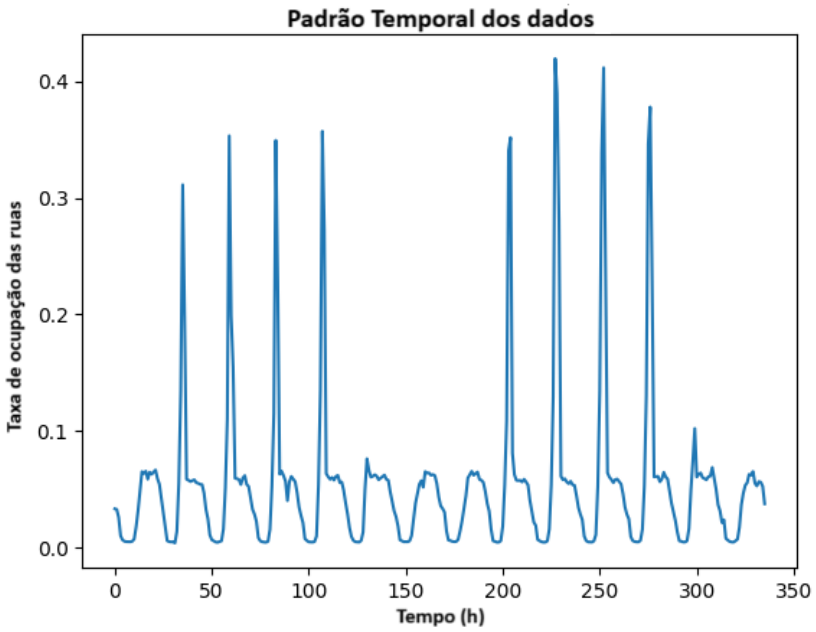

Figura 6.3: Componente cíclica de uma das séries apresentadas nos dados de taxa de tráfego

6.2.3 Otimização de hiperparâmetros

Após aplicar os processamentos considerados necessários nos dados, passamos para a etapa de otimização de hiperparâmetros. Devido à extensão dos dados, a opção escolhida foi considerar um conjunto menor de séries para reduzir o tempo de execução. A quantidade de indivíduos por população foi mantida em 10, enquanto o número de gerações foi então reduzido para 40.

Devido à extensão significativa da base de dados deste estudo, foram tomadas duas decisões adicionais para reduzir o custo computacional da execução do NSGAII. Primeiramente, como no estudo de caso anterior, optou-se por otimizar apenas o subconjunto de parâmetros apresentados na Tabela 6.6, de modoa facilitar o processo de convergência do modelo, mesmo com um espaço de busca reduzido.

A segunda decisão foi otimizar os parâmetros para apenas um dos grupos definidos na etapa de pré-processamento. Esse procedimento agiliza a definição da solução que será aplicada nas demais etapas do estudo de caso. Assim, a melhor configuração é utilizada para a previsão em todos os grupos de séries formados anteriormente.

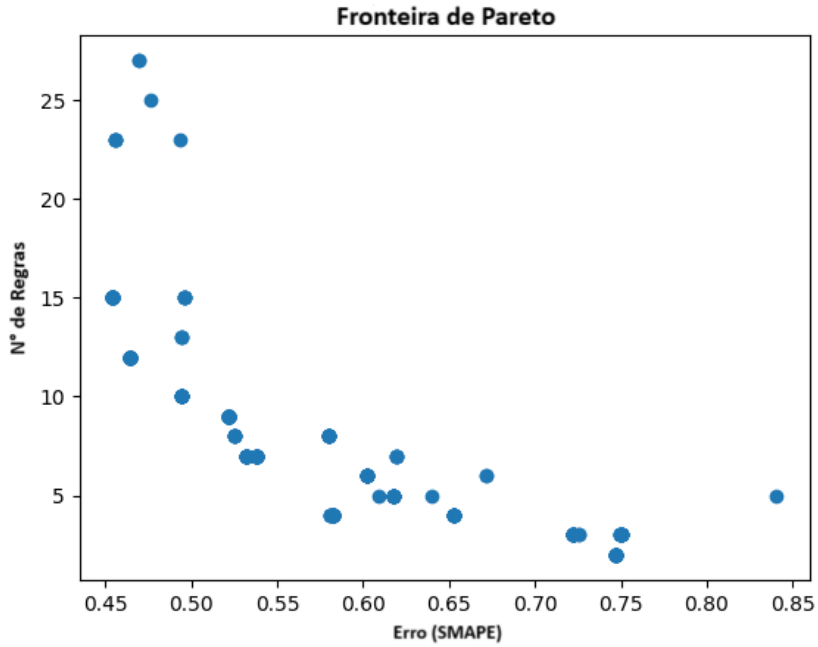

Figura 6.4: Fronteira de Pareto no estudo de caso de Taxa de Tráfego

Tabela 6.12: Indivíduos escolhidos do espaço de amostragem e seu valor *fitness* para o estudo de taxa de tráfego

Solução			Valor *fitness*	
Configuração 1	Nº Máx. de Regras	3	RRSE	0,454
	Mín. Ativação de Premissas	0,643	N° de Regras	105
	N° de Funções de Pertinência	7	N° de Antecedentes	1
Configuração 2	Nº Máx. de Regras	3	RRSE	0,521
	Mín. Ativação de Premissas	0,695	N° de Regras	63
	N° de Funções de Pertinência	9	N° de Antecedentes	1

A Figura 6.4 apresenta a fronteira de Pareto resultante do estudo de caso da taxa de tráfego. Novamente, é possível observar um comportamento esperado, onde a precisão do modelo é inversamente proporcional à sua interpretabilidade. O número de regras representa a média das bases de regras formadas para cada dia da semana.

Para a avaliação final do modelo, foram escolhidas duas configurações: a que obteve a melhor precisão no valor *fitness* e a segunda com um *fitness* mais equilibrado. A Tabela 6.12 apresenta ambos os indivíduos escolhidos e seus respectivos valores de *fitness*.

Quanto aos demais hiperparâmetros necessários para a execução do e-AutoMFIS2, assim como no estudo de caso anterior, utilizaram-se os mesmos parâmetros estabelecidos durante testes preliminares com o e-AutoMFIS2, conforme apresentado na Tabela 6.13.

Tabela 6.13: Parâmetros sem otimização usados no estudo de caso Taxa de ocupação de ruas

Parâmetro	**Melhor configuração**
Lag	**24**
NumInput	**14**
NumPred	**12**
FuzzyMethod	**Uniforme**
ActivationMethod	**Ativação média não-nula**
DefuzzMethod	**Altura ponderada**

6.2.4 Análise de Resultado

O primeiro experimento deste estudo de caso consistiu na comparação direta entre as duas versões do e-AutoMFIS. Este experimento foi motivado pelo uso de um conjunto reduzido de séries durante a seleção de hiperparâmetros, conforme mencionado anteriormente. Portanto, ambos os modelos foram treinados utilizando o mesmo subconjunto de séries empregado na etapa de otimização e sob as mesmas configurações. A Tabela 6.14 apresenta os resultados das duas configurações selecionadas para ambas as versões.

Tabela 6.14: Indivíduos escolhidos do espaço de amostragem e seu valor *fitness* para o estudo de taxa de tráfego

Configurações	Métricas	e-AutoMFIS2	e-AutoMFIS
1	Acurácia (RRSE)	**0,55**	**0,56**
	N° de Regras	**108**	**694**
	Comp. de Antecedentes	**1**	**2,28**
	Ativ. Média de Regras	**5,87**	**11,53**
2	Acurácia (RRSE)	**0,57**	**0,58**
	N° de Regras	**87**	**210**
	Comp. de Antecedentes	**1**	**2,09**
	Ativ. Média de Regras	**3,73**	**3,19**

Para o segundo teste, considerou-se a previsão da série completa para viabilizar a comparação dos resultados do e-AutoMFIS2 com os melhores resultados obtidos pela sua versão anterior e outras técnicas tradicionais de modelagem, conforme apresentado na Tabela 6.15. É importante mencionar que, para este teste, utilizou-ser apenas a primeira configuração selecionada, devido ao custo computacional necessário para testar todos os agrupamentos.

A Tabela 6.16 mostra os resultados de todos os modelos testados para todas as variáveis presentes na base de dados de taxa de ocupação de ruas. Observa-se que a nova versão não obteve grandes ganhos em relação à sua versão anterior, porém os resultados continuam competitivos em comparação com os modelos mais tradicionais. Quanto à interpretabilidade, diferentemente da precisão, os resultados (conforme apresentados na Tabela 6.17) demonstraram uma melhora substancial, com a redução significativa da base de regras, enquanto a quantidade de antecedentes e a média de ativação de regras continuaram a indicar um nível de simplicidade na interpretação dos resultados.

Tabela 6.15: Métodos usados para a base de dados Taxa de ocupação de ruas

Método	**Descrição**
AR	Modelo autorregressivo
LRidge	Modelo VAR (*Vector autoregression*) com regularização L2
LSVR	Modelo VAR com função de objetivo SVM, proposto em [118]
TRMF	Modelo autorregressivo usando *Temporal regularized matrix factorization*, proposto em [116]
VAR-MLP	Modelo híbrido usando Redes Neurais MLP (*Multilayer Perceptron*) e método ARIMA, proposto em [119]
RNN-GRU	Modelo RNN usando GRU (*Gated Recurrent Units*)
LSTNet-Skip	Modelo proposto por [113] usando camadas skip-RNN
LSTNet-Attn	Modelo proposto por [113] usando camadas de *Temporal Attention*

Tabela 6.16: Resultado obtido para a base de dados Taxa de ocupação de ruas

Método	**Resultado (RRSE)**
AR	0,62
LRidge	0,60
LSVR	0,59
TRMF	0,64
VAR-MLP	0,61
RNN-GRU	0,56
LSTNet-Skip	0,49
LSTNet-Attn	0,53
e-AutoMFIS	0,52 ± 0,07
e-AutoMFIS2	0,51 ± 0,04

Em ambos os testes, não houve melhora na acurácia do modelo. Uma possível explicação é a mudança na abordagem de subamostragem adotada no e-AutoMFIS2, que privilegia a previsão de um alvo específico, como no estudo de caso anterior. Quando todas as variáveis têm importância igual, essa abordagem pode ser prejudicial. Quanto à interpretabilidade, observa-se novamente uma melhora em relação à versão anterior, possivelmente devido à subamostragem semi-guiada, que, ao utilizar dados mais correlacionados, simplifica o processo de formação de regras.

Tabela 6.17: Resultados das métricas de interpretabilidade das versões do e-AutoMfis no estudo de caso Taxa de Tráfego

e-AutoMFIS V1		e-AutoMFIS2	
Base de Regras	210 regras/série	Base de Regras	96 regras/série
N° de Antecedentes	2,7	N° de Antecedentes	1
Ativação Média	3,5 regras	Ativação Média	4,57 regras

6.3 Competição M3

Para o terceiro estudo de caso, foi selecionada a base de dados usada na competição M3, que é conhecida por suas competições na área de previsão de séries temporais, abrangendo uma variedade de problemas em diferentes domínios. Isso a torna uma escolha interessante para avaliar o desempenho de novos modelos desenvolvidos. As séries selecionadas para a Competição M3 estão listadas na Tabela 6.18, juntamente com a quantidade de registros disponíveis em diferentes períodos de tempo.

Tabela 6.18: Informações sobre as séries da Competição M3

Séries	**Períodos**			
	Mensal	**Trimestral**	**Anual**	**Outros**
Finanças	58	76	145	29
Indústria	102	83	334	0
Micro	146	204	474	4
Macro	83	336	312	0
Demografia	245	57	111	0
Outros	1	0	52	141

A escolha dessa base de dados se deve, também, ao seu histórico de uso com as versões anteriores do modelo em estudo. Seguindo os estudos de caso anteriores, consideraarm-se especificamente dados mensais relacionados a finanças.

Em trabalhos anteriores, como o de [9], os dados foram submetidos a uma clusterização hierárquica e, em seguida, agrupados em 3 clusters distintos para análise. Posteriormente, o estudo conduzido por [10] concentrou-se no grupo em que o primeiro trabalho apresentou baixo desempenho, a fim de investigar se as modificações aplicadas ao modelo original foram benéficas. Portanto, este estudo segue a mesma abordagem de comparação do modelo proposto com sua versão anterior, utilizando um horizonte de previsão de 18 passos adiante.

6.3.1 Avaliação do Modelo

Para analisar a acurácia do modelo desenvolvido, foi adotada a métrica estabelecida para a avaliação da competição, o SMAPE (Symmetric Mean Absolute Percentage Error), definido pela Equação 6-4.

$$SMAPE = \frac{100\%}{n} \sum_{i=1}^{n} \frac{|\hat{y}_i - y_i|}{(|y_i| + |\hat{y}_i|)/2} \tag{6-4}$$

onde $\hat{y}i$ é o i-ésimo valor previsto pelo modelo, yi é o i-ésimo valor atual e n é a quantidade total de elementos. Quanto às métricas de interpretabilidade, seguimos os padrões tradicionalmente utilizados para modelos SIF, conforme observado em estudos de caso anteriores.

6.3.2 Processamento de Dados

Inicialmente, foi necessário dividir os conjuntos de dados para treinamento, validação, seleção de hiperparâmetros e avaliação final do modelo. Mantendo o horizonte de previsão definido anteriormente, os 18 últimos registros foram reservados para teste, enquanto os 36 registros anteriores foram divididos entre teste NSGAII e validação, respectivamente. O restante dos dados foi utilizado para a formação da base de regras.

Estabelecida a divisão, realizou-se uma inspeção visual das séries que compõem o *cluster*. Como é possível notar na Figura 6.5, todas as séries exibem uma tendência de crescimento com o passar do tempo. Essa tendência pode prejudicar a modelagem do e-AutoMFIS, já que a base de regras, uma vez formada, é estática. Portanto, a tendência foi removida das por meio da técnica de regressão linear.

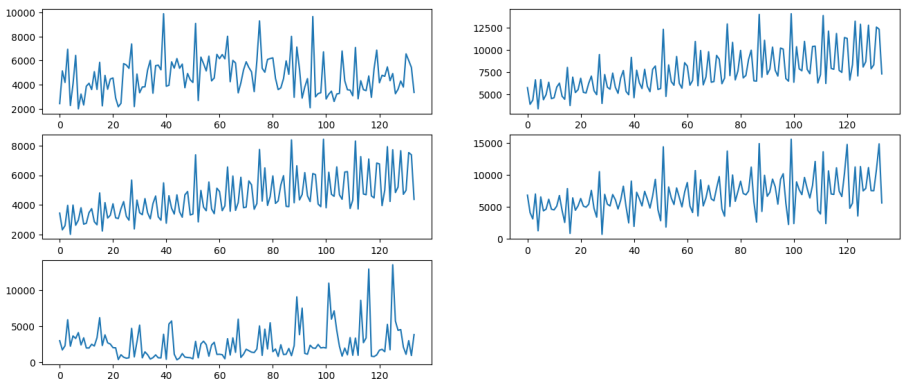

Figura 6.5: Registro histórico de 5 séries financeiras da base de dados Competição M3

Outra etapa importante é a validação da correlação entre as séries envolvidas na base de dados. Esta informação orientaa o agrupamento das séries e facilita a construção da base de regras. A Figura 6.6 mostra que as séries 1, 2 e 3 são as que possuem maior correlação entre si. Portanto, elas foram escolhidas para utilização nas próximas etapas.

Figura 6.6: Matriz de correlação das séries no estudo de caso Competição M3

6.3.3 Otimização de hiperparâmetros

Assim como nos estudos anteriores, os principais hiperparâmetros do modelo foram selecionados por meio do algoritmo NSGAII. Uma diferença importante em relação aos estudos anteriores é que os valores de *fitness* para a seleção dos melhores indivíduos foram baseados na média de três execuções para cada indivíduo. Isso foi possível devido ao tamanho relativamente menor da base de dados, tornando a execução do algoritmo menos custosa em termos computacionais e permitindo uma escolha mais robusta dos indivíduos. Outro ponto relevante é que, assim como nos demais estudos de casos, apenas os parâmetros apresentados na Tabela 6.6 foram otimizados pelo algoritmo, enquanto os demais parâmetros (vide Tabela 6.19) foram definidos em testes preliminares com o modelo.

Tabela 6.19: Parâmetros sem otimização usados no estudo de caso Competição M3

Parâmetro	Melhor configuração
Lag	**24**
NumInput	**12**
NumPred	**10**
FuzzyMethod	**Agrupamento**
ActivationMethod	**Ativação média não-nula**
DefuzzMethod	**Altura ponderada**

Analisando a fronteira de Pareto apresentada na Figura 6.7, observa-se que a variação da quantidade de regras gera ganhos mínimos em relação à métrica de precisão do modelo. Portanto, o uso dos indivíduos que apresentaram maiores bases de regras não resultaria em um ganho significativo de precisão.

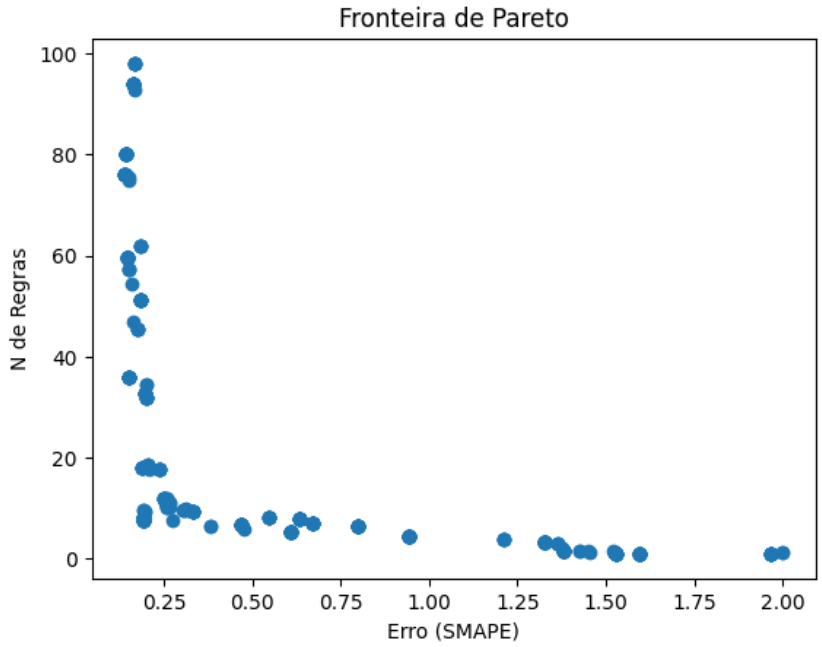

Figura 6.7: Fronteira de Pareto no estudo de caso de Competição M3

A Tabela 6.20 mostra os indivíduos escolhidos com seus respectivos valores de *fitness*. Observando essa última informação, notamos que a seleção dos indivíduos foi guiada pela observação feita anteriormente, optando por indivíduos mais equilibrados na relação precisão/interpretabilidade.

Tabela 6.20: Indivíduos escolhidos do espaço de amostragem e seu valor *fitness* para o estudo de caso competição M3

Solução			Valor *fitness*	
Configuração 1	Nº Máx. de Regras	3	SMAPE Médio	0,134
	Mín. Ativ. de Premissas	0,477	N° Médio de Regras	16,33
	N° de Funções de Pert.	3	N° Médio Antecedentes	1
Configuração 2	Nº Máx. de Regras	3	SMAPE Médio	0,174
	Mín. Ativ. de Premissas	0,469	N° Médio de Regras	14,33
	N° de Funções de Pert.	3	N° Médio Antecedentes	1

6.3.4 Análise de Resultados

Neste estudo de caso específico, as condições para a comparação com os resultados reportados em [10] não foram atendidas, pois, como mencionado anteriormente, nem todas as variáveis foram utilizadas na análise final. Portanto, as comparações entre os modelos serão realizadas exclusivamente com base nos resultados coletados usando os hiperparâmetros escolhidos na etapa anterior.

Para complementar o estudo, também foi utilizado o modelo estatístico clássico vetorial autoregressivo (VAR), conhecido por sua capacidade de lidar de forma satisfatória com problemas de séries temporais multivariadas. Sua modelagem baseia-se em uma função linear de cada variável e seus respectivos valores defasados, bem como os valores defasados de outras variáveis presentes na base. Essas características tornam o VAR um ótimo modelo para comparação com as duas versões do e-AutoMFIS.

A Tabela 6.21 apresenta os resultados dos três modelos utilizados para previsão. Observa-se que ambas as configurações das duas versões do e-AutoMFIS apresentaram desempenho melhor do que o modelo estatístico VAR, o que mostra novamente que o e-AutoMFIS2 pode competir com modelos mais clássicos. Ao comparar as versões do e-AutoMFIS entre si, observa-se que, novamente em termos de precisão, os modelos demonstraram estar muito próximos.

Em termos de interpretabilidade, os resultados da Tabela 6.22 mostram que a versão atual do modelo continua a apresentar resultados superiores aos do seu antecessor, reforçando que o uso dos valores de correlação e autocorrelação para orientar a subamostragem do modelo auxilia na geração de bases de regras mais compactas.

Tabela 6.21: Resultados das versões do e-AutoMFIS e o modelo VAR para a Competição M3

Modelos	SMAPE
e-AutoMFIS2 (Configuração 1)	0,148
e-AutoMFIS2 (Configuração 2)	0,153
e-AutoMFIS V1 (Configuração 1)	0,162
e-AutoMFIS V1 (Configuração 2)	0,177
Vector Autoregression	0.23

Tabela 6.22: Resultados das métricas de interpretabilidade das versões do e-AutoMfis no estudo de caso Competição M3

Configurações	e-AutoMFIS2		e-AutoMFIS V1	
1	Base de Regras	32,67	Base de Regras	54
	N° de Antecedentes	1	N° de Antecedentes	1
	Ativação Média	20,91	Ativação Média	43,56
2	Base de Regras	32	Base de Regras	56,67
	N° de Antecedentes	1	N° de Antecedentes	1
	Ativação Média	20,56	Ativação Média	37,22

6.4 Taxa de câmbio

Para o último estudo de caso deste trabalho, foi selecionada uma base de dados organizada por [117], que representa as variações da taxa de câmbio de oito países: Austrália, Grã-Bretanha, Canadá, Suíça, Japão, Nova Zelândia e Cingapura. Os dados foram coletados diariamente entre 1990 e 2016, totalizando 7588 registros. Para esta análise, foi utilizado um horizonte de previsão de 24 passos, conforme determinado em estudos anteriores, apresentados em [117] e [10]. A acurácia do modelo é novamente avaliada com base na previsão de todas as séries que compõem a base de dados.

A escolha desta base está relacionada ao desempenho da primeira versão do e-AutoMFIS, que enfrentou dificuldades na extração der informações de forma eficiente para a formação das regras e, consequentemente, para uma previsão mais acurada.

6.4.1 Avaliação do Modelo

Para avaliar a acurácia do modelo, será utilizada a métrica RRSE (Root Relative Squared Error), mantendo-se a consistência com o estudo de caso anterior. Essa escolha facilita a comparação com os resultados de [117] e [10].

Em relação à interpretabilidade, foram mantidas as métricas de tamanho de regras, quantidade de antecedentes e média de ativação de regras, conforme realizado nos experimentos anteriores.

6.4.2 Processamento de Dados

Os dados foram divididos em quatro grupos para evitar vazamentos de dados entre treinamento, validação e teste. A divisão ocorreu da seguinte maneira: 20% para o teste final do modelo, 20% para teste do algoritmo genético, 10% para validação e 50% para treinamento.

Como observado na Figura 6.8, é evidente que as séries exibem um comportamento estocástico, tornando desafiadora a extração de padrões ou relações entre as variáveis para formar uma base de regras apropriadas para previsão. Portanto, a primeira medida adotada foi a diferenciação das séries. Essa técnica é comum na área financeira e redefine o problema para a previsão da variação da taxa de câmbio. Além disso, a diferenciação remove os componentes não estacionários das séries, os quais são prejudiciais ao e-AutoMFIS2.

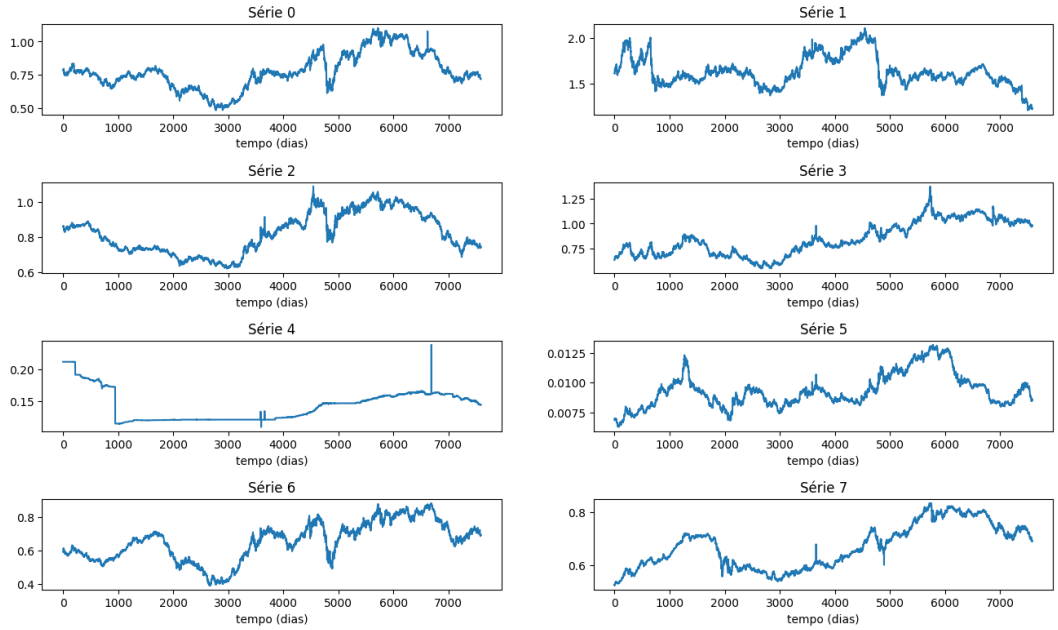

Figura 6.8: Comportamento das séries do estudo de caso - Taxa de Câmbio

Após a diferenciação das séries (Figura 6.9), observa-se a geração de alguns picos, que podem ser considerados *outliers*. Esses valores atípicos podem prejudicar a formação dos conjuntos e, consequentemente, prejudicar as etapas de formação de regras e previsão do modelo. Assim, foi empregada uma abordagem comum de tratamento de *outliers*, na qual os valores que, ao serem subtraídos pela média da série, excedem em mais de duas vezes o seu desvio padrão são substituídos pela média da série.

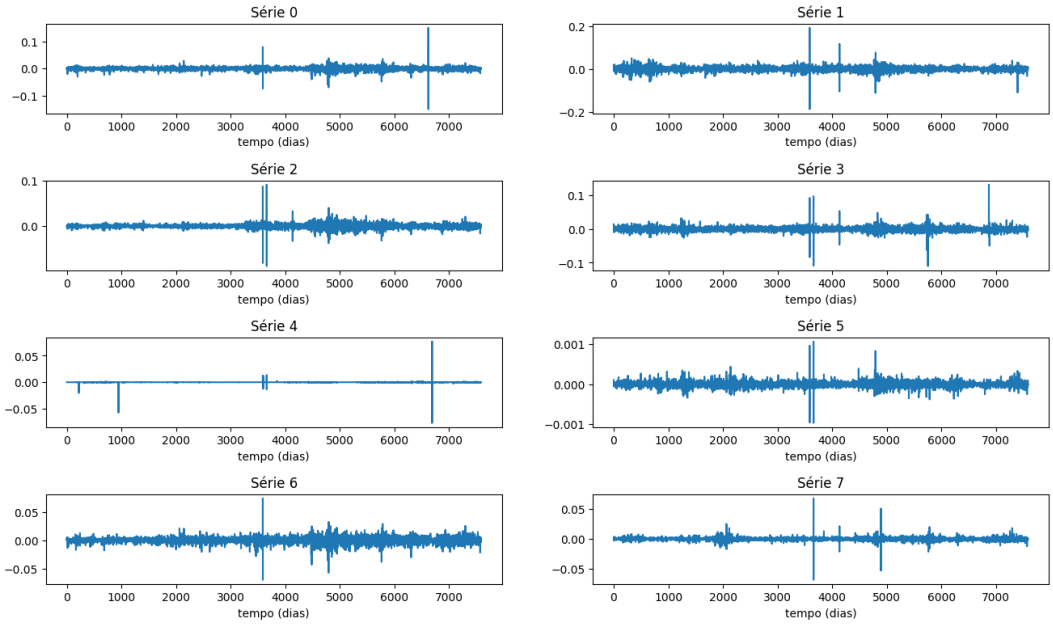

Figura 6.9: Comportamento das séries do estudo de caso - Taxa de Câmbio

Uma consequência dessas transformações nos dados foi o estreitamento da distribuição dos valores processados, o que poderia prejudicar a extração de conhecimento, privilegiando apenas o conjunto com maior densidade de valores. Para resolver esse problema, os dados foram normalizados usando a técnica z-score, conforme descrito na Equação 6-5. Essa abordagem ajusta o valor médio ($\bar{x}$) e o desvio padrão (σ) das variáveis, suavizando o estreitamento dos dados.

$$z_i = \frac{x_i - \bar{x}}{\sigma} \tag{6-5}$$

6.4.3 Otimização de hiperparâmetros

No processo de otimização dos hiperparâmetros do e-AutoMFIS2, foi mantido o mesmo número de indivíduos e gerações utilizados nos demais estudos de caso, ou seja, 10 indivíduos e 50 gerações. O NSGA-II foi configurado com o objetivo de minimizar a métrica de acurácia RRSE e, simultaneamente, minimizar o número de regras no modelo e o número médio de antecedentes por regra para avaliação da interpretabilidade.

Assim como nos demais casos apresentados, somente os parâmetros apresentados na Tabela 6.6 foram otimizados para evitar um aumento desnecessário na complexidade do processo de otimização. Os demais parâmetros da configuração do e-AutoMFIS2 foram definidos por testes iniciais com o modelo e são apresentados na Tabela 6.23.

Tabela 6.23: Parâmetros sem otimização usados no estudo de caso Taxa de Câmbio

Parâmetro	Melhor configuração
Lag	**24**
NumInput	**14**
NumPred	**12**
Fuzzy+Method	**Agrupamento**
ActivationMethod	**Ativação média não-nula**
DefuzzMethod	**Altura ponderada**

Analisando a fronteira de Pareto na Figura 6.10, observa-se que o espaço de busca de soluções é substancialmente menor do que a amostragem original de 500 soluções. Isso ocorre devido à predominância de soluções com bases de regras que possuem muitos consequentes iguais, levando a erros muito próximos entre elas.

Para selecionar o indivíduo que equilibra acurácia e interpretabilidade, avaliou-se a relação entre ganhos e perdas dessas métricas. O indivíduo com melhor acurácia (0,0177) foi descartado devido à sua base de regras extensa (178), que comprometeria a interpretabilidade. A escolha foi baseada na busca pelo melhor compromisso entre acurácia e interpretabilidade, resultando no indivíduo apresentado na Tabela 6.24.

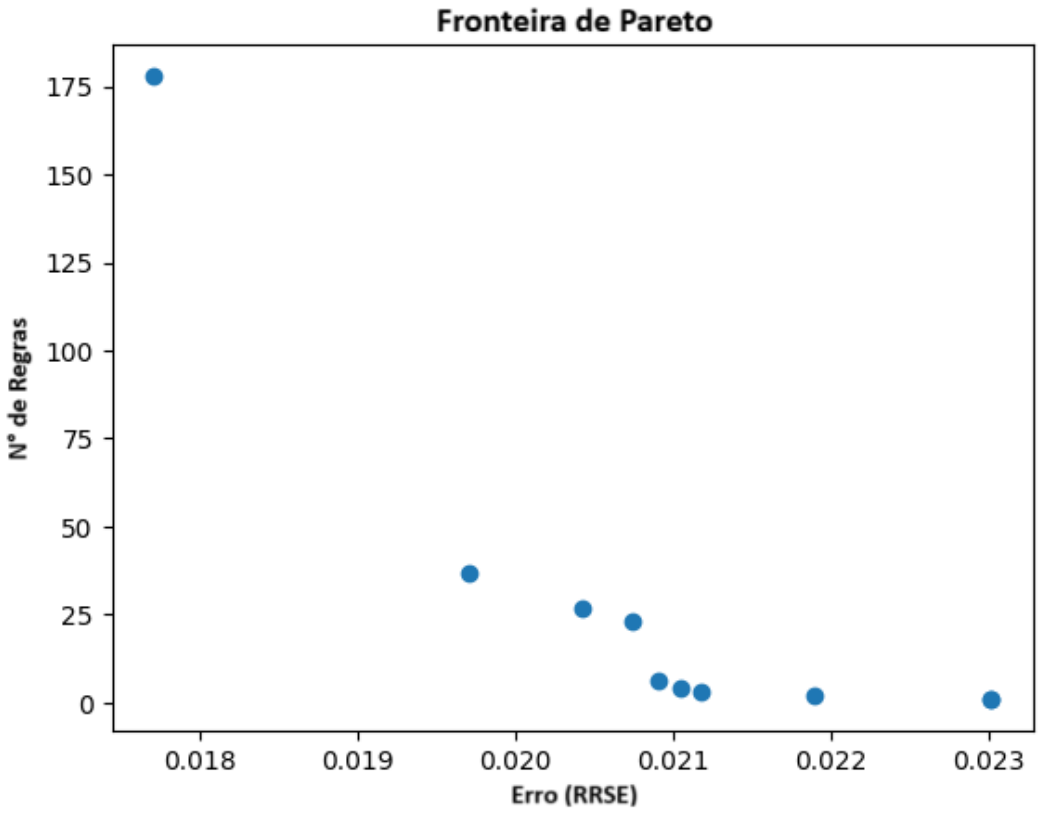

Figura 6.10: Fronteira de Pareto no estudo de caso de Taxa de Câmbio

Tabela 6.24: Indivíduo escolhido do espaço de amostragem e seu valor *fitness* para o estudo de concentração de gases no ar

Solução		Valor *fitness*	
Nº Máximo de Antecedentes	3	Acurácia (MAE)	0,0208
Mínima Ativação de Premissas	0,646	Número de Regras	23
Número de Funções de Pertinência	9	Número de Antecedentes	1

6.4.4 Análise de Resultados

Antes de apresentar os resultados, é importante destacar uma modificação no processo de treinamento do modelo. Durante a avaliação das previsões do modelo no conjunto de teste, foi observado que as métricas de acurácia deterioravam significativamente após algumas etapas de previsão. Para mitigar essa questão, foi decidido que o modelo seria retreinado utilizando os dados de teste já previstos, a fim de recalibrar a base de regras e melhorar a acurácia das previsões subsequentes.

Neste processo de retreinamento, os dados de treino e validação do modelo foram combinados com os dados de teste já previstos, respeitando a ordem cronológica dos registros, para formar uma nova base de treinamento. Dessa nova base, os últimos 24 registros eram reservados para constituir o novo conjunto de validação do modelo.

Outro aspecto relevante a ser destacado diz respeito às métricas que serão apresentadas. Embora o processo de previsão do e-AutoMFIS2 tenha sido conduzido em partes, o cálculo do erro foi realizado considerando os dados de teste como um todo. Dessa forma, a possibilidade de comparar os resultados obtidos com a versão anterior do e-AutoMFIS, conforme apresentado no estudo de caso em [10], foi preservada. Por outro lado, as métricas de interpretabilidade aqui apresentadas representam a média de cada base que foi gerada durante a execução do modelo.

Como essa base é proveniente de [117], assim como o estudo de caso da taxa de tráfego, os métodos utilizados para comparação são detalhados na tabela 6.15. A tabela 6.25 apresenta o desempenho destes métodos e da primeira versão do e-AutoMFIS, juntamente com o resultado do e-AutoMFIS2.

Tabela 6.25: Métodos usados na base de dados Taxa de Câmbio

Método	**Resultado (RRSE)**
AR	0,0445
LRidge	0,0675
LSVR	0,0662
TRMF	0,0563
VAR-MLP	0,0578
RNN-GRU	0,0626
LSTNet-Skip	0,0449
LSTNet-Attn	0,0590
e-AutoMFIS V1	0,0613 ± 0,0081
e-AutoMFIS2	0,0298 ± 0,0008

Analisando os resultados, observa-se que o e-AutoMFIS2 conseguiu superar não apenas o seu antecessor, mas também os melhores resultados obtidos para esta base. Ao verificar as bases de conhecimento geradas, constata-se que o problema relatado em [10], no qual a maioria dos consequentes gerados tinham valores iguais, foi contornado em alguns casos durante a etapa de otimização de hiperparâmetros. Isso sugere que uma busca mais profunda pela melhor configuração em conjunto com o retreinamento periódico foram capazes de contornar esse problema.

Em termos de interpretabilidade, conforme apresentado na tabela 6.26, pode-se observar que, como nos demais estudos de caso, as métricas mostraram melhoras em relação ao seu antecessor. Como mencionado anteriormente, a subamostragem semi-guiada, ao apontar dados mais correlacionados, gera regras mais simples e bases de regras menores.

Tabela 6.26: Resultados das métricas de interpretabilidade das versões do e-AutoMfis no estudo de caso de Taxa de Câmbio

e-AutoMFIS2		e-AutoMFIS V1	
Base de Regras	29,5 regras/série	Base de Regras	58 regras/série
N° de Antecedentes	1	N° de Antecedentes	3,5
Ativação Média	12,24 regras	Ativação Média	4,1 regras

No segundo teste, a primeira versão do e-AutoMFIS foi usada com os mesmos critérios e configuração usados na versão 2. Como mostra a tabela 6.27, a precisão das duas versões são bem próximas, reforçando novamente que a busca por uma melhor configuração pode levar a uma melhora substancial da acurácia para ambas as versões do e-AutoMFIS. Em relação à interpretabilidade, as alterações no modo de subamostragem do modelo facilitaram o processo de formação de regras ao gerar amostras que já possuíam uma correlação.

Tabela 6.27: Resultados das versão usando os mesmos parâmetros no estudo de caso de taxa de câmbio

Métricas	e-AutoMFIS2	e-AutoMFIS V1
RRSE	0,0298	0,0322
Base de Regras	29,5	80,5
N° de Antecedentes	1 regras/série	1,03 regras/série
Ativação média	12,91 regras	13,34 regras

6.5 Considerações sobre os estudos de casos

Após a execução dos quatro estudos de casos, é possível fazer algumas considerações em relação a essa nova versão do modelo. Primeiramente, em comparação com métodos mais tradicionais, o e-AutoMFIS2 é altamente competitivo em termos de precisão, inclusive no estudo de caso da taxa de câmbio, onde a primeira versão do e-AutoMFIS não demonstrou resultados tão eficazes. Ao comparar as duas versões do modelo, observa-se que as alterações tiveram impacto em ambas as frentes de avaliação, resultando em modelos mais precisos com auditabilidade mantida.

Em relação ao e-AutoMFIS2, os estudos de caso evidenciaram uma questão importante. Em alguns cenários, nos quais todas as variáveis têm importância igual na previsão, a abordagem de subamostragem pode não ser ideal. Isso ocorre porque ela utiliza os valores de correlação entre as séries para definir a amostra, e em alguns casos, a combinação das variáveis pode ser benéfica para uma variável específica, mas não para as demais, o que pode prejudicar o desempenho geral do modelo.

Um ponto relevante a ser destacado é que, assim como no caso do seu antecessor e-AutoMFIS, o uso direto do modelo, na maioria das vezes, não é a abordagem ideal para aproveitar todo o seu potencial. Devido às particularidades de cada série, é necessário um conhecimento prévio para tratar os problemas de forma mais específica, por meio do pre-processamento e análise dos dados. Em caso de necessidade, uma intervenção na forma do treinamento do modelo, como foi feito no último estudo de caso apresentado, pode ser benéfica.

7
Conclusão e trabalhos futuros

Os modelos de explicabilidade e interpretabilidade (XAI) têm ganhado destaque devido à sua capacidade de fornecer resultados com alta acurácia, aliada à capacidade de serem auditáveis. Dentre esses modelos, o e-AutoMFIS demonstrou um grande potencial ao competir com modelos estabelecidos na literatura, embora ainda seja um ponto inicial na jornada para tornar a síntese automática de sistemas *Fuzzy* mais robusta. Com essa consideração, este trabalho propôs alterações na estrutura do e-AutoMFIS para aprimorá-lo ainda mais em termos de acurácia e interpretabilidade.

Para alcançar essa evolução, duas mudanças foram implementadas na arquitetura do modelo: a otimização de hiperparâmetros via algoritmos genéticos e a subamostragem semi-guiada com base nos valores de correlação cruzada e autocorrelação da base de dados. Posteriormente, quatro conjuntos de dados analisados anteriormente foram utilizados para validar as alterações. O e-AutoMFIS2 foi comparado com seu antecessor em termos de acurácia e interpretabilidade, além de ser comparado apenas em acurácia com outros modelos tradicionais.

No que diz respeito à acurácia, o modelo proposto apresentou resultados competitivos em relação aos proporcionados por aqueles com os quais foi comparado, demonstrando que a complexidade do modelo não é um fator determinante para a obtenção de bons resultados. Quanto à interpretabilidade, o e-AutoMFIS2 apresentou, em geral, resultados superiores aos do seu antecessor, gerando bases de regras mais concisas, com menos antecedentes e uma média menor de ativação de regras.

Um ponto relevante observado nos testes foi que, ao usar as mesmas configurações de hiperparâmetros definidas pelo NSGA-II no e-AutoMFIS, os resultados de acurácia foram muito próximos aos do modelo proposto neste trabalho, sugerindo que a busca por hiperparâmetros de forma otimizada foi fundamental para a melhoria da acurácia no modelo. No entanto, ao comparar as métricas de interpretabilidade, o e-AutoMFIS2 manteve uma vantagem sobre seu antecessor na maioria dos casos, indicando que a subamostragem semi-guiada foi o principal fator influenciador dessa melhoria, uma vez que essa técnica é exclusiva do e-AutoMFIS2.

Os resultados destacam que o e-AutoMFIS2 conseguiu explorar o potencial de crescimento de seu antecessor, melhorando ambos os aspectos analisados. Como esperado, o uso do NSGA-II para a otimização dos hiperparâmetros mostrou-se superior à busca exaustiva, pois explorou de forma mais eficiente o espaço de busca, identificando uma fronteira de Pareto de soluções ótimas de acordo com os objetivos do usuário. Além disso, a subamostragem semi-guiada facilitou o processo de formação de regras ao criar subamostras com valores correlacionados, tanto entre séries quanto em termos de defasagem, facilitando a extração de conhecimento e a formação da base de regras.

Apesar desses pontos positivos, é importante ressaltar que ainda é necessário realizar uma preparação e análise cuidadosas dos dados para garantir o bom desempenho do modelo. Por exemplo, no processo de otimização, é recomendável utilizar apenas um subconjunto de hiperparâmetros para otimização, a fim de reduzir o custo computacional e manter a convergência do modelo. Além disso, a definição de alguns parâmetros com base na análise dos dados pode ser mais indicada na maioria dos casos.

Quanto a possíveis aprimoramentos, o e-AutoMFIS2 apresenta áreas que podem ser exploradas. Um desses pontos é que todos os métodos de correlações utilizadas neste trabalho são lineares, não tendo sido exploradas relações não-lineares entre as séries. Assim, considerar métodos não-lineares poderá ser benéfico ao desenvolvimento modelo.

Outro ponto para exploração é o fato de que todos os parâmetros são padronizados para todas as séries apresentadas ao modelo, mas cada série pode se beneficiar de tratamentos personalizados, como o número de conjuntos *Fuzzy* por variável, pois, devido as particularidade de cada série, o número de conjuntos *Fuzzy* ótimos pode variar entre elas. Isso poderia potencializar a acurácia do modelo sem comprometer sua interpretabilidade.

Por fim, uma transformação efetiva do e-AutoMFIS2 em um modelo adaptável, capaz de lidar com mudanças nos dados de forma iterativa, pode ser benéfica. Isso foi observado durante o estudo de caso da taxa de câmbio, onde uma mudança no comportamento dos dados foi solucionada com a geração de novas regras a partir de dados mais recentes, permitindo ao modelo se adaptar a esse novo comportamento. Portanto, tornar o modelo compatível com o aprendizado em fluxo contínuo pode ser útil para lidar com a questão das bases de regras estáticas.

Referências bibliográficas

[1] BECK, N.; KATZ, J. N.. **Modeling dynamics in time-series–cross-section political economy data**. Annual review of political science, 14:331–352, 2011.

[2] TSAY, R. S.. **Analysis of financial time series**. John wiley & sons, 2005.

[3] TOULOUMI, G.; ATKINSON, R.; TERTRE, A. L.; SAMOLI, E.; SCHWARTZ, J.; SCHINDLER, C.; VONK, J. M.; ROSSI, G.; SAEZ, M.; RABSZENKO, D. ; OTHERS. **Analysis of health outcome time series data in epidemiological studies**. Environmetrics: The official journal of the International Environmetrics Society, 15(2):101–117, 2004.

[4] ZEDNIK, C.. **Solving the black box problem: A normative framework for explainable artificial intelligence**, 2021.

[5] MESTIKOU, M. A.; SMETI, K. E. ; HACHAICHI, Y.. **Artificial intelligence and machine learning in financial services market developments and financial stability implications**, 2017.

[6] TJOA, E.; GUAN, C.. **A survey on explainable artificial intelligence (xai): Toward medical xai**. IEEE transactions on neural networks and learning systems, 32(11):4793–4813, 2020.

[7] MONTAVON, G.; SAMEK, W. ; MÜLLER, K.-R.. **Methods for interpreting and understanding deep neural networks**. 2018.

[8] RIBEIRO, M. T.; SINGH, S. ; GUESTRIN, C.. **Nothing else matters: Model-agnostic explanations by identifying prediction invariance**. arXiv preprint arXiv:1611.05817, 2016.

[9] COUTINHO, J. R.. **Automfis: Fuzzy inference system for multivariate time series forecasting**. In: 2016 IEEE INTERNATIONAL CONFERENCE ON FUZZY SYSTEMS (FUZZ-IEEE), p. 2120–2127. IEEE, 2016.

[10] CARVALHO, T. M.. **e-AutoMFIS: Modelo interpretável para previsão de séries multivariadas usando comitês de Sistemas de Inferência Fuzzy**. PhD thesis, 2021.

[11] DEB, C.; ZHANG, F.; YANG, J.; LEE, S. E. ; SHAH, K. W.. **A review on time series forecasting techniques for building energy consumption**. Renewable and Sustainable Energy Reviews, 74:902–924, 2017.

[12] KATRIS, C.. **A time series-based statistical approach for outbreak spread forecasting: Application of covid-19 in greece**. Expert systems with applications, 166:114077, 2021.

[13] LILA, M. F.; MEIRA, E. ; OLIVEIRA, F. L. C.. **Forecasting unemployment in brazil: A robust reconciliation approach using hierarchical data**. Socio-Economic Planning Sciences, 82:101298, 2022.

[14] MONTGOMERY, D. C.; JENNINGS, C. L. ; KULAHCI, M.. **Introduction to time series analysis and forecasting**. John Wiley & Sons, 2015.

[15] ZHANG, G. P.. **Time series forecasting using a hybrid arima and neural network model**. Neurocomputing, 50:159–175, 2003.

[16] YATISH, H.; SWAMY, S.. **Recent trends in time series forecasting–a survey**. International Research Journal of Engineering and Technology (IRJET), 7(04):5623–5628, 2020.

[17] PANT, A.; RAJPUT, R.. **Time Series Analysis of Gold Price Using R**. 10 2019.

[18] POUZOLS, F. M.; LENDASSE, A.. **Effect of different detrending approaches on computational intelligence models of time series**. p. 1–8, 2010.

[19] GUEDES, H. A. S.; PRIEBE, P. D. S. ; MANKE, E. B.. **Tendências em séries temporais de precipitação no norte do estado do rio grande do sul, brasil**. Revista Brasileira de Meteorologia, 34:283–291, 2019.

[20] ADHIKARI, R.; AGRAWAL, R. K.. **An introductory study on time series modeling and forecasting**. arXiv preprint arXiv:1302.6613, 2013.

[21] CHATFIELD, C.. **Time-series forecasting**. Chapman and Hall/CRC, 2000.

[22] GRANGER, C. W. J.; NEWBOLD, P.. **Forecasting economic time series**. Academic press, 2014.

[23] SWANSON, N. R.; WHITE, H.. **A model-selection approach to assessing the information in the term structure using linear models and artificial neural networks**. Journal of Business & Economic Statistics, 13(3):265–275, 1995.

[24] LAWRENCE, M. J.; EDMUNDSON, R. H. ; O'CONNOR, M. J.. **The accuracy of combining judgemental and statistical forecasts**. Management Science, 32(12):1521–1532, 1986.

[25] VAN DEN BROEKE, M.; DE BAETS, S.; VEREECKE, A.; BAECKE, P. ; VANDERHEYDEN, K.. **Judgmental forecast adjustments over different time horizons**. Omega, 87:34–45, 2019.

[26] GRANGER, C. W.; JEON, Y.. **Long-term forecasting and evaluation**. International journal of forecasting, 23(4):539–551, 2007.

[27] RAEESI, M.; MESGARI, M. ; MAHMOUDI, P.. **Traffic time series forecasting by feedforward neural network: a case study based on traffic data of monroe**. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 40:219–223, 2014.

[28] DE ANDRADE, L. C. M.; DA SILVA, I. N.. **Very short-term load forecasting based on arima model and intelligent systems**. In: 2009 15TH INTERNATIONAL CONFERENCE ON INTELLIGENT SYSTEM APPLICATIONS TO POWER SYSTEMS, p. 1–6. IEEE, 2009.

[29] POTTER, C. W.; NEGNEVITSKY, M.. **Very short-term wind forecasting for tasmanian power generation**. IEEE Transactions on power systems, 21(2):965–972, 2006.

[30] MADDALA, G. S.; LAHIRI, K.. **Introduction to econometrics**, volumen 2. Macmillan New York, 1992.

[31] R ROSCA, E.. **Stationary and non-stationary time series**. The USV Annals of Economics and Public Administration, 10(1):177–186, 2011.

[32] DICKEY, D. A.; FULLER, W. A.. **Distribution of the estimators for autoregressive time series with a unit root**. Journal of the American statistical association, 74(366a):427–431, 1979.

[33] THINH, D. T.; QUAN, N. B. H. ; MANEETIEN, N.. **Implementation of moving average filter on stm32f4 for vibration sensor application**. In: 2018 4TH INTERNATIONAL CONFERENCE ON GREEN TECHNOLOGY AND SUSTAINABLE DEVELOPMENT (GTSD), p. 627–631. IEEE, 2018.

[34] CORTÉS-IBÁÑEZ, J. A.; GONZÁLEZ, S.; VALLE-ALONSO, J. J.; LUENGO, J.; GARCÍA, S. ; HERRERA, F.. **Preprocessing methodology for time series: An industrial world application case study**. Information sciences, 514:385–401, 2020.

[35] RAUDYS, A.; LENČIAUSKAS, V. ; MALČIUS, E.. **Moving averages for financial data smoothing**. In: INFORMATION AND SOFTWARE TECHNOLOGIES: 19TH INTERNATIONAL CONFERENCE, ICIST 2013, KAUNAS, LITHUANIA, OCTOBER 2013. PROCEEDINGS 19, p. 34–45. Springer, 2013.

[36] PEDERSEN, T. M.. **Alternative linear and non-linear detrending techniques: A comparative analysis based on euro-zone data**. In: MONOGRAPHS OF OFFICIAL STATISTICS, DOCUMENT DEL THIRD EUROSTAT COLLOQUIUM ON MODERN TOOLS FOR BUSINESS CYCLE ANALYSIS, LUXEMBURGO, EUROSTAT, p. 51–85, 2004.

[37] KOHNS, D.; BHATTACHARJEE, A.. **Developments on the bayesian structural time series model: trending growth**, 2020.

[38] BONABEAU, E.; DORIGO, M. ; THERAULAZ, G.. **Swarm intelligence: from natural to artificial systems**. Oxford university press, 1999.

[39] BROWN, D. E.; HUNTLEY, C. L.. **A practical application of simulated annealing to clustering**. Pattern recognition, 25(4):401–412, 1992.

[40] NARA, K.; HAYASHI, Y.; IKEDA, K. ; ASHIZAWA, T.. **Application of tabu search to optimal placement of distributed generators**. In: 2001 IEEE POWER ENGINEERING SOCIETY WINTER MEETING. CONFERENCE PROCEEDINGS (CAT. NO. 01CH37194), volumen 2, p. 918–923. IEEE, 2001.

[41] LAN, S.; FAN, W.; YANG, S.; PARDALOS, P. M. ; MLADENOVIC, N.. **A survey on the applications of variable neighborhood search algorithm in healthcare management**. Annals of mathematics and artificial intelligence, p. 1–35, 2021.

[42] KUMAR, V.; KUMAR, D.. **An astrophysics-inspired grey wolf algorithm for numerical optimization and its application to engineering design problems**. Advances in Engineering Software, 112:231–254, 2017.

[43] KATOCH, S.; CHAUHAN, S. S. ; KUMAR, V.. **A review on genetic algorithm: past, present, and future**. Multimedia tools and applications, 80:8091–8126, 2021.

[44] PACHECO, M. A. C.; OTHERS. **Algoritmos genéticos: princípios e aplicações**. ICA: Laboratório de Inteligência Computacional Aplicada. Departamento de Engenharia Elétrica. Pontifícia Universidade Católica do Rio de Janeiro. Fonte desconhecida, 28, 1999.

[45] MAULIK, U.. **Medical image segmentation using genetic algorithms**. IEEE Transactions on information technology in biomedicine, 13(2):166–173, 2009.

[46] SHAYEGH BOROUJENI, H.; MOGHADAM CHARKARI, N.; BEHROUZIFAR, M. ; TAHERI MAKHSOOS, P.. **Tracking multiple variable-sizes moving objects in lfr videos using a novel genetic algorithm approach**. In: KNOWLEDGE TECHNOLOGY: THIRD KNOWLEDGE TECHNOLOGY WEEK, KTW 2011, KAJANG, MALAYSIA, JULY 18-22, 2011. REVISED SELECTED PAPERS, p. 143–153. Springer, 2012.

[47] AHUACTZIN, J. M.; TALBI, E.-G.; BESSIERE, P. ; MAZER, E.. **Using genetic algorithms for robot motion planning**. In: GEOMETRIC REASONING FOR PERCEPTION AND ACTION: WORKSHOP GRENOBLE, FRANCE, SEPTEMBER 16–17, 1991 SELECTED PAPERS, p. 84–93. Springer, 1993.

[48] XHAFA, F.; SUN, J.; BAROLLI, A.; BIBERAJ, A. ; BAROLLI, L.. **Genetic algorithms for satellite scheduling problems**. Mobile Information Systems, 8(4):351–377, 2012.

[49] BAJER, D.; MARTINOVIĆ, G. ; BREST, J.. **A population initialization method for evolutionary algorithms based on clustering and cauchy deviates**. Expert Systems with Applications, 60:294–310, 2016.

[50] DE MELO, V. V.; DELBEM, A. C. B.. **Investigating smart sampling as a population initialization method for differential evolution in continuous problems**. Information Sciences, 193:36–53, 2012.

[51] OTERO, J. A. B.. **Algoritmos genéticos aplicados à solução do problema inverso biomagnético**. Master's thesis, 2016.

[52] GOLDBERG, D. E.; DEB, K.. **A comparative analysis of selection schemes used in genetic algorithms**. In: FOUNDATIONS OF GENETIC ALGORITHMS, volumen 1, p. 69–93. Elsevier, 1991.

[53] SANTANA, I. C.. **Métodos de seleção em algoritmos genéticos inspirados em torneios esportivos= Selection operators in genetic algorithms inspired by sport tournaments**. PhD thesis, [sn], 2017.

[54] TAVARA, E. G. M.. **ALGORITMO GENÉTICO MULTIOBJETIVO NA PREDIÇÃO DE ESTRUTURAS PROTEICAS NO MODELO HIDROFÓBICO-POLAR**. PhD thesis, 2012.

[55] JIMENEZ, J. D. T.. **Computacional para Localização de Projéteis de Armas de Fogo inseridos no Corpo Humano, por meio de Medições Magnéticas de Alta Sensibilidade**. PhD thesis, PUC-Rio, 2017.

[56] ANUMAK COMPANY. **How to use genetic algorithm for hyperparameter tuning in ml models**, 2022. ACessado em: 9 de Jan. 2024.

[57] SHEKAR, B.; DAGNEW, G.. **Grid search-based hyperparameter tuning and classification of microarray cancer data**. In: 2019 SECOND INTERNATIONAL CONFERENCE ON ADVANCED COMPUTATIONAL AND COMMUNICATION PARADIGMS (ICACCP), p. 1–8. IEEE, 2019.

[58] BELETE, D. M.; HUCHAIAH, M. D.. **Grid search in hyperparameter optimization of machine learning models for prediction of hiv/aids test results**. International Journal of Computers and Applications, 44(9):875–886, 2022.

[59] JOY, T. T.; RANA, S.; GUPTA, S. ; VENKATESH, S.. **Fast hyperparameter tuning using bayesian optimization with directional derivatives**. Knowledge-Based Systems, 205:106247, 2020.

[60] ALBAHLI, S.. **Efficient hyperparameter tuning for predicting student performance with bayesian optimization**. Multimedia Tools and Applications, p. 1–25, 2023.

[61] DI FRANCESCOMARINO, C.; DUMAS, M.; FEDERICI, M.; GHIDINI, C.; MAGGI, F. M.; RIZZI, W. ; SIMONETTO, L.. **Genetic algorithms**

for hyperparameter optimization in predictive business process monitoring. Information Systems, 74:67–83, 2018.

[62] ELGELDAWI, E.; SAYED, A.; GALAL, A. R. ; ZAKI, A. M.. **Hyperparameter tuning for machine learning algorithms used for arabic sentiment analysis**. In: INFORMATICS, volumen 8, p. 79. MDPI, 2021.

[63] GEORGE, T.; AMUDHA, T.. **Genetic algorithm based multi-objective optimization framework to solve traveling salesman problem**. In: ADVANCES IN COMPUTING AND INTELLIGENT SYSTEMS: PROCEEDINGS OF ICACM 2019, p. 141–151. Springer, 2020.

[64] MOUSAVI-AVVAL, S. H.; RAFIEE, S.; SHARIFI, M.; HOSSEINPOUR, S.; NOTARNICOLA, B.; TASSIELLI, G. ; RENZULLI, P. A.. **Application of multi-objective genetic algorithms for optimization of energy, economics and environmental life cycle assessment in oilseed production**. Journal of Cleaner Production, 140:804–815, 2017.

[65] JAIMES, A. L.; MARTÍNEZ, S. Z.; COELLO, C. A. C. ; OTHERS. **An introduction to multiobjective optimization techniques**. Optimization in Polymer Processing, 1:29, 2009.

[66] PADHYE, N.; DEB, K.. **Multi-objective optimisation and multi-criteria decision making in sls using evolutionary approaches**. Rapid Prototyping Journal, 17(6):458–478, 2011.

[67] CASANOVA, C.; SCHAB, E.; PRADO, L. ; ROTTOLI, G. D.. **Hierarchical clustering-based framework for a posteriori exploration of pareto fronts: application on the bi-objective next release problem**. Frontiers in Computer Science, 5:1179059, 2023.

[68] DAS, I.; DENNIS, J. E.. **Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems**. SIAM journal on optimization, 8(3):631–657, 1998.

[69] ÖZKALE, C.; FIĞLALI, A.. **Evaluation of the multiobjective ant colony algorithm performances on biobjective quadratic assignment problems**. Applied Mathematical Modelling, 37(14-15):7822–7838, 2013.

[70] LALWANI, S.; SINGHAL, S.; KUMAR, R. ; GUPTA, N.. **A comprehensive survey: Applications of multi-objective particle swarm optimi-**

zation (mopso) algorithm. Transactions on combinatorics, 2(1):39–101, 2013.

[71] LI, B.; LI, J.; TANG, K. ; YAO, X.. **Many-objective evolutionary algorithms: A survey**. ACM Computing Surveys (CSUR), 48(1):1–35, 2015.

[72] KONAK, A.; COIT, D. W. ; SMITH, A. E.. **Multi-objective optimization using genetic algorithms: A tutorial**. Reliability engineering & system safety, 91(9):992–1007, 2006.

[73] FONSECA, C. M.; FLEMING, P. J. ; OTHERS. **Genetic algorithms for multiobjective optimization: formulationdiscussion and generalization.** In: ICGA, volumen 93, p. 416–423. Citeseer, 1993.

[74] HORN, J.; NAFPLIOTIS, N. ; GOLDBERG, D. E.. **A niched pareto genetic algorithm for multiobjective optimization**. In: PROCEEDINGS OF THE FIRST IEEE CONFERENCE ON EVOLUTIONARY COMPUTATION. IEEE WORLD CONGRESS ON COMPUTATIONAL INTELLIGENCE, p. 82–87. Ieee, 1994.

[75] HAJELA, P.; LIN, C. Y.. **Genetic search strategies in multicriterion optimal design**. Structural optimization, 4:99–107, 1992.

[76] MURATA, T.; ISHIBUCHI, H. ; OTHERS. **Moga: multi-objective genetic algorithms**. In: IEEE INTERNATIONAL CONFERENCE ON EVOLUTIONARY COMPUTATION, volumen 1, p. 289–294. IEEE Piscataway, 1995.

[77] SRINIVAS, N.; DEB, K.. **Muiltiobjective optimization using nondominated sorting in genetic algorithms**. Evolutionary computation, 2(3):221–248, 1994.

[78] DEB, K.; PRATAP, A.; AGARWAL, S. ; MEYARIVAN, T.. **A fast and elitist multiobjective genetic algorithm: Nsga-ii**. IEEE transactions on evolutionary computation, 6(2):182–197, 2002.

[79] VERMA, S.; PANT, M. ; SNASEL, V.. **A comprehensive review on nsga-ii for multi-objective combinatorial optimization problems**. IEEE access, 9:57757–57791, 2021.

[80] LEWIS, F. L.; TIM, W. K.; WANG, L.-Z. ; LI, Z.. **Deadzone compensation in motion control systems using adaptive fuzzy logic control**. IEEE Transactions on Control Systems Technology, 7(6):731–742, 1999.

[81] NEDELJKOVIC, I.. **Image classification based on fuzzy logic**. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 34(30):3–7, 2004.

[82] ISMAIL, Z.; EFENDI, R. ; DERIS, M. M.. **Application of fuzzy time series approach in electric load forecasting**. New Mathematics and Natural Computation, 11(03):229–248, 2015.

[83] MULUDI, K.; SYARIF, A.; WANTORO, A. ; OTHERS. **Implementation of fuzzy-based model for prediction of prostate cancer**. In: JOURNAL OF PHYSICS: CONFERENCE SERIES, volumen 1751, p. 012041. IOP Publishing, 2021.

[84] LUCAS, P. O.; ORANG, O.; SILVA, P. C.; MENDES, E. ; GUIMARAES, F. G.. **A tutorial on fuzzy time series forecasting models: Recent advances and challenges**. Learning and Nonlinear Models, 19(2):29–50, 2022.

[85] SONG, Q.; CHISSOM, B. S.. **Fuzzy time series and its models**. Fuzzy sets and systems, 54(3):269–277, 1993.

[86] DE LIMA, P. C.; OTHERS. **Scalable models for probabilistic forecasting with fuzzy time series**. 2019.

[87] LIU, H.; HUSSAIN, F.; TAN, C. L. ; DASH, M.. **Discretization: An enabling technique**. Data mining and knowledge discovery, 6:393–423, 2002.

[88] HUARNG, K.. **Effective lengths of intervals to improve forecasting in fuzzy time series**. Fuzzy sets and systems, 123(3):387–394, 2001.

[89] YE, F.; ZHANG, L.; ZHANG, D.; FUJITA, H. ; GONG, Z.. **A novel forecasting method based on multi-order fuzzy time series and technical analysis**. Information Sciences, 367:41–57, 2016.

[90] SINGH, P.; BORAH, B.. **Forecasting stock index price based on m-factors fuzzy time series and particle swarm optimization**. International Journal of Approximate Reasoning, 55(3):812–833, 2014.

[91] HUARNG, K.; YU, T. H.-K.. **The application of neural networks to forecast fuzzy time series**. Physica A: Statistical mechanics and its applications, 363(2):481–491, 2006.

[92] YOLCU, O. C.; LAM, H.-K.. **A combined robust fuzzy time series method for prediction of time series**. Neurocomputing, 247:87–101, 2017.

[93] GAUTAM, S. S.; ABHISHEKH. **A novel moving average forecasting approach using fuzzy time series data set**. Journal of Control, Automation and Electrical Systems, 30:532–544, 2019.

[94] SULANDARI, W.; YUDHANTO, Y.. **Forecasting trend data using a hybrid simple moving average-weighted fuzzy time series model**. In: 2015 INTERNATIONAL CONFERENCE ON SCIENCE IN INFORMATION TECHNOLOGY (ICSITECH), p. 303–308. IEEE, 2015.

[95] TSENG, F.-M.; TZENG, G.-H.; YU, H.-C. ; YUAN, B. J.. **Fuzzy arima model for forecasting the foreign exchange market**. Fuzzy sets and systems, 118(1):9–19, 2001.

[96] SONG, Q.; CHISSOM, B. S.. **Forecasting enrollments with fuzzy time series—part i**. Fuzzy sets and systems, 54(1):1–9, 1993.

[97] SONG, Q.; CHISSOM, B. S.. **Forecasting enrollments with fuzzy time series—part ii**. Fuzzy sets and systems, 62(1):1–8, 1994.

[98] CHEN, S.-M.. **Forecasting enrollments based on fuzzy time series**. Fuzzy sets and systems, 81(3):311–319, 1996.

[99] YU, H.-K.. **Weighted fuzzy time series models for taiex forecasting**. Physica A: Statistical Mechanics and its Applications, 349(3-4):609–624, 2005.

[100] ALONSO, J. M.; CASTIELLO, C. ; MENCAR, C.. **Interpretability of fuzzy systems: Current research trends and prospects**. Springer handbook of computational intelligence, p. 219–237, 2015.

[101] PEDRYCZ, W.; GOMIDE, F.. **An introduction to fuzzy sets: analysis and design**. MIT press, 1998.

[102] MENCAR, C.. **Interpretability of fuzzy systems**. In: INTERNATIONAL WORKSHOP ON FUZZY LOGIC AND APPLICATIONS, p. 22–35. Springer, 2013.

[103] ANGELOV, P. P.; BUSWELL, R. A.. **Automatic generation of fuzzy rule-based models from data by genetic algorithms**. Information Sciences, 150(1-2):17–31, 2003.

[104] KONDRATENKO, Y. P.; KOZLOV, A. V.. **Generation of rule bases of fuzzy systems based on modified ant colony algorithms**. Journal of Automation and Information Sciences, 51(3), 2019.

[105] RONG, H.-J.; YANG, Z.-X.; WONG, P. K.; VONG, C. M. ; ZHAO, G.-S.. **Self-evolving fuzzy model-based controller with online structure and parameter learning for hypersonic vehicle**. Aerospace Science and Technology, 64:1–15, 2017.

[106] TIKK, D.; GEDEON, T. D. ; WONG, K. W.. **A feature ranking algorithm for fuzzy modelling problems**. In: INTERPRETABILITY ISSUES IN FUZZY MODELING, p. 176–192. Springer, 2003.

[107] TALAVERA, L.. **Feature selection as a preprocessing step for hierarchical clustering**. In: ICML, volumen 99, p. 389–397, 1999.

[108] PARK, C. H.. **A feature selection method using hierarchical clustering**. In: MINING INTELLIGENCE AND KNOWLEDGE EXPLORATION: FIRST INTERNATIONAL CONFERENCE, MIKE 2013, TAMIL NADU, INDIA, DECEMBER 18-20, 2013. PROCEEDINGS, p. 1–6. Springer, 2013.

[109] E SILVA, P. C. D. L.; JUNIOR, C. A. S.; ALVES, M. A.; SILVA, R.; COHEN, M. W. ; GUIMARÃES, F. G.. **Forecasting in non-stationary environments with fuzzy time series**. Applied Soft Computing, 97:106825, 2020.

[110] KÜFFNER, R.; PETRI, T.; WINDHAGER, L. ; ZIMMER, R.. **Fuzzy effect calculation example.** 2 2013.

[111] LIU, Z.; HU, B.; WANG, L.; WU, F.; GAO, W. ; WANG, Y.. **Seasonal and diurnal variation in particulate matter (pm 10 and pm 2.5) at an urban site of beijing: analyses from a 9-year study**. Environmental Science and Pollution Research, 22(1):627–642, 2015.

[112] MAKRIDAKIS, S.; ANDERSEN, A.; CARBONE, R.; FILDES, R.; HIBON, M.; LEWANDOWSKI, R.; NEWTON, J.; PARZEN, E. ; WINKLER, R.. **The accuracy of extrapolation (time series) methods: Results of a forecasting competition**. Journal of forecasting, 1(2):111–153, 1982.

[113] LAI, G.; CHANG, W.-C.; YANG, Y. ; LIU, H.. **Modeling long-and short-term temporal patterns with deep neural networks**. In: THE 41ST INTERNATIONAL ACM SIGIR CONFERENCE ON RESEARCH & DEVELOPMENT IN INFORMATION RETRIEVAL, p. 95–104, 2018.

[114] DE VITO, S.; PIGA, M.; MARTINOTTO, L. ; DI FRANCIA, G.. **Co, no2 and nox urban pollution monitoring with on-field calibrated electronic nose by automatic bayesian regularization**. Sensors and Actuators B: Chemical, 143(1):182–191, 2009.

[115] CHAI, T.; DRAXLER, R. R.. **Root mean square error (rmse) or mean absolute error (mae)**. Geoscientific model development discussions, 7:1525–1534, 2014.

[116] YU, H.-F.; RAO, N. ; DHILLON, I. S.. **Temporal regularized matrix factorization for high-dimensional time series prediction**. Advances in neural information processing systems, 29:847–855, 2016.

[117] WANG, Y.; LIAO, W. ; CHANG, Y.. **Gated recurrent unit network-based short-term photovoltaic forecasting**. Energies, 11(8):2163, 2018.

[118] VAPNIK, V.; GOLOWICH, S. E. ; SMOLA, A. J.. **Support vector method for function approximation, regression estimation and signal processing**. In: ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS, p. 281–287, 1997.

[119] ZHANG, G. P.. **Time series forecasting using a hybrid arima and neural network model**. Neurocomputing, 50:159–175, 2003.