

# Aprimoramentos de um Sistema Fuzzy-Genético para Análise de Séries Temporais

Marco Antônio da C. Ferreira, Adriano S. Koshiyama, Marley M.B.R. Vellasco e Ricardo Tanscheit  
Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro  
Rua Marquês de São Vicente, 225 – Gávea/Rio de Janeiro, RJ  
Email: [macfe, adriano, marley, ricardo]@ele.puc-rio.br

**Resumo**—Este trabalho trata de um Sistema Fuzzy-Genético aplicado à previsão de séries temporais, denominado de Genetic Programming Fuzzy Inference System for Forecasting problems (GPFIS-Forecast). O modelo GPFIS-Forecast difere da abordagem convencional em Sistemas Fuzzy-Genéticos pois busca uma maior integração com a meta-heurística: Programação Genética Multigênica. Além disso, facilita a inclusão de operadores t-norma, t-conorma, negação e modificadores linguísticos, assim como utiliza métodos de associação e operadores de agregação. Para avaliar a abordagem proposta, foram usadas as competições NN3 e M3 como base para os experimentos realizados. Os resultados demonstram a potencialidade do modelo GPFIS-Forecast, destacando-se a compatibilidade das bases de regras criadas.

**Keywords**—Sistemas Fuzzy-Genéticos, Análise de Séries Temporais e Programação Genética

## I. INTRODUÇÃO

Diariamente são encontradas situações de incerteza sobre o valor de um ativo ou gestão de um processo em um dado instante de tempo futuro (um mês à frente, por exemplo). Uma forma de antecipar ou mitigar essa incerteza de forma sistemática é fazer uso de modelos de previsão. A abordagem tradicional evoca o uso de modelos estocásticos, como Holt-Winters, Auto-Regressivos Integrados de Médias Móveis (ARIMA), etc. [16], [18], cuja metodologia de estimação tende a ser restritiva quanto às características da série temporal sob análise, requisitando em alguns casos estacionariedade e normalidade do processo [10]. Modelos inteligentes de previsão, como Redes Neurais e Programação Genética [13], [20] são capazes de fornecer soluções eficazes quando o uso de modelos estatísticos é inviável ou quando estes fornecem resultados pouco acurados.

Busca-se, com os modelos acima referidos, previsões acuradas, negligenciando-se a extração de conhecimento da série temporal sob análise. Este conhecimento é relevante em áreas como Econometria [10] e Ecologia [30], por exemplo, em que informações sobre como os momentos passados da série temporal afetam o presente são vitais para a compreensão do impacto de uma medida econômica, ou sobre a ocorrência de um evento natural. Sistemas Fuzzy para Previsão (SFP) [23] surgem como uma abordagem que fornece acurácia e também interpretabilidade linguística dos resultados obtidos. No âmbito dos SFP, empregam-se comumente modelos Neuro-Fuzzy [17], [29] e Fuzzy-Genéticos [7], [28].

No que tange aos Sistemas Fuzzy-Genéticos (SFGs) para Previsão, a meta-heurística genética é usada para encontrar os parâmetros de um SFP (quantidades relativas às funções de

pertinência, base de regras, etc.), que em geral se centram em um Sistema Fuzzy do Tipo Takagi-Sugeno [1], [4], [8], [11], [12]. Em consequência, tais SFGs dão pouca importância à questão da interpretabilidade das bases de regras. Além disso, procedimentos para seleção de consequentes, assim como para ponderação de regras não têm recebido foco especial.

Este trabalho apresenta aprimoramentos e novas aplicações do modelo Genetic Programming Fuzzy Inference System for Forecasting problems (GPFIS-Forecast), um SFG para previsão baseado em Programação Genética Multigênica (PGMG) [15], [27]. Este SFG foi proposto em [19], e, dada a sua característica ampla de atuação em problemas de Classificação, Regressão e Controle, pouco se investiu em aprimorar as componentes ligadas à Previsão de Séries Temporais. Portanto, este trabalho propõe aumentar a quantidade de possibilidades de configurações para o GPFIS-Forecast, avaliando a sua contribuição tanto para a acurácia dos resultados obtidos como para a interpretabilidade da base de regras.

Este trabalho está organizado da seguinte forma: a seção II discorre sobre a Programação Genética Multigênica, meta-heurística evolutiva na qual o GPFIS-Forecast está baseado. A seção III apresenta as etapas para elaboração do GPFIS-Forecast: fuzzificação, inferência, defuzzificação e avaliação. A seção IV apresenta os experimentos realizados, seus resultados e discussões, e a seção V conclui o artigo.

## II. PROGRAMAÇÃO GENÉTICA MULTIGÊNICA

A Programação Genética (PG) [20], [24] é uma técnica da Computação Evolutiva inspirada nos conceitos de seleção natural e recombinação genética. Permite sintetizar programas de computador que solucionem problemas sem a necessidade de serem programados para tal. A PGMG pode ser encarada como uma generalização da PG tradicional, pois denota um indivíduo como um complexo de estrutura em árvores, que, da mesma forma que na PG, recebe um conjunto de terminais  $X_{ij}$  (atributos em reconhecimento de padrões, defasagens de séries temporais, etc.), buscando predizer a saída  $Y_i$ .

A representação da PGMG é similar à da PG no tocante à estrutura em árvore. Contudo, um indivíduo para a PGMG é um complexo de estruturas em árvore, conforme Figura 1.

Cada árvore desta estrutura é uma solução parcial para o problema. É fácil ver que quando  $D = 1$ , a PGMG reduz-se à solução obtida por uma PG clássica. Os processos de avaliação e de seleção são efetuados de forma similar ao realizado na PG Clássica. Com relação aos operadores de recombinação, a operação de mutação na PGMG é similar à da PG clássica.

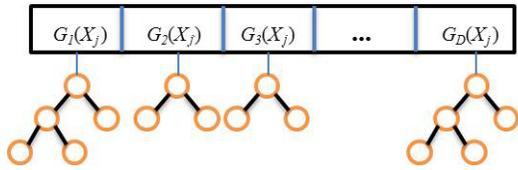


Figura 1. Exemplo de um indivíduo multigênico.

Para o cruzamento, é necessário definir em que nível – baixo ou alto – esta operação é realizada. A Figura 2b apresenta um indivíduo multigênico com cinco equações ( $D = 5$ ) efetuando uma operação de cruzamento de baixo nível, enquanto a Figura 2a apresenta a operação de mutação.

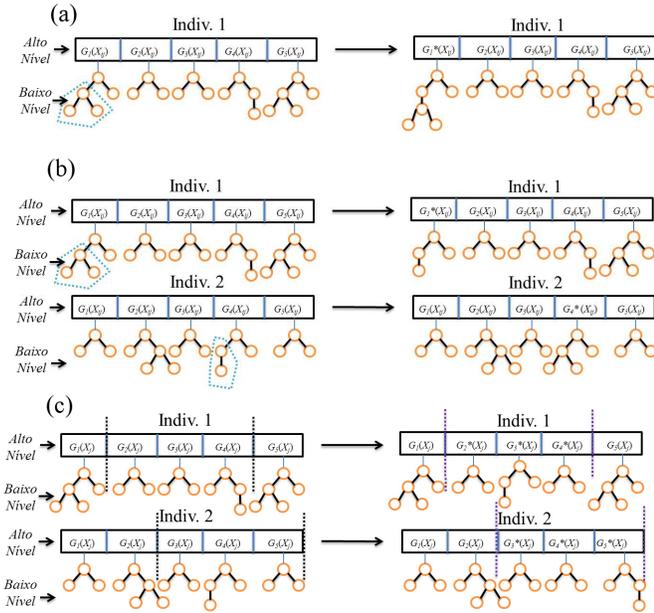


Figura 2. Operadores: (a) mutação, (b) cruzamento de baixo nível e (c) cruzamento de alto nível.

O baixo nível é o espaço onde é possível manipular as estruturas (Terminais e Operações Matemáticas) das equações presentes em um indivíduo. No caso, tanto a mutação quanto o cruzamento de baixo nível na PGMG são semelhantes ao efetuado na PG clássica.

Um exemplo de cruzamento de alto nível para um indivíduo com cinco equações é apresentado na Figura 2c. O alto nível é o espaço em que se manipulam de forma macro as equações presente no indivíduo. Logo, verifica-se que, a partir de dois pontos aleatórios, são permutadas equações de um indivíduo para o outro.

A próxima seção trata do modelo GPFIS-Forecast. Neste, a PGMG é usada como meta-heurística para formação da base de regras, de modo que cada árvore presente nos indivíduos decodifica uma regra (mais especificamente somente os termos antecedentes da regra). Cada indivíduo fornece o seu conjunto de regras, de maneira que o modelo explora métodos e operadores para refinar a solução expressa em cada codificação.

### III. MODELO GPFIS-FORECAST

O modelo GPFIS-Forecast é um SFG do tipo Pittsburgh [14]. O primeiro passo para a sua aplicação é o mapeamento de valores precisos em graus de pertinência a conjuntos fuzzy (Fuzzificação). Inicia-se, então, o processo de Inferência Fuzzy, dividido em 3 subetapas: (i) geração das premissas das regras (Formulação); (ii) união de uma premissa a um termo consequente (Associação) e (iii) agregação dos graus de ativação de cada regra fuzzy (Agregação). Finalmente, a Defuzzificação e a Avaliação são efetuadas.

#### A. Fuzzificação

Em análise de séries temporais univariadas, a principal informação usada para descrever e prever o comportamento de uma série temporal  $y_t \in Y$  no instante  $t$  ( $t = 1, \dots, T$ ) consiste nos seus respectivos  $p$  atrasos ou defasagens  $y_{t-p} \in Y_p$  ( $p = 1, \dots, P$ ). Em cada defasagem são associados  $L$  conjuntos fuzzy  $A_{lp} = \{(y_{t-p}, \mu_{A_{lp}}(y_{t-p})) | y_{t-p} \in Y_p\}$  ( $l = 1, \dots, L$ ), cada um definido por uma função de pertinência  $\mu_{A_{lp}} : Y_p \rightarrow [0, 1]$  que assinala a cada observação  $y_{t-p}$  um grau de pertinência  $\mu_{A_{lp}}(y_{t-p})$  ao conjunto fuzzy  $A_{lp}$ . Similarmente,  $K$  conjuntos fuzzy  $B_k$  ( $k = 1, \dots, K$ ) são associados a  $Y$  (variável de saída).

Em geral, três aspectos norteiam o processo de definição das funções de pertinência: (i) forma (triangular, trapezoidal, etc.); (ii) suporte do conjunto  $\mu_{A_{lp}}(y_{t-p})$ ; e (iii) um rótulo linguístico, qualificando apropriadamente o subespaço compreendido por  $\mu_{A_{lp}}(y_{t-p})$  com um adjetivo correspondente. Idealmente, todas essas tarefas deveriam ser legadas a um especialista, mas, como em muitos casos um especialista está indisponível, é muito comum [2], [14] definir as funções de pertinência conforme mostrado na Figura 3.

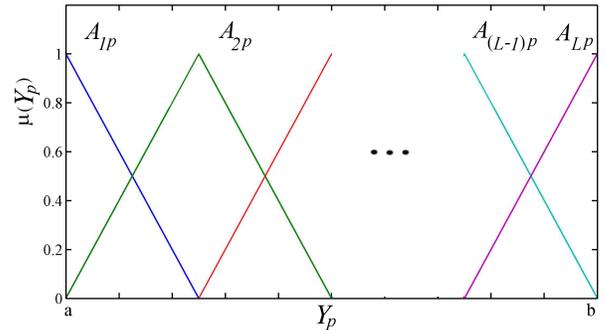


Figura 3. Funções de pertinência para as defasagens  $y_{t-p} \in Y_p$ . Para a saída  $Y$ , basta substituir cada  $A_{lp}$  por  $B_k$ .

O GPFIS-Forecast possibilita a utilização de diversos tipos de funções de pertinência (Figuras 4a e 4c) e de diferentes formas de particionamento (ou granularidade). Ao invés de se particionar o universo de discurso da variável de maneira uniforme (cf. Figura 4a), pode-se também usar a informação dos quartis (Figura 4b). Este tipo de particionamento é denominado Tukey.

#### B. Inferência Fuzzy

1) *Formulação*: Uma premissa de regra fuzzy é comumente definida por:

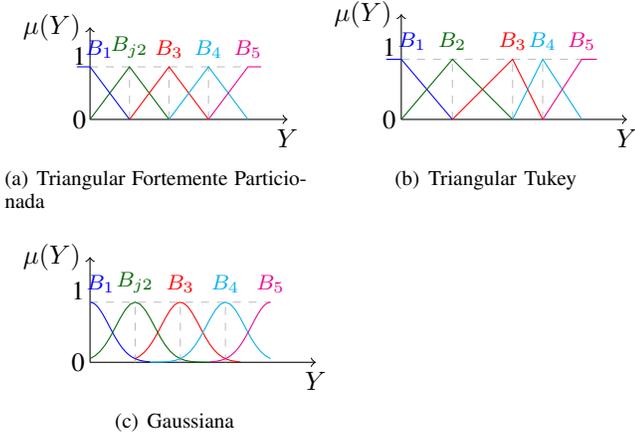


Figura 4. Funções de Pertinência com granularidade 5.

“Se  $Y_1$  é  $A_{11}$  e ... e  $Y_p$  é  $A_{1p}$  e ... e  $Y_P$  é  $A_{1P}$ ”

ou, em termos matemáticos:

$$\mu_{A_d}(\mathbf{y}_{t,P}) = \mu_{A_{11}}(y_{t-1}) * \dots * \mu_{A_{1P}}(y_{t-P}) \quad (1)$$

onde  $\mu_{A_d}(y_{t-1}, \dots, y_{t-P}) = \mu_{A_d}(\mathbf{y}_{t,P})$  é o grau de pertinência conjunto dos  $P$  atrasos de  $y_t$  com respeito à  $d$ -ésima premissa ( $d = 1, \dots, D$ ), computado a partir de uma  $t$ -norma  $*$ . Uma premissa pode ser elaborada combinando-se as  $\mu_{A_{1p}}(y_{t-p})$  por meio de  $t$ -normas,  $t$ -conormas, modificadores linguísticos e operadores de negação. Como consequência, o número de combinações possíveis cresce com o número de defasagens, conjuntos e operadores fuzzy. O GPFIS-Forecast emprega a PGMG para buscar as combinações mais promissoras de  $\mu_{A_{1p}}(y_{t-p})$ , isto é, premissas de regras fuzzy. A Figura 5 exemplifica uma solução típica proporcionada pela PGMG.

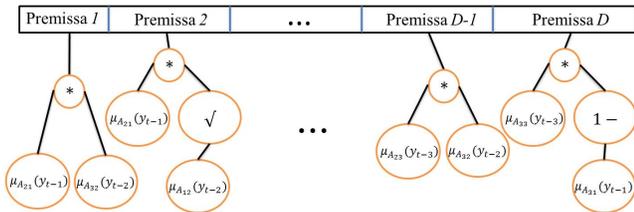


Figura 5. Exemplo de um conjunto de premissas codificadas em um indivíduo da PGMG.

Por exemplo, a premissa 1 representa:  $\mu_{A_1}(\mathbf{y}_{t,P}) = \mu_{A_{21}}(y_{t-1}) * \mu_{A_{32}}(y_{t-2})$ , que, em termos linguísticos, significa “Se  $Y_1$  é  $A_{21}$  e  $Y_2$  é  $A_{32}$ ”. Seja  $\mu_{A_d}(\mathbf{y}_{t,P})$  a  $d$ -ésima premissa codificada na  $d$ -ésima árvore/função de um indivíduo da PGMG. A Tabela I apresenta os componentes disponíveis no GPFIS-Forecast, alguns deles utilizados para compor as soluções mostradas na Figura 5.

No GPFIS-Forecast, as funções de pertinência  $\mu_{A_{1p}}(y_{t-p})$  dizem respeito aos conjuntos fuzzy de entrada que, na terminologia de PG, referem-se aos terminais, enquanto que o conjunto de funções é trocado pelos operadores fuzzy. A partir disto, a PGMG é usada para se obter um conjunto de premissas

Tabela I. CONJUNTOS FUZZY DE ENTRADA E OPERADORES FUZZY USADOS PARA GERAR A SOLUÇÃO ILUSTRADA NA FIGURA 5.

Conjuntos Fuzzy de Entrada		$\mu_{A_{11}}(y_{t-1}), \dots, \mu_{A_{L1}}(y_{t-1}), \dots, \mu_{A_{1P}}(y_{t-P}), \dots, \mu_{A_{LP}}(y_{t-P})$
Operadores Fuzzy	T-norma Mínimo	$\text{Mínimo}(\mu_{A_{ij}}(y_{t-j1}), \mu_{A_{ij}}(y_{t-j2}))$
	T-norma Lukasiewicz	$\text{Máximo}(0, \mu_{A_{ij}}(y_{t-j1}) + \mu_{A_{ij}}(y_{t-j2}) - 1)$
	Modificador Diluidor	$[\mu_{A_{ij}}(y_{t-j1})]^{1/2}$
	Modificador Concentrador	$[\mu_{A_{ij}}(y_{t-j1})]^2$

de regras fuzzy  $\mu_{A_d}(\mathbf{y}_{t,P})$ . De forma a elaborar uma base de regras plena, torna-se necessário associar a cada  $\mu_{A_d}(\mathbf{y}_{t,P})$  um termo consequente  $B_k$ . A etapa de Associação é responsável por isto.

2) *Associação*: Há basicamente duas formas para a definição de um termo consequente mais compatível com uma premissa de regra fuzzy: (i) permitir que a PGMG realize essa busca automaticamente (um procedimento comumente empregado em vários trabalhos); ou (ii) empregar métodos que, a partir de informações da base de dados, definam para uma premissa dada pela PGMG o termo consequente mais apropriado. No GPFIS-Forecast a opção (ii) é a adotada, com o principal objetivo de evitar que uma premissa com amplo suporte na base de dados, ou que esteja apta a prever corretamente um subespaço da variável de saída, acabe sendo associada a um termo consequente pouco apropriado. Este procedimento reduz o espaço de busca, pois, ao invés de investigar todos os elementos que compõem uma regra, restringe-se à elaboração de cada  $\mu_{A_d}(\mathbf{y}_{t,P})$ . Para tal, deve-se adotar uma métrica que meça a compatibilidade entre  $\mu_{A_d}(\mathbf{y}_{t,P})$  e um termo consequente  $B_k$ . No modelo GPFIS-Forecast, emprega-se o Grau de Confiança Fuzzy ( $FCD_k$ ), dado por:

$$FCD_k = \frac{\sum_{t=1}^T \mu_{A_d}(\mathbf{y}_{t,P}) \mu_{B_k}(y_t)}{\sqrt{\sum_{t=1}^T \mu_{A_d}(\mathbf{y}_{t,P})^2} \sqrt{\sum_{t=1}^T \mu_{B_k}(y_t)^2}} \in [0, 1] \quad (2)$$

onde  $\sum_{t=1}^T \mu_{A_d}(\mathbf{y}_{t,P}) \mu_{B_k}(y_t)$  é o grau de compatibilidade entre a  $d$ -ésima premissa e o  $k$ -ésimo termo consequente, enquanto que  $\sqrt{\sum_{t=1}^T \mu_{A_d}(\mathbf{y}_{t,P})^2}$  e  $\sqrt{\sum_{t=1}^T \mu_{B_k}(y_t)^2}$  são fatores de normalização. Quando  $\mu_{A_d}(\mathbf{y}_{t,P}) = \mu_{B_k}(y_t)$  para todo  $t$ ,  $FCD_k = 1$ , ou seja, a premissa e o termo consequente são totalmente compatíveis. O termo consequente de  $\mu_{A_d}(\mathbf{y}_{t,P})$  é definido pelo  $k$  que maximiza a métrica  $FCD_k$ . Uma premissa com  $FCD_k = 0$ , para todo  $k$ , não é associada a nenhum termo consequente e, portanto, não é considerada como uma regra fuzzy.

3) *Agregação*: Uma premissa associada ao  $k$ -ésimo termo consequente (ou seja, uma regra fuzzy) é denotada por  $\mu_{A_d^{(k)}}(\mathbf{y}_{t,P})$ , que, em termos linguísticos significa: “Se  $Y_1$  é  $A_{11}$ , e ..., e  $Y_P$  é  $A_{1P}$ , então  $Y$  é  $B_k$ ”. Portanto, toda a base de regras pode ser descrita como  $\mu_{A_1^{(k)}}(\mathbf{y}_{t,P}), \dots, \mu_{A_D^{(k)}}(\mathbf{y}_{t,P}), \forall k = 1, \dots, K$ . Dada uma nova observação temporal  $y_t^*$  e, por conseguinte, seus  $P$  atrasos  $\mathbf{y}_{t,P}^*$ , é possível que haja uma ativação de diversas premissas, associadas a um mesmo termo consequente ou não. De maneira a gerar um valor consensual, a etapa de Agregação busca combinar os graus de ativação de todas as regras de um mesmo termo consequente.

Considere-se  $D^{(k)}$  o número de regras fuzzy do  $k$ -ésimo termo consequente ( $d^{(k)} = 1^{(k)}, 2^{(k)}, \dots, D^{(k)}$ ). Dado um operador de agregação  $g : [0, 1]^{D^{(k)}} \rightarrow [0, 1]$  (ver [3]), o grau de pertinência predito de  $\mathbf{y}_{t,P}^*$  ao  $k$ -ésimo conjunto fuzzy ( $\hat{\mu}_{B_k}(y_t^*)$ ) é computado por:

$$\hat{\mu}_{B_1}(y_t^*) = g[\mu_{A_{1(1)}}(\mathbf{y}_{t,P}^*), \dots, \mu_{A_{D(1)}}(\mathbf{y}_{t,P}^*)] \quad (3)$$

$$\hat{\mu}_{B_2}(y_t^*) = g[\mu_{A_{1(2)}}(\mathbf{y}_{t,P}^*), \dots, \mu_{A_{D(2)}}(\mathbf{y}_{t,P}^*)] \quad (4)$$

$$\dots$$

$$\hat{\mu}_{B_K}(y_t^*) = g[\mu_{A_{1(K)}}(\mathbf{y}_{t,P}^*), \dots, \mu_{A_{D(K)}}(\mathbf{y}_{t,P}^*)] \quad (5)$$

Há vários operadores de agregação disponíveis na literatura [3], sendo o de Máximo o mais usado entre eles. Não obstante, outros operadores tais como a média aritmética ou a média ponderada são passíveis de uso. Com respeito à média aritmética ponderada, torna-se necessário resolver o problema de Mínimos Quadrados Restrito (MQR) de forma a determinar os pesos da combinação linear:

$$\min : \sum_{t=1}^T (\hat{\mu}_{B_k}(y_t) - \sum_{d^{(k)}=1}^{D^{(k)}} w_{d^{(k)}} \mu_{A_{d^{(k)}}}(\mathbf{y}_{t,P}))^2 \quad (6)$$

$$s.a. : \sum_{d^{(k)}=1}^{D^{(k)}} w_{d^{(k)}} = 1 \text{ e } w_{d^{(k)}} \geq 0$$

onde  $w_{d^{(k)}}$  é o peso ou o grau de influência de  $\mu_{A_{d^{(k)}}}(\mathbf{y}_{t,P})$  na predição dos valores associados ao  $k$ -ésimo termo consequente. O problema de minimização acima é típico de Programação Quadrática, cuja solução é facilmente computada a partir dos algoritmos descritos em [26]. De maneira a proporcionar um operador de agregação que alie a característica de t-conorma do Máximo à minimização do erro, introduz-se o operador Max-MQR, dado por:

$$\hat{\mu}_{B_1}(y_t^*) = \max[w_{1(1)}\mu_{A_{1(1)}}(\mathbf{y}_{t,P}^*), \dots, w_{D(1)}\mu_{A_{D(1)}}(\mathbf{y}_{t,P}^*)]$$

$$\hat{\mu}_{B_2}(y_t^*) = \max[w_{1(2)}\mu_{A_{1(2)}}(\mathbf{y}_{t,P}^*), \dots, w_{D(2)}\mu_{A_{D(2)}}(\mathbf{y}_{t,P}^*)]$$

...

$$\hat{\mu}_{B_K}(y_t^*) = \max[w_{1(K)}\mu_{A_{1(K)}}(\mathbf{y}_{t,P}^*), \dots, w_{D(K)}\mu_{A_{D(K)}}(\mathbf{y}_{t,P}^*)]$$

onde cada  $w_{d^{(k)}}$  é combinado a cada  $\mu_{A_{d^{(k)}}}(\mathbf{y}_{t,P}^*)$ , sendo por fim agregado por meio do operador Máximo. Obtidos os valores para cada  $\hat{\mu}_{B_1}(y_t^*), \dots, \hat{\mu}_{B_K}(y_t^*)$ , a etapa de Defuzzificação utiliza essa informação para fornecer um valor  $\hat{y}_t^*$  preciso.

### C. Defuzzificação

Há diversos operadores de Defuzzificação disponíveis, sendo que os mais utilizados são Centro de Área, Média dos Máximos e Altura [25]. No modelo GPFIS-Forecast opta-se pela versão normalizada do método da Altura (7), dada por:

$$\hat{y}_t = \frac{c_1 \hat{\mu}_{B_1}(y_t) + \dots + c_K \hat{\mu}_{B_K}(y_t)}{\hat{\mu}_{B_1}(y_t) + \dots + \hat{\mu}_{B_K}(y_t)} \quad (7)$$

A partir desta informação precisa, é possível avaliar o indivíduo do modelo GPFIS-Forecast.

### D. Avaliação

O procedimento de Avaliação é definido por um objetivo primário – minimização de erro – e um secundário – redução de complexidade. O objetivo primário é responsável pela ordenação dos indivíduos na população, ao passo que o secundário é usado como critério de desempate.

Uma função objetivo bastante empregada em problemas de previsão, e usada ao longo dos experimentos, é o Erro Médio Simétrico Absoluto Percentual (SMAPE) [19]:

$$SMAPE = \frac{\sum_{t=1}^T \frac{|y_t - \hat{y}_t|}{(y_t + \hat{y}_t)/2}}{T} \quad (8)$$

O melhor indivíduo na população é a solução que minimiza a equação (8). Ao mesmo tempo, usando o critério secundário, o GPFIS-Forecast tenta reduzir a complexidade da base de regras por meio de uma simples heurística: Pressão Lexicográfica Parcimoniosa [21]. Essa técnica é usada somente na fase de seleção: dados dois indivíduos com a mesma aptidão, o melhor entre eles é o que possuir o menor número de nós. Menor número de nós indica regras com poucos elementos antecedentes, modificadores linguísticos ou operadores de negação, assim como poucas premissas ( $\mu_{A_d}(\mathbf{y}_{t,P})$ ), ou seja, uma base de regras mais enxuta. Após a etapa de Avaliação, os indivíduos são selecionados (pelo procedimento do Torneio) e recombinados por meio dos operadores de mutação e cruzamentos de baixo e de alto níveis. Esse processo é repetido até que um critério de parada seja atendido. Quando isto ocorre, a última população é retornada e o melhor indivíduo providencia as suas previsões.

## IV. ESTUDOS DE CASO

### A. Descrição dos Experimentos

Esta seção aborda a metodologia utilizada para obtenção dos resultados e as principais métricas para avaliação de desempenho. Devido à grande quantidade de parâmetros a serem testados (tais como t-normas, métodos de associação, operadores de agregação, funções de pertinência, etc.), para grandes base de dados o esforço computacional requerido praticamente inviabiliza a investigação de todas as combinações possíveis. Em casos como esse, é recomendada a utilização de experimentos sequenciais [22], em que cada experimento é realizado em blocos causalizados, em vários níveis. Um exemplo seria avaliar, dentre  $k$  tipos de t-normas e mantendo fixos os outros parâmetros do modelo, aquela que proporciona o menor erro em  $b$  blocos, para, em seguida, avaliar modificadores, novamente mantendo fixos os outros parâmetros do modelo, inclusive o melhor parâmetro do experimento anterior. No final de cada experimento é efetuado o teste não-paramétrico de Friedman [9], com o intuito de identificar se alguma das configurações se destaca das demais e qual a melhor configuração atual.

Foram realizados 36 experimentos agrupados em classes, todos eles executados sem o pré-processamento das séries temporais, de maneira a garantir a interpretabilidade dos resultados levantados. A classe base é aquela que utiliza a melhor configuração obtida por Koshiyama [19]. Na classe 1, investiga-se a utilização de diferentes tipos funções de pertinência (triangular, gaussiana e Tukey) e granularidade (1, 2 e 3) das variáveis

de entrada e saída; na classe de experimentos 2, analisam-se as t-normas produto, mínimo e Lukasiewicz, mantendo fixos os demais parâmetros da melhor configuração da classe anterior; a classe 3 agrega os experimentos com distintos modificadores (concentrador e diluidor), novamente mantendo fixos os demais parâmetros da melhor configuração da classe anterior; na classe 4, investigam-se métodos de associação; na classe 5, métodos de agregação e, na classe 6, métodos de defuzzificação.

Cada experimento é repetido 10 vezes e o número de gerações da PGMG para as classe acima é 25, exceto para a classe base, em que se utilizam 100 gerações. Não se realiza pré-processamento nos dados de entrada, pois isso afetaria a interpretabilidade do modelo, visto que se deseja obter informação da série temporal íntegra e não do seu resíduo. As demais configurações fixas, comuns a todos os experimentos, são resumidas na Tabela II. Os experimentos nas sete classes são executados com a base de dados da competição NN3, que visou a premiar o modelo baseado técnicas de Inteligência Computacional que efetuasse a previsão de 11 e 111 séries temporais da maneira mais acurada possível. Essa competição requer um horizonte de previsão fixo de 18 meses, uma a janela de previsão de 12 períodos e avaliação por SMAPE.

Em seguida, a melhor configuração das sete classes foi executada com a base de dados mensais Industriais da competição M3, com a programação genética evoluindo com 100 gerações. Esta competição tinha como objetivo expandir e incentivar a pesquisa na área de previsão. O horizonte de previsão para as séries mensais desta competição é de 18 períodos e a janela de previsão é de 12 períodos.

Tabela II. PRINCIPAIS CONFIGURAÇÕES DO MODELO GPFIS-FORECAST.

Parâmetro	Valor
Tamanho da população	100
Altura máxima da árvore	5
Tamanho do torneio	2
Taxa de cruzamento de alto nível	50%
Taxa de cruzamento de baixo nível	85%
Taxa de mutação	10%
Taxa de clonagem	5%
Taxa de elitismo	1%
Pressão lexicográfica	Sim

## B. Resultados e Discussões

Nos experimentos, a melhor configuração obtida (em termos de acurácia) na execução das sete classes de experimentos foi a seguinte: (1) função de pertinência tipo Tukey com granularidade 3; (2) t-norma Łukasiewicz, e modificadores de diluição e concentração; (3) associação pelo Grau de Confiança Fuzzy (4) Agregação pelo método do máximo MQR e (5) defuzzificação pelo método da altura e a função de ajuste sendo a minimização do SMAPE. Os resultados para esta configuração são apresentados nas Tabelas III e IV, nas quais o GPFIS-Forecast aparece na nona e vigésima segunda posições nos rankings das competições NN3 e M3, respectivamente. Para cada série, uma mesma configuração foi repetida 10 vezes, de forma que os resultados refletem a média dos SMAPES para estas 10 repetições. Se forem considerados apenas as melhores base de regras obtidas em 10 repetições,

o modelo atinge a segunda posição no ranking da NN3 e a décima terceira posição no ranking da M3.

Os resultados que dizem respeito à interpretabilidade, expressos pela média de regras e pelo número médio de elementos nas premissas, são excelentes e justificam plenamente o uso da metodologia empregada, que busca um compromisso entre acurácia e interpretabilidade linguística. Como pretendido, as bases de regras são efetivamente compactas para ambas as competições. Os resultados (médios) de 1,4 e 1,25 elementos por premissa favorecem em muito a interpretabilidade, ainda mais considerando que as janelas têm tamanho 12. Saliente-se ainda que, em nome desta interpretabilidade, mesmo com eventual prejuízo da acurácia, não houve pré-processamento dos dados.

Tabela III. MELHORES RESULTADOS DA NN3 PARA 111 SÉRIES.

Rank	Participante	SMAPE	Média de regras	Número médio de elementos nas premissas
-	Stat. Contender - Wildi	14,84%	-	-
-	Stat. Benchmark - Theta Method (Nikolopoulos)	14,89%	-	-
1	Illies, Jäger, Kosuchinas, Rincon, Sakenas, Vaskevicius	15,18%	-	-
-	Stat. Benchmark - ForecastPro (Stellwagen)	15,44%	-	-
-	CI Benchmark - Theta AI (Nikolopoulos)	15,66%	-	-
-	Stat. Benchmark - Autobox (Reilly)	15,95%	-	-
2	Adeodato, Vasconcelos, Arnaud, Chunha, Monteiro	16,17%	-	-
3	Flores, Anaya, Ramirez, Morales	16,31%	-	-
4	Chen, Yao	16,55%	-	-
5	D'yakonov	16,57%	-	-
6	Kamel, Atiya, Gayar, El-Shishiny	16,92%	-	-
7	Abou-Nasr	17,54%	-	-
8	Theodosiou, Swamy	17,55%	-	-
-	CI Benchmark - Naive MLP (Crone)	17,84%	-	-
9	<b>GPFIS-Forecast</b>	<b>17,95%</b>	<b>15</b>	<b>1,4</b>
10	de Vos	18,24%	-	-

Fonte: Retirado do site da NN3 [6].

Tabela IV. RANKING M3 PARA BASE DE DADOS INDUSTRIAIS MENSIS

Ranking	Métodos de Previsão	SMAPE	Média de regras	Número médio de elementos nas premissas
1	Forecastpro	11,61 %	-	-
2	ForecastX	11,73 %	-	-
3	B-J automatic	11,94 %	-	-
4	RBF	12,12 %	-	-
5	Dampen	12,13 %	-	-
6	THETA	12,22 %	-	-
7	Single	12,32 %	-	-
8	Comb S-H-D	12,35 %	-	-
9	Autobox1	12,82 %	-	-
10	THETA <sub>sm</sub>	12,87 %	-	-
11	Autobox2	12,98 %	-	-
12	ARARMA	13,01 %	-	-
13	Naive2	13,20 %	-	-
14	SMARTFCS	13,25 %	-	-
15	Auto-ann	13,30 %	-	-
16	Flors-Pearc2	13,32 %	-	-
17	Winter	13,37 %	-	-
18	Holt	13,55 %	-	-
19	Autobox3	13,58 %	-	-
20	PP-Autocast	13,78 %	-	-
21	Robust Trend	13,92 %	-	-
22	<b>GPFIS-Forecast</b>	<b>14,50 %</b>	<b>16</b>	<b>1,25</b>
23	Flores/Pearce1	14,71 %	-	-

Fonte: Retirado do site da M3 [5].

## V. CONCLUSÕES

Este trabalho apresentou aperfeiçoamentos ao modelo GPFIS-Forecast no que tange as operações de fuzzificação e inferência fuzzy (formulação, associação e agregação). Foram

realizados diversos experimentos, considerando-se as séries das competições NN3 e M3, sem pré-processamento, e identificou-se a configuração que proporcionou os melhores resultados. Deve-se salientar que buscou-se um compromisso entre acurácia e interpretabilidade linguística, que certamente seria perdida em um pré-processamento.

Pretende-se, futuramente, identificar regras recorrentes (e benéficas) e utilizá-las para povoar indivíduos na primeira geração, proporcionando uma vantagem evolutiva e, em consequência, reduzindo mais ainda o esforço computacional. Outro aspecto será a elaboração de procedimentos para a geração de intervalos de previsão com base no conjunto fuzzy de saída do processo de inferência.

#### REFERÊNCIAS

- [1] José Luis Aznarte, Jesús Alcalá-Fdez, Antonio Arauzo-Azofra, and José Manuel Benítez. Financial time series forecasting with a bio-inspired fuzzy model. *Expert Systems with Applications*, 39(16):12302–12309, 2012.
- [2] F.J. Berlanga, A.J. Rivera, M.J. del Jesus, and F. Herrera. Gp-coach: Genetic programming-based learning of compact and accurate fuzzy rule-based classification systems for high-dimensional problems. *Information Sciences*, 180(8):1183–1200, 2010.
- [3] Tomasa Calvo, Anna Kolesárová, Magda Komorníková, and Radko Mesiar. Aggregation operators: Properties, classes and construction methods. In Tomasa Calvo, Gaspar Mayor, and Radko Mesiar, editors, *Aggregation Operators*, volume 97 of *Studies in Fuzziness and Soft Computing*, pages 3–104. Physica-Verlag HD, 2002.
- [4] Yuehui Chen, Bo Yang, A. Abraham, and Lizhi Peng. Automatic design of hierarchical takagi-sugeno type fuzzy systems using evolutionary algorithms. *IEEE Transactions on Fuzzy Systems*, 15(3):385–397, 2007.
- [5] M3 Competition. <http://forecasters.org/resources/time-series-data/m3-competition/>, jun 2015.
- [6] NN3 Competition. <http://www.neural-forecasting-competition.com/NN3/results.htm>, jun 2015.
- [7] O. Cordon, F. Gomide, F. Herrera, F. Hoffmann, and L. Magdalena. Ten years of genetic fuzzy systems: current framework and new trends. *Fuzzy sets and systems*, 141(1):5–31, 2004.
- [8] O. Cordon and F. Herrera. Hybridizing genetic algorithms with sharing scheme and evolution strategies for designing approximate fuzzy rule-based systems. *Fuzzy Sets and Systems*, 118(2):235–255, 2001.
- [9] Joaquín Derrac, Salvador García, Daniel Molina, and Francisco Herrera. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm and Evolutionary Computation*, 1(1):3–18, 2011.
- [10] W. Enders. *Applied econometric time series*. John Wiley & Sons, New Jersey, 2008.
- [11] Marian B. Gorzalczy and Filip Rudzinski. Genetic fuzzy rule-based modelling of dynamic systems using time series. In Leszek Rutkowski, Marcin Korytkowski, Rafal Scherer, Ryszard Tadeusiewicz, Lotfi A. Zadeh, and Jacek M. Zurada, editors, *Swarm and Evolutionary Computation*, volume 7269 of *Lecture Notes in Computer Science*, pages 231–239. Springer Berlin Heidelberg, 2012.
- [12] Esmail Hadavandi, Hassan Shavandi, and Arash Ghanbari. An improved sales forecasting approach by the integration of genetic fuzzy systems and data clustering: Case study of printed circuit board. *Expert Systems with Applications*, 38(8):9392–9399, 2011.
- [13] S. Haykin. *Neural networks: a comprehensive foundation*. Prentice Hall, New Jersey, 2007.
- [14] F. Herrera. Genetic fuzzy systems: taxonomy, current research trends and prospects. *Evolutionary Intelligence*, 1(1):27–46, 2008.
- [15] Mark Hinchliffe, Hugo Hiden, Ben McKay, Mark Willis, Ming Tham, and Geoffery Barton. Modelling chemical process systems using a multi-gene genetic programming algorithm. In John R. Koza, editor, *Late Breaking Papers at the Genetic Programming 1996 Conference Stanford University July 28-31, 1996*, pages 56–65, Stanford University, CA, USA, 1996. Stanford University.
- [16] Charles C. Holt. Forecasting seasonals and trends by exponentially weighted moving averages. *International Journal of Forecasting*, 20(1):5 – 10, 2004.
- [17] N. Kasabov and Q. Song. Denfis: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *IEEE Transactions on Fuzzy Systems*, 10(2):144–154, 2002.
- [18] G. Kirchgässner, J. Wolters, and U. Hassler. *Introduction to Modern Time Series Analysis*. Springer Texts in Business and Economics. Springer Berlin Heidelberg, 2012.
- [19] Adriano Soares Koshiyama, Marley Maria Bernardes Rebuszi Vellasco, and Ricardo Tanscheit. *GPFIS: Um Sistema Fuzzy-Genético Genérico baseado em Programação Genética*. Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro, 228p., Dissertação de Mestrado, Rio de Janeiro, 2014.
- [20] J. R. Koza. *Genetic programming: on the programming of computers by means of natural selection*. The MIT Press, Massachusetts, 1992.
- [21] Sean Luke and Liviu Panait. Lexicographic parsimony pressure. In W. B. Langdon, E. Cantú-Paz, K. Mathias, R. Roy, D. Davis, R. Poli, K. Balakrishnan, V. Honavar, G. Rudolph, J. Wegener, L. Bull, M. A. Potter, A. C. Schultz, J. F. Miller, E. Burke, and N. Jonoska, editors, *GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 829–836, New York, 2002. Morgan Kaufmann Publishers.
- [22] D.C. Montgomery. *Design and Analysis of Experiments, 6th Edition Set*. John Wiley & Sons, Limited, 2007.
- [23] A. K. Palit and D. Popovic. *Computational intelligence in time series forecasting: theory and engineering applications*. Springer-Verlag, New York, 2005.
- [24] R. Poli, W. B. Langdon, and N. F. McPhee. *A Field Guide to Genetic Programming*. Lulu.com, Rayleigh, 2008.
- [25] Shounak Roychowdhury and Witold Pedrycz. A survey of defuzzification strategies. *International Journal of Intelligent Systems*, 16(6):679–695, 2001.
- [26] B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Massachusetts, 2001.
- [27] Dominic Seanson, Mark Willis, and Gary Montague. Co-evolution of non-linear PLS model components. *Journal of Chemometrics*, 21(12):592–603, 2007.
- [28] A. Tsakonas. Local and global optimization for takagi-sugeno fuzzy system by memetic genetic programming. *Expert Systems with Applications*, 40(8):3282–3298, 2013.
- [29] M. M. B. R. Vellasco, M. A. C. Pacheco, L. S. Ribeiro-Neto, and F. J. de Souza. Electric load forecasting: evaluating the novel hierarchical neuro-fuzzy bsp model. *International journal of electrical power & energy systems*, 26(2):131–142, 2004.
- [30] A. F. Zuur, E. N. Ieno, and G. M. Smith. *Analysing ecological data*. Springer, New York, 2007.