



**Centro Federal de Educação Tecnológica de Minas Gerais**

Professor: Alisson Marques da Silva

Disciplina: I.C

Aluno: Celso Vinícius Sudário

Matrícula: 20203003611

Divinópolis, 19 de Novembro de 2024

## Atividade 02 - Sistema Fuzzy de Takagi-Sugeno

### 1. Descrição da Implementação

Este trabalho apresenta a implementação de um sistema fuzzy baseado no modelo Takagi-Sugeno (TS) de primeira ordem, projetado para aproximar uma função real  $f(x)$  por meio de combinações ponderadas de regras fuzzy. A escolha pela primeira ordem deve-se à tentativa de modelar relações lineares locais, o que busca melhorar a precisão em relação ao modelo de ordem zero.

O sistema foi desenvolvido na linguagem Python, empregando bibliotecas para tarefas complementares. A biblioteca NumPy foi utilizada para os cálculos matemáticos, incluindo a implementação do método Recursive Least Squares (RLS). Já a biblioteca Matplotlib foi responsável pela geração dos gráficos comparativos. Além disso, como requisito da atividade, funções de pertinência, operadores fuzzy e o cálculo do erro foram implementados diretamente, sem o uso de bibliotecas ou estruturas prontas, seguindo as diretrizes propostas para fins didáticos e para oferecer maior envolvimento prático no desenvolvimento.

Por fim, o sistema é composto por três elementos fundamentais: funções de pertinência, operadores fuzzy e consequentes ajustáveis. Esses elementos foram configurados de maneira a tentar explorar a competência e a usabilidade do modelo. Cada componente é detalhado a seguir.

#### 1.1. Funções de Pertinência

As funções de pertinência são responsáveis por determinar o grau de ativação das regras fuzzy. Neste sistema, três tipos de funções foram utilizados:

**Função Gaussiana** Modela a pertinência com uma curva simétrica descrita pela equação:

$$\mu(x) = e^{-\frac{(x-c)^2}{2\sigma^2}},$$

onde  $c$  é o centro e  $\sigma$  define a largura da curva.

**Função Triangular** Representa a pertinência por meio de segmentos lineares ascendentes e descendentes:

$$\mu(x) = \begin{cases} 0, & x \leq a \text{ ou } x \geq c, \\ \frac{x-a}{b-a}, & a < x \leq b, \\ \frac{c-x}{c-b}, & b < x < c, \end{cases}$$

onde  $a$ ,  $b$  e  $c$  são os limites inferior, médio e superior da função.

**Função Trapezoidal** Inclui uma região plana onde a pertinência alcança o valor máximo:

$$\mu(x) = \begin{cases} 0, & x \leq a \text{ ou } x \geq d, \\ \frac{x-a}{b-a}, & a < x \leq b, \\ 1, & b < x \leq c, \\ \frac{d-x}{d-c}, & c < x < d, \end{cases}$$

onde  $a$ ,  $b$ ,  $c$  e  $d$  delimitam as regiões de transição.

Essas funções buscam capturar diferentes comportamentos de pertinência para a entrada  $x$ , com variações que vão de curvas suaves (gaussianas) a representações mais rígidas (triangulares e trapezoidais).

## 1.2. Operadores Fuzzy

Os operadores fuzzy são utilizados para combinar os graus de pertinência das regras. No sistema implementado, os seguintes operadores foram avaliados:

$$\text{Min: } \mu_c(x) = \min\{\mu_1(x), \mu_2(x), \dots, \mu_R(x)\},$$

$$\text{Prod: } \mu_c(x) = \prod_{i=1}^R \mu_i(x),$$

$$\text{Max: } \mu_c(x) = \max\{\mu_1(x), \mu_2(x), \dots, \mu_R(x)\},$$

Esses operadores foram implementados diretamente no código, de forma que sua substituição seja dinâmica, buscando proporcionar a avaliação de diferentes combinações de pertinência e seu impacto nos resultados do modelo.

## 1.3. Configuração do Sistema Fuzzy Takagi-Sugeno

O modelo fuzzy foi configurado com 10 regras fuzzy (clusters), que segmentam o universo de discurso da entrada  $x$ . Cada regra é associada a uma função consequente de

primeira ordem:

$$y_i(x) = a_i \cdot x + b_i,$$

onde os coeficientes  $a_i$  e  $b_i$  são ajustados de forma dinâmica por meio do método Recursive Least Squares (RLS). A saída do sistema é definida por:

$$y(x) = \frac{\sum_{i=1}^R \mu_i(x) \cdot y_i(x)}{\sum_{i=1}^R \mu_i(x)},$$

em que  $\mu_i(x)$  representa o grau de ativação da  $i$ -ésima regra.

A configuração do sistema no código inclui o número de regras, o tipo de função de pertinência, o operador fuzzy selecionado e o ajuste dinâmico dos consequentes. Abaixo, um exemplo de configuração utilizada:

```
1 fuzzy_system = FuzzySystemTS(  
2     num_rules=10,  
3     order=1,  
4     membership_type='triangular',  
5     operator='min'  
6 )  
7 fuzzy_system.define_membership_functions((0, 10))
```

## 1.4. Ajuste dos Consequentes

O ajuste dos consequentes é realizado por meio do método Recursive Least Squares (RLS), que busca minimizar iterativamente o erro quadrático entre a saída predita  $y(x)$  e o valor real  $f(x)$ . O método é representado pelas equações:

$$\mathbf{K} = \frac{\mathbf{P}\mathbf{x}}{\lambda + \mathbf{x}^\top \mathbf{P}\mathbf{x}}, \quad \theta = \theta + \mathbf{K} \cdot (y - \mathbf{x}^\top \theta),$$
$$\mathbf{P} = \frac{\mathbf{P} - \mathbf{K}\mathbf{x}^\top \mathbf{P}}{\lambda}.$$

Nessas equações,  $\lambda$  representa o fator de esquecimento,  $\mathbf{P}$  é a matriz de covariância, e  $\theta$  é o vetor de parâmetros ajustáveis ( $a_i, b_i$ ). Essa abordagem busca garantir que o modelo se ajuste eficientemente às dinâmicas da função real.

## 2. Resultados e Análise

Nesta seção, são apresentados os resultados gráficos das aproximações fuzzy obtidas pelas diferentes configurações testadas. Os gráficos incluem as curvas da função real  $f(x)$  e das aproximações fuzzy, além dos gráficos de erro correspondentes. A métrica utilizada para avaliação foi o RMSE (Root Mean Squared Error), conforme definido anteriormente.

## 2.1. Configuração 1: Pertinência Gaussiana e Operador Produto

Na Figura 1, a aproximação fuzzy utilizando funções de pertinência gaussianas em combinação com o operador produto é exibida. A configuração busca capturar o comportamento da função real, apresentando um RMSE de 0.3466. O gráfico de erro correspondente, ilustrado na Figura 2, indica que os desvios mais significativos ocorrem nos picos da função real, sugerindo uma limitação na captura de variações inesperadas.

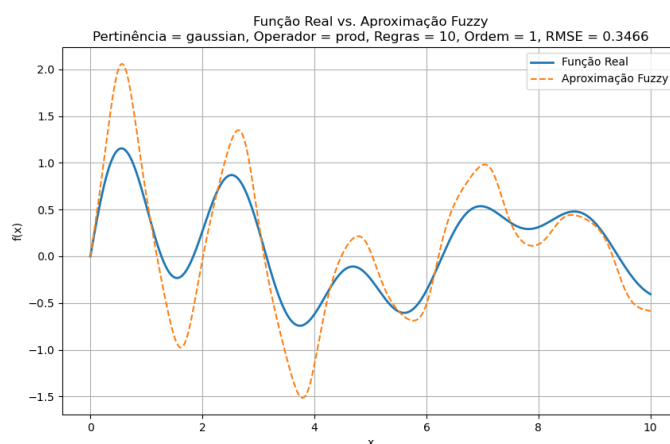


Figura 1. Função Real vs. Aproximação Fuzzy: Pertinência Gaussiana, Operador Produto.

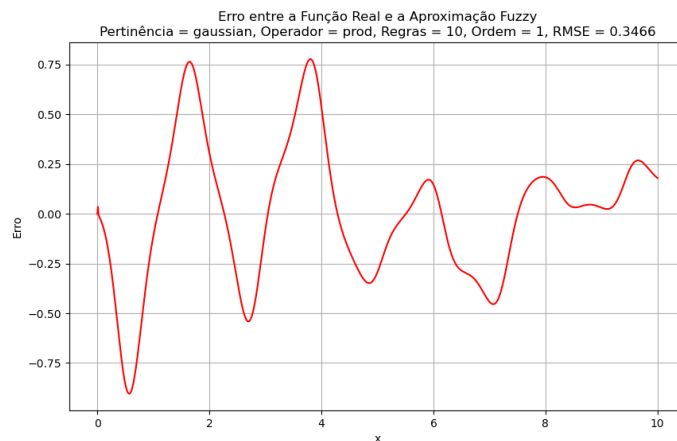
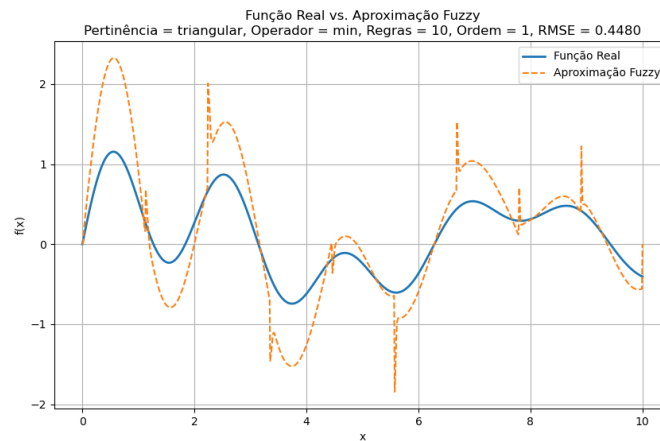


Figura 2. Erro entre a Função Real e a Aproximação Fuzzy: Pertinência Gaussiana, Operador Produto.

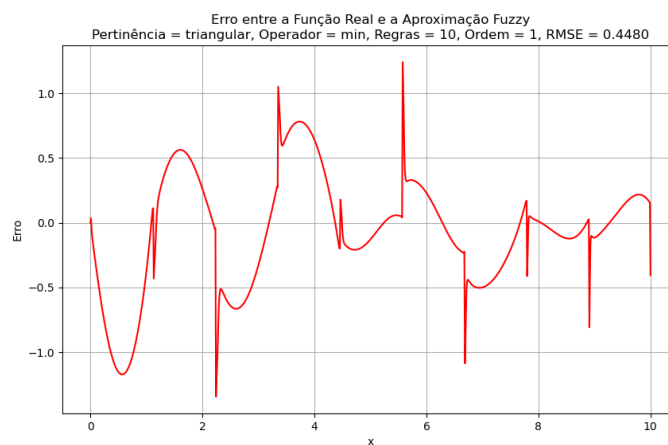
## 2.2. Configuração 2: Pertinência Triangular e Operador Mínimo

A Figura 3 apresenta os resultados obtidos com funções de pertinência triangulares e o operador mínimo. Esta configuração alcançou um RMSE de 0.4480, valor superior ao da configuração anterior, indicando uma menor precisão no ajuste da função real. O

gráfico de erro na Figura 4 demonstra maiores variações durante as transições na função, evidenciando dificuldades no acompanhamento de mudanças bruscas.



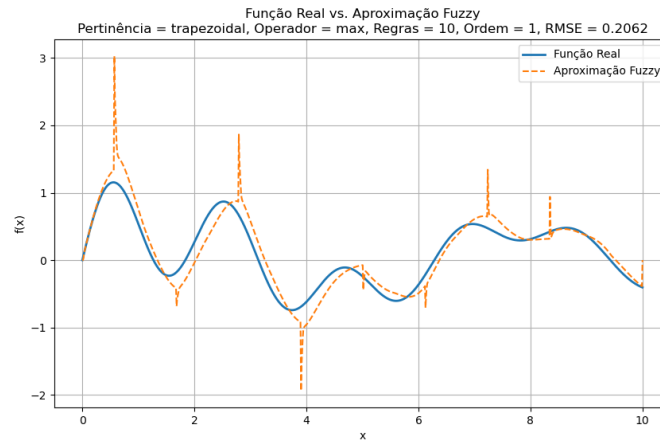
**Figura 3. Função Real vs. Aproximação Fuzzy: Pertinência Triangular, Operador Mínimo.**



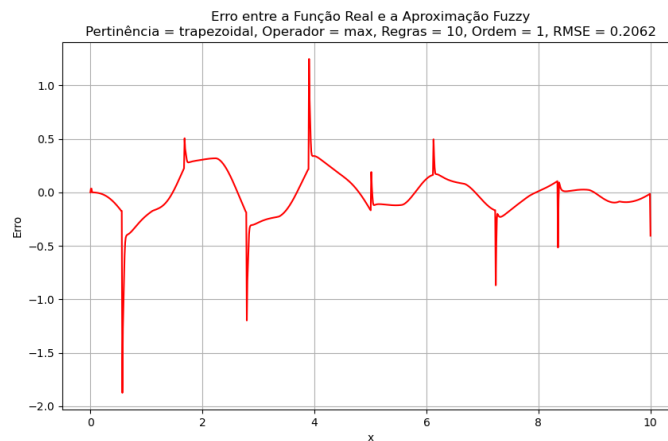
**Figura 4. Erro entre a Função Real e a Aproximação Fuzzy: Pertinência Triangular, Operador Mínimo.**

### 2.3. Configuração 3: Pertinência Trapezoidal e Operador Máximo

Os resultados da configuração com funções de pertinência trapezoidais e o operador máximo estão apresentados na Figura 5. Esta combinação alcançou o menor RMSE entre as opções testadas, com um valor de 0.2062. O gráfico de erro na Figura 6 evidencia um desempenho mais consistente, com menores desvios, especialmente em regiões planas da função real. Essa configuração busca maximizar a suavidade e o alinhamento com o comportamento original da função.



**Figura 5. Função Real vs. Aproximação Fuzzy: Pertinência Trapezoidal, Operador Máximo.**



**Figura 6. Erro entre a Função Real e a Aproximação Fuzzy: Pertinência Trapezoidal, Operador Máximo.**

### 3. Discussão e Comparação

Nesta seção, são analisados os resultados obtidos, enfatizando os impactos das configurações de operadores fuzzy e funções de pertinência no desempenho do sistema. A avaliação foi conduzida com base na métrica RMSE, destacando as combinações que apresentaram os menores valores de erro.

#### 3.1. Métrica de Erro e Desempenho

A Tabela 1 sintetiza os valores de RMSE obtidos para cada configuração testada, associando a função de pertinência e o operador fuzzy utilizados.

**Tabela 1. Resultados das combinações testadas com base no RMSE.**

<b>Configuração</b>	<b>Função de Pertinência</b>	<b>Operador</b>	<b>RMSE</b>
1	Gaussiana	Produto	0.3466
2	Triangular	Mínimo	0.4480
3	Trapezoidal	Máximo	0.2062

A combinação envolvendo função de pertinência trapezoidal e operador máximo obteve o menor RMSE, com um valor de 0.2062. Este resultado pode indicar que a função trapezoidal busca capturar transições lineares de forma mais eficiente, reduzindo desvios significativos. Em contraste, a configuração com pertinência triangular e operador mínimo apresentou o maior RMSE, sugerindo que dificuldades em lidar com transições bruscas impactam negativamente a aproximação.

### **3.2. Impacto das Configurações**

Os resultados destacam que as escolhas de funções de pertinência e operadores fuzzy exercem influência significativa sobre o desempenho do modelo Takagi-Sugeno. O operador máximo, empregado na Configuração 3, buscou produzir aproximações mais estáveis e com menor erro médio, enquanto o operador produto, usado na Configuração 1, obteve resultados intermediários e mais sensíveis às variações da função real. O operador mínimo, por outro lado, apresentou maior sensibilidade, levando a erros localizados mais expressivos.

Em relação às funções de pertinência, as gaussianas apresentaram suavidade consistente, porém com menor capacidade de adaptação a picos e transições rápidas. As funções triangulares, devido ao seu comportamento mais rígido, amplificaram variações na função real, resultando em picos no gráfico de erro. As funções trapezoidais equilibraram suavidade e flexibilidade, contribuindo para uma aproximação mais uniforme e menos suscetível a grandes desvios.

### **3.3. Conclusão**

Os resultados obtidos sugerem que a configuração com pertinência trapezoidal e operador máximo busca fornecer um bom equilíbrio entre simplicidade e precisão. Essa combinação apresentou o menor erro médio entre as configurações analisadas, mostrando-se a mais adequada para este caso específico. Por outro lado, configurações como pertinência triangular com operador mínimo evidenciam limitações em lidar com transições complexas da função real.

De forma geral, a otimização do modelo fuzzy Takagi-Sugeno requer atenção cuidadosa na escolha das funções de pertinência e operadores fuzzy, além da configuração

do número de regras. Essa análise reforça a necessidade de personalização das combinações para atender aos requisitos específicos de cada problema, buscando reduzir o erro e aprimorar a aproximação.

## 4. Modelo e Compilação

### 4.1. Reprodução

Para garantir a reprodutibilidade dos resultados deste estudo sobre aproximações fuzzy no modelo Takagi-Sugeno, todo o código-fonte, incluindo as implementações das funções de pertinência, operadores fuzzy e a configuração do sistema de primeira ordem, estão disponíveis em um repositório público no GitHub[2]. O diretório `src/` organiza os scripts em módulos Python, permitindo tanto a reprodução direta dos resultados apresentados quanto a execução de novos experimentos. Buscando assegurar a consistência dos resultados, o código não será alterado após a data de entrega do trabalho, com intuito de garantir que as análises descritas permaneçam alinhadas com os scripts disponíveis.

### 4.2. Execução

O programa realiza automaticamente as análises do sistema fuzzy Takagi-Sugeno, testando três combinações principais de funções de pertinência (Gaussiana, Triangular, Trapezoidal) e operadores fuzzy (Produto, Mínimo, Máximo). A execução é feita por meio do script principal `main.py`, sem necessidade de configurações manuais. Os resultados incluem métricas de erro (RMSE), gráficos comparativos e gráficos de erro, sendo exibidos no terminal e na tela.

Para executar, siga os passos abaixo:

- Certifique-se de que `numpy`, `matplotlib` e `scipy` estão instalados:

```
pip install numpy matplotlib scipy
```

- Execute o programa com:

```
python main.py
```

Os resultados serão exibidos na tela e armazenados automaticamente.

## Referências

- [1] Alisson Marques da Silva. Inteligência computacional - sistemas fuzzy: Inferência 02. Apresentação de slides, 2024. Disponível para acesso mediante solicitação.
- [2] Celso Vinícius. <https://github.com/celzin/IC-Atividade-02>, 2024. Acesso em: 19 nov. 2024.