

Estudo de Casos para Consolidação de Servidores na Computação em Nuvem baseada em ANFIS

Juliano Buss¹, Bruna Novack¹, Héliida Santos², Giancarlo Lucca³,
Lizandro Oliveira³, Bruno Moura⁴, Rafael Bastos¹, Anderson Cruz⁵,
Adenauer Yamin¹, Renata Reiser¹

¹Universidade Federal do Pelotas (UFPel), Pelotas-RS, Brasil

²Universidade Federal do Rio Grande (FURG), Rio Grande-RS, Brasil

³Universidade Católica de Pelotas (UCPel), Pelotas-RS, Brasil

⁴Universidade Federal do Pampa (Unipampa), Bagé-RS, Brasil

⁵Universidade Federal do Rio Grande do Norte (UFRN), Natal-RN, Brasil

{jsbuss, bcdnovack, rrbastos, adenauer, reiser}@inf.ufpel.edu.br

{giancarlo.lucca, lizandro.oliveira}@ucpel.edu.br

helida@furg.br, brunomoura@unipampa.edu.br, anderson@imd.ufrn.br

Abstract. *Cloud Computing is enabled by data centers and computing infrastructures, based on the premise that costs are directly related to resource usage rates, with significant control over energy consumption generated by service demands. This article explores the challenges of achieving server consolidation in Cloud Computing, with an emphasis on optimizing energy consumption. The work aims to develop a flexible approach that promotes the optimization of resource management in Cloud Computing, using Neural Network techniques, considering new results from the application of an adaptive neuro-fuzzy inference system.*

Resumo. *A Computação em Nuvem é viabilizada por centros de dados e infraestruturas computacionais, com a premissa de que os custos estão diretamente relacionados às taxas de uso dos recursos, com controle expressivo do consumo de energia gerado pelas demandas de serviços. Este artigo explora os desafios para obter a consolidação de servidores na Computação em Nuvem, com ênfase na otimização do consumo energético. O trabalho visa desenvolver uma abordagem flexível que promova a otimização do gerenciamento de recursos na Computação em Nuvem, via técnicas de Redes Neurais, considerando novos resultados aplicando um sistema de inferência neuro-fuzzy adaptativo.*

1. Introdução

A consolidação dinâmica de servidores é uma estratégia essencial para melhorar a utilização de recursos e reduzir o consumo energético em *datacenters* de Computação em Nuvem (CN). *Datacenters* são responsáveis por elevados custos de energia, que podem

chegar a milhões de dólares, evidenciando a preocupação com a otimização de recursos para garantir a sustentabilidade [Khan and Zakarya 2021, Saxena and Singh 2024].

No contexto da consolidação de servidores em ambientes de CN, o grupo de pesquisa LUPS/UFPel desenvolveu duas abordagens complementares. A primeira, denominada Int-FLBCC [Moura 2022], fornece a um escalonador uma lista com os níveis de utilização dos servidores físicos, gerada por meio de um Sistema de Inferência Fuzzy (SIF). Com base nessas informações, o sistema realiza o mapeamento das Máquinas Virtuais (MV) entre os servidores físicos disponíveis. A segunda abordagem, denominada Hybrid-FLBCC [Bastos 2025], incorpora técnicas de inteligência computacional para realizar a seleção de atributos, com o objetivo de aprimorar o desempenho do processo.

As Redes Neurais (RN) [Wang 2003] destacam-se como alternativas para o emprego no contexto da consolidação de servidores na CN. Este trabalho propõe a utilização de uma RN integrada aos princípios da Lógica Fuzzy (LF) [Zadeh 1994], com o intuito de aplicar técnicas de aprendizado computacional para viabilizar a consolidação de servidores em ambientes de CN, explorando a Política de Alocação (PA) **Intervalo interquartil (IQR)** e a Política de Seleção (PS) **Seleção Aleatória (RS)** para identificar a MV candidata à migração, bem como o servidor físico de destino [Beloglazov and Buyya 2012].

Utilizando a seleção de atributos [Guyon and Elisseeff 2003] desenvolvida em [Bastos et al. 2024], investiga-se a aplicação do *Adaptive Neuro-Fuzzy Inference System* (ANFIS) [Jang 1993] como mecanismo para prover suporte à consolidação de servidores em ambientes de CN. O diferencial dessa abordagem se dá pelo uso de uma RN que combina o aprendizado adaptativo aos princípios da LF, adaptando Funções de Pertinência (FP) sobre diferentes cenários e provendo a geração automática das regras do SIF, onde em um sistema fuzzy clássico, precisam ser definidas por um especialista.

O artigo está organizado da seguinte forma: A Seção (2) apresenta trabalhos relacionados na aplicação do ANFIS em ambiente de CN. As Seções (3,4,5) discutem os conceitos fundamentais das áreas que serão exploradas neste artigo. A Seção (6) apresenta a metodologia abordada para a execução do ANFIS no ambiente de CN, assim como os resultados da pesquisa. A Seção (7) descreve as considerações finais e trabalhos futuros.

2. Trabalhos relacionados

[Wu et al. 2020b] utiliza ANFIS na previsão da carga de uso de CPU e memória, alocando uma MV com os recursos exatos para suprir a necessidade da aplicação. [Alaei et al. 2021] classifica cargas de trabalho futuras usando o ANFIS, onde dependendo do resultado da regra fuzzy gerada, que representa o nível de utilização dos recursos, diferentes metodologias são aplicadas para realizar o balanceamento de acordo com o nível de uso ou risco de falhas. [Wu et al. 2020a] utiliza ANFIS para prever a carga de utilização da memória e de CPU das tarefas, configurando os recursos das MV de acordo com as necessidades. [Prathiba and Sankar 2024] ordena a fila de tarefas de acordo com seus requisitos, onde ANFIS é utilizado para identificar o estado da MV, organizando a lista de máquinas de acordo com seus recursos disponíveis, pareando tarefas complexas para as máquinas com melhores condições.

3. Lógica Fuzzy

Na LF [Zadeh 1988], as avaliações estão baseadas nas relações de pertinência entre Conjunto Fuzzy (CF), as quais não são rigidamente definidas e generalizam a função característica da abordagem clássica. Neste contexto, um elemento pode pertencer a vários CF, mas com graus de pertinência distintos no intervalo $[0, 1]$ [Zadeh 1965]. Assim, a abordagem fuzzy provê uma modelagem matemática que considera também termos linguísticos subjetivos da linguagem natural (como aproximadamente, em torno de) que viabilizam representar, armazenar e processar em um computador, informações incertas, a exemplo do que faz o ser humano. A formularização matemática da extensão da função característica clássica é expressa pela Definição (3.1).

Definição 3.1 A FP $\mu_A(x): X \rightarrow U$, onde $U \rightarrow [0, 1]$, relaciona o elemento $x \in X$, de forma a definir o quanto ele pertence ao subconjunto A , através do grau de pertinência associado, representado por $\mu_A(x)$, $0 \leq \mu_A(x) \leq 1$. Assim, a expressão clássica de um CF A em um universo $X \neq \emptyset$ é dado pela expressão: $A = \{(x, \mu_A(x)) : x \in X\}$.

Os componentes do SIF são [Ross 2010]: **Base de dados:** Define a FP utilizada nas regras fuzzy; **Base de regras:** Contém as regras fuzzy se-então; **Fuzzificação:** Utiliza as FP pré-estabelecidas para mapear cada variável de entrada do sistema em graus de pertinência de cada CF; **Inferência fuzzy:** Processa as regras fuzzy para derivar conclusões sobre as entradas fuzzy; **Defuzzificação:** Converte os resultados fuzzy produzidos pela inferência em um valor nítido, utilizando FP na parte consequente das regras.

A Figura 1 apresenta a arquitetura de um SIF, destacando o módulo de regras se-então, que fornece a base para a tomada de decisões e explicita as relações entre entradas e saídas [Jang et al. 1997], permitindo raciocinar sobre as informações e realizar inferências lógicas a partir dos dados.

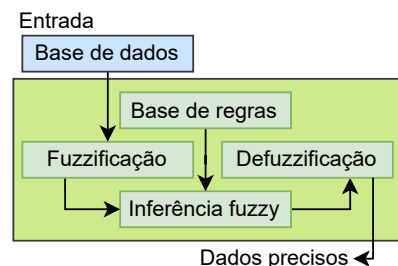


Figura 1. SIF-Arquitetura.

4. Computação em Nuvem

A CN é um paradigma computacional em que grandes empresas, como Google e Microsoft, fornecem abstração para categorias como infraestrutura, plataforma e software como serviço, oferecendo serviços digitais hospedados em *datacenters* [Kavis 2014].

A consolidação de servidores busca otimizar o uso dos recursos disponíveis, simultaneamente reduzindo o consumo energético na CN. Por meio de técnicas de virtualização, múltiplas MV subutilizadas podem ser consolidadas em menos servidores físicos, elevando a taxa média de utilização dos recursos computacionais. A migração de MV é um procedimento amplamente empregado para atingir essa consolidação dinâmica [Uddin et al. 2021]. Ao realocar MV de servidores ociosos para um único servidor mais eficiente, é possível desligar os servidores excedentes, economizando energia.

Diferentes métodos de consolidação de servidores clássicos são apresentados na literatura [Beloglazov and Buyya 2012, Ahmad et al. 2020]. As PA têm como função identificar hosts sobrecarregados, avaliando a lista de servidores físicos disponíveis.

Quando detectada uma sobrecarga, é gerada uma lista das MV que precisam ser migradas, processo realizado com base na PS. A seguir, apresentam-se, respectivamente, a PA e PS consideradas para este trabalho [Beloglazov and Buyya 2012]: **IQR**: O limite de utilização de CPU é adaptado com base em uma medida estatística, no intervalo inter-quartil, sendo igual à diferença entre o terceiro quartil e o primeiro quartil; **RS**: Seleciona uma MV alocada no host para ser migrada de acordo com uma variável aleatória.

5. Conceitos do ANFIS

A arquitetura e o processo de aprendizagem subjacentes ao ANFIS consistem em um SIF implementado na estrutura de RN adaptativa [Wang 2003]. Essas abordagens de computação flexíveis são amplamente aplicadas para modelar o comportamento de especialistas, integrando o conhecimento humano fornecido pelo SIF (na forma de regras fuzzy se-então) e algoritmos de aprendizagem híbridos [Walia et al. 2015].

ANFIS considera funções não lineares para controlar parâmetros essenciais, seguindo o modelo de inferência Takagi-Sugeno [Takagi and Sugeno 1985]. A seguir, apresenta-se a descrição das etapas e camadas que compõem a sua arquitetura, considerando uma rede com i nodos por camada, em que o i -ésimo nodo da camada $\mathcal{K}$ é denotado pelo seu sinal de saída $O_i^{\mathcal{K}}$ [Jang 1993]:

- 1 **Camada de fuzzificação**: Cada neurônio nesta camada representa uma FP na premissa das regras. Esta etapa converte valores de entrada nítidos em um conjunto de variáveis fuzzy. Os graus de associação são dados: $O_i^1 = \mu_{A_i}(x)$, onde μ_{A_i} é a FP calculada pelo i -ésimo neurônio e x é uma entrada do sistema, e A_i é a variável linguística (baixo, médio, alto) associada ao nodo. O_i^1 é a FP de A_i , representando o quanto x pertence em A_i ;
- 2 **Camada de regra (Antecedente)**: Cada neurônio correspondente a uma regra fuzzy calcula a força de disparo (peso) da regra usando um operador de agregação conjuntivo fuzzy (mínimo, máximo, produto), unificando os sinais de entrada: $O^2 = \omega_i = \mu_{A_i}(x) \cdot \mu_{B_i}(y)$, a saída do nodo representa a força de ativação da regra, utilizando um agregador fuzzy que satisfaça as propriedades do operador AND (t-norma) ou OR (t-conorma);
- 3 **Camada de normalização (Normalização)**: Normaliza os graus de ativação da regra, garantindo a soma de seus pesos seja 1: $O^3 = \bar{\omega}_i = \frac{\omega_i}{\omega_1 + \omega_2}$, onde ω representa os pesos das regras fuzzy no ANFIS;
- 4 **Camada de defuzzificação (Consequente)**: Considerando o peso calculado da camada anterior, a contribuição da regra é dada multiplicando o valor na camada de normalização $\bar{\omega}_i$ com parâmetros do consequente da regra f_i , dado como: $O_i^4 = \bar{\omega}_i \cdot f_i = \bar{\omega}_i \cdot (p_i \cdot x + q_i \cdot y + r_i)$, onde $\{p_i, q_i, r_i\}$ formam um conjunto de parâmetros, uma função linear;
- 5 **Camada de saída (Soma)**: obtém um único número da saída do CF agregado e realiza uma soma de todos os sinais de entrada calculados pelos neurônios. A agregação das saídas das regras fuzzy, o consequente da regra, é unificada através da média ponderada das saídas da regra: $O_i^5 = \sum_i \bar{\omega}_i \cdot f_i = \frac{\sum_i \omega_i \cdot f_i}{\sum_i \omega_i}$.

6. Modelo proposto

A Figura 3, apresenta as etapas consideradas desde a leitura do *datasets* envolvendo diferentes configurações no ambiente de CN, até o final da execução do modelo ANFIS, obtendo métricas de avaliação do modelo.

Uma representação gráfica das camadas que formam o ANFIS é apresentada na Figura 2. As setas entre as camadas indicam o fluxo do processamento: Da camada 1 à 2, cada entrada é avaliada em todas as FP; Da 2 à 3, os pesos ω_i são normalizados pela soma total; Da 3 à 4, aplica-se o consequente das regras ponderado pelos pesos; E da 4 à 5, soma-se ponderadamente as saídas fuzzy, resultando no valor defuzzificado pela média ponderada das equações lineares da camada 4.

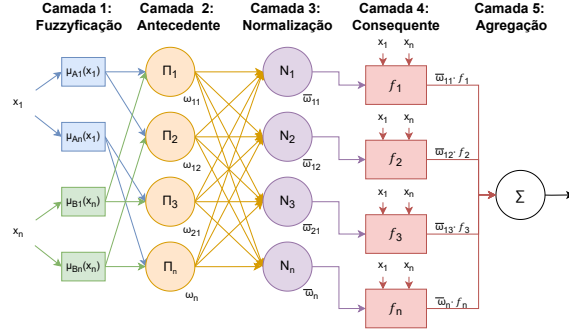


Figura 2. Estrutura Sugeno do modelo ANFIS. Adaptado de [Jang 1993, Shoeibi et al. 2022].

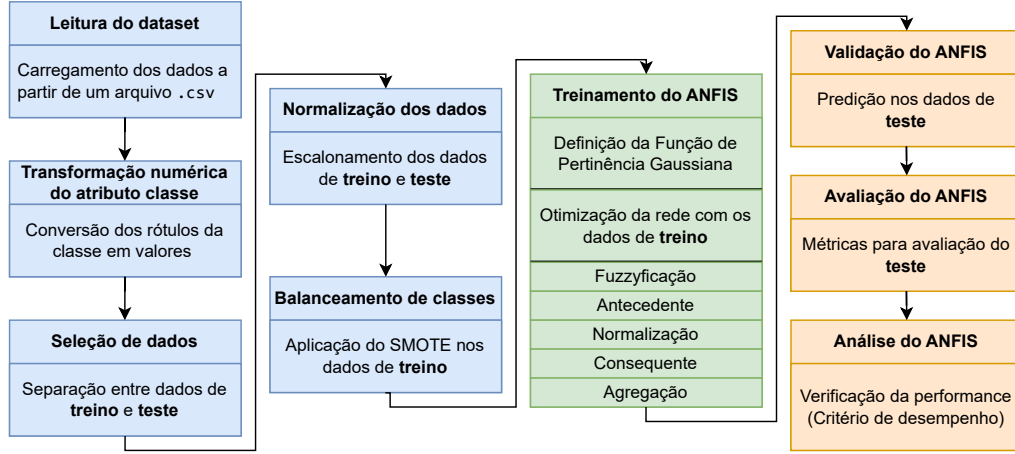


Figura 3. Fluxograma das etapas de execução do modelo ANFIS.

A PA e PS escolhidas são IQR e RS, simulados através do Cloudsim [Calheiros et al. 2011], utilizando cargas de trabalho e dados do projeto PlanetLab¹, empregando dados do mundo real do projeto CoMon [Park and Pai 2006]. Uma seleção de atributos é aplicada para selecionar um subconjunto de dados que melhor represente o *dataset* completo obtido através do CloudSim, removendo dados irrelevantes ou redundantes. A primeira iteração da seleção de variáveis verifica o atributo que apresenta o melhor desempenho de classificação. Na próxima iteração, essa variável é combinada com outras em busca da melhor combinação. O processo é repetido até que não haja mais nenhuma melhoria no desempenho de classificação.

Os *datasets* considerados possuem informações referentes ao uso de recursos das máquinas do ambiente de CN, representando o estado do host físico a cada 300 segundos, incluindo os seguintes atributos capturados, **CPU**: Uso do processador; **Memória**: Nível de utilização da memória RAM; **Banda**: Uso da banda larga; **Armazenamento**: Armazenamento disponível; **Energia**: Consumo de energia; **MIPS**: Medida de desempenho de um processador; **Classe**: Rótulo de utilização do host físico (subutilizado - C_0 , regular - C_1 ou sobrecarregado - C_2). O *dataset A* é composto pelos atributos: CPU, Memória,

¹<https://planetlab.cs.princeton.edu>

Banda, Armazenamento; e o **dataset B**: CPU, Energia, Armazenamento, Banda; ambos com 20.160 amostras totais. O uso destes *datasets* é dado pela extensão do trabalho anterior [Bastos et al. 2024], abordando ANFIS para a consolidação de servidores.

A separação dos dados entre treino e teste considera a leitura integral do *dataset*, gerando subconjuntos de dados, um contendo 80% das amostras para treino da rede, e outro com 20% das amostras para teste [Singh and Lone 2020]. Uma normalização *Min-Max* sobre os subconjuntos possibilita tanto a execução do modelo ANFIS para estes dados, quanto formatando os dados de um modo que sejam aplicáveis nas FP.

A técnica de sobreamostragem minoritária sintética (SMOTE) [Chawla et al. 2002] busca nivelar a quantidade de dados quando analisados pelas suas classificações, transformando um *dataset* desbalanceado em balanceado [Sun et al. 2009]. Sua aplicação se dá nos dados presentes no subconjunto de treino do modelo, contendo o vazamento de dados, evitando que informações do subconjunto de teste estejam misturados na construção do aprendizado do modelo [Lemaître et al. 2024].

A Figura 4 apresenta as funções Gaussianas para todos os atributos do *dataset*, identificando as variáveis linguísticas baixo (azul), médio (verde) e alto (vermelho), com parâmetros de média (μ) e desvio padrão (σ) indicados para cada função [Saatchi 2024], se aproximando do comportamento das funções exploradas nos trabalhos anteriores do grupo [Moura 2022].

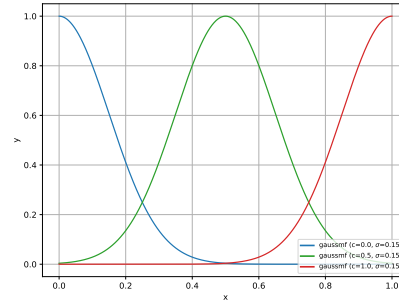


Figura 4. FP Gaussianas iniciais.

Duas épocas de treinamento são consideradas para a execução, explorando a adaptação do ANFIS através dos algoritmos híbridos nas etapas de propagação direta e retropropagação [Jang 1993, Karaboga and Kaya 2019], sendo suficiente para gerar resultados satisfatórios logo após a primeira época de treino e execução dos algoritmos [Jang 1996]. Para obter as métricas de avaliação sobre a capacidade de classificação do modelo, é realizado um pós-processamento de discretização dos valores da saída do ANFIS, obtendo valores inteiros aplicáveis às métricas. Essa abordagem está alinhada ao discutido em [Kuhn et al. 2013], destacando situações em que, por razões práticas, pode ser apropriado transformar uma saída contínua em categórica. Apesar de existir perda de informação, ela viabiliza a aplicação de métricas de avaliação da classificação.

Para os dois *datasets* analisados, as Tabelas 1 e 2 apresentam as Matrizes de Confusão (MC) [Sathyanarayanan and Tantri 2024, Markoulidakis et al. 2021], calculadas a partir de 20% das amostras do *dataset* completo, o subconjunto de teste. Nessas matrizes, as linhas representam as classes reais (C_i), e a coluna (C_j) as classes previstas pelo modelo, onde cada célula ($C_{i,j}$) indica o número de amostras com uma classe real (linha) classificadas como uma classe prevista (coluna). Os valores na diagonal principal correspondem às classificações corretas. As métricas extraídas da MC são calculadas para cada classe N do *dataset* ($C_1, C_2, \dots, C_i, \dots, C_N$): **Verdadeiro Positivo (VP)**: Classe real é positiva e modelo previu positivo; **Falso Positivo (FP)**: Classe real é negativa, mas modelo previu positivo; **Verdadeiro Negativo (VN)**: Classe real é negativa e modelo previu negativo; e

Falso Negativo (FN): Classe real é positiva, mas modelo previu negativa.

Tabela 1. MC - dataset A.

	C_0	C_1	C_2
C_0	258	8	15
C_1	421	3311	11
C_2	5	0	3

Tabela 2. MC - dataset B.

	C_0	C_1	C_2
C_0	269	10	2
C_1	238	3491	14
C_2	0	0	8

As Tabelas 3 e 4 apresentam as métricas de avaliação do modelo, calculadas individualmente para cada classe C_N do *dataset*, obtidas através do Scikit-learn [Pedregosa et al. 2025b], com arredondamento padrão do método. A Tabela contém: **Classe:** Nível de utilização presente no *dataset*; **P:** Precisão, mede quantas das previsões positivas feitas estavam corretas $\left(\frac{VP(C_i)}{VP(C_i)+FP(C_i)}\right)$; **R:** Revogação, indica quantos dos casos realmente positivos foram identificados corretamente pelo modelo $\left(\frac{VP(C_i)}{VP(C_i)+FN(C_i)}\right)$ **F1:** F1-score, média entre precisão e revogação $\left(2 \cdot \frac{P(C_i) \cdot R(C_i)}{P(C_i) + R(C_i)}\right)$ [Markoulidakis et al. 2021]. A acurácia apresenta a proporção de previsões corretas em relação ao total de amostras $\left(\frac{\sum_{i=1}^N VP(C_i)}{\sum_{i=1}^N \sum_{j=1}^N C_{i,j}}\right)$ [Markoulidakis et al. 2021, Pedregosa et al. 2025a].

Tabela 3. Métricas - dataset A.

Classe	P	R	F1
Subutilizado (C_0)	0.38	0.92	0.53
Regular (C_1)	1.00	0.88	0.94
Sobrecarregado (C_2)	0.10	0.38	0.16

Tabela 4. Métricas - dataset B.

Classe	P	R	F1
Subutilizado (C_0)	0.53	0.96	0.68
Regular (C_1)	1.00	0.93	0.96
Sobrecarregado (C_2)	0.33	1.00	0.50

A Curva Característica de Operação do Receptor (ROC) [Tan et al. 2005, Fawcett 2006] é um método utilizado para avaliar o desempenho em classificações binárias. A curva é construída a partir da variação de um limiar de decisão, representando graficamente a relação entre a Taxa de Verdadeiros Positivos (TPR) no eixo Y e a Taxa de Falsos Positivos (FPR) no eixo X. Cada ponto na curva ROC reflete o comportamento do classificador para um limiar, verificando a capacidade do modelo. A Área Sob a Curva ROC (AUC) é uma métrica que quantifica a performance do classificador. Esta métrica de avaliação foi considerado conforme o trabalho anterior [Bastos et al. 2024].

ANFIS gera um único valor para cada dado presente no subconjunto de teste, que não é necessariamente um número inteiro correspondente a alguma classe, que representa diferentes estados de utilização das máquinas no ambiente de CN. A literatura apresenta as métricas de AUC e acurácia na avaliação do ANFIS em diferentes áreas do conhecimento [Tien Bui et al. 2018, Kusumadewi et al. 2023, Costache et al. 2020].

A metodologia adotada em [Toosi et al. 2006] é considerada para converter a saída do modelo ANFIS em classes, viabilizando a avaliação via métricas de desempenho, e entre elas, a curva ROC. Considera-se a utilização de um parâmetro Γ , convertendo as múltiplas opções em um problema binário. Neste contexto, caso o valor arredondado da

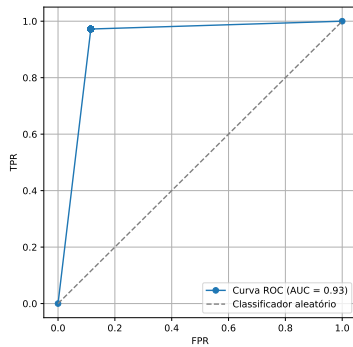


Figura 5. Curva ROC para o *data-set A* (AUC = 0.93).

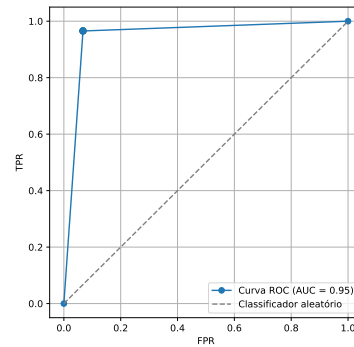


Figura 6. Curva ROC para o *data-set B* (AUC = 0.95).

saída ANFIS pelo parâmetro Γ seja equivalente ao número de alguma das classes de ataque apresentada no artigo (0 - normal, 1 - Probe, 2 - Dos, 3 - U2R, 4 - R2L), a classificação é associada em (1); Caso contrário, se o valor de saída da rede for arredondado para (0), preserva-se a classificação em (0). Para analisar o impacto deste parâmetro Γ no desempenho de classificação, a metodologia propõe variar Γ em um intervalo $[0, 0.5]$.

Para cada valor de Γ , realiza-se uma classificação das instâncias, e são calculadas as métricas da MC de VP, FP, VN e FN. Com base nestes conceitos, calcula-se para cada Γ as taxas correspondentes de $TPR = \frac{VP}{VP + FN}$ e $FPR = \frac{FP}{FP + VN}$ considerando a transformação do problema em binário. Essa curva mostra graficamente a relação entre a TPR e a FPR à medida que Γ varia, avaliando o desempenho sobre diferentes limiares.

Na conversão, as classes (subutilizado - 0 e sobrecarregado - 2) são agrupadas como (1), e a classe (regular - 1) recebe a conversão final binária (0), avaliando o nível normal de utilização da máquina (1), contra (0 e 2), que representa o cenário de necessidade de migração. Nesta conversão binária, calculam-se as taxas de TPR e FPR referente a cada valor de Γ . Nas Figuras 5 e 6, são apresentadas as curvas ROC obtidas para os dois *datasets A* e *B* nesta abordagem. Cada imagem demonstra a variação da TPR em função da FPR para diferentes valores do parâmetro Γ , juntamente com a métrica AUC.

7. Conclusão

Neste trabalho, foi apresentado a utilização do ANFIS sobre dois diferentes casos na CN, apresentando as métrica de avaliação do modelo adaptada sobre os resultados de teste do ANFIS. Apesar de ainda existirem parâmetros passíveis de alterações, a acurácia geral do *dataset A* foi de 88.59%, e 93.45% para o *dataset B*, indicando um desempenho promissor nas condições analisadas. As FP adaptadas e regras do SIF geradas pelo ANFIS serão exploradas em trabalhos futuros, realizando a recomendação de migração em ambiente de CN através de configurações geradas e adaptadas automaticamente pela RN.

Agradecimentos

Os autores agradecem as seguintes agências de fomento: CAPES, CNPq (309160/2019-7; 311429/2020-3, 3305805/2021-5, 150160/2023-2), FAPERGS (24/2551-0001396-2, 24/2551-0000631-1), PqG/FAPERGS (21/2551-0002057-1) e FAPERGS/CNPq (23/2551-0000126-8), PRONEX (16/2551-0000488-9).

Referências

- Ahmad, B., Maroof, Z., McClean, S., Charles, D., and Parr, G. (2020). Economic impact of energy saving techniques in cloud server. *Cluster Computing*, 23(2):611–621.
- Alaei, M., Khorsand, R., and Ramezanpour, M. (2021). An adaptive fault detector strategy for scientific workflow scheduling based on improved differential evolution algorithm in cloud. *Applied Soft Computing*, 99:106895.
- Bastos, R., Seibert, V., Maia, G., de Moura, B. P., Lucca, G., Yamin, A., and Reiser, R. (2024). Exploratory data analysis in cloud computing environments for server consolidation via fuzzy classification models. In *Proceedings of the 26th International Conference on Enterprise Information Systems*, pages 636–643. INSTICC, SCITEPRESS - Science and Technology Publications.
- Bastos, R. R. (2025). *Hybrid-FLBCC: Uma Abordagem para o Gerenciamento de Recursos na Computação em Nuvem integrando a Lógica Fuzzy Valorada Intervalarmente e Aprendizado de Máquina*. Tese, Centro de Desenvolvimento Tecnológico. Universidade Federal de Pelotas, Pelotas. Tese de Doutorado em desenvolvimento.
- Beloglazov, A. and Buyya, R. (2012). Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. *Concurrency and Computation: Practice and Experience*, 24(13):1397–1420.
- Calheiros, R. N., Ranjan, R., Beloglazov, A., De Rose, C. A., and Buyya, R. (2011). Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software: Practice and experience*, 41(1):23–50.
- Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.
- Costache, R., Țîncu, R., Elkhachy, I., Pham, Q. B., Popa, M. C., Diaconu, D. C., Avand, M., Costache, I., Arabameri, A., and Bui, D. T. (2020). New neural fuzzy-based machine learning ensemble for enhancing the prediction accuracy of flood susceptibility mapping. *Hydrological Sciences Journal*, 65(16):2816–2837.
- Fawcett, T. (2006). An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874.
- Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182.
- Jang, J., Sun, C., and Mizutani, E. (1997). Neuro-fuzzy and soft computing-a computational approach to learning and machine intelligence. *IEEE Transactions on Automatic Control*, 42(10):1482–1484.
- Jang, J.-S. (1993). Anfis: adaptive-network-based fuzzy inference system. *IEEE transactions on systems, man, and cybernetics*, 23(3):665–685.
- Jang, J.-S. (1996). Input selection for anfis learning. In *Proceedings of IEEE 5th International Fuzzy Systems*, volume 2, pages 1493–1499 vol.2.

- Karaboga, D. and Kaya, E. (2019). Adaptive network based fuzzy inference system (ANFIS) training approaches: a comprehensive survey. *Artificial Intelligence Review*, 52:2263–2293.
- Kavis, M. J. (2014). *Architecting the cloud: design decisions for cloud computing service models (SaaS, PaaS, and IaaS)*. John Wiley & Sons.
- Khan, A. A. and Zakarya, M. (2021). Energy, performance and cost efficient cloud data-centres: A survey. *Computer Science Review*, 40:100390.
- Kuhn, M., Johnson, K., et al. (2013). *Applied predictive modeling*, volume 26. Springer.
- Kusumadewi, S., Rosita, L., and Wahyuni, E. G. (2023). Stability of classification performance on an adaptive neuro fuzzy inference system for disease complication prediction. *Int J Artif Intell ISSN*, 2252(8938):8938.
- Lemaître, G., Nogueira, F., and Aridas, C. (2024). Common pitfalls — imbalanced-learn 0.11.0 documentation. https://imbalanced-learn.org/stable/common_pitfalls.html. Acesso em: 13 junho. 2025.
- Markoulidakis, I., Kopsiaftis, G., Rallis, I., and Georgoulas, I. (2021). Multi-class confusion matrix reduction method and its application on net promoter score classification problem. In *Proceedings of the 14th Pervasive technologies related to assistive environments conference*, pages 412–419.
- Moura, B. (2022). *Uma Abordagem Flexível para Consolidação Dinâmica de Servidores na Computação em Nuvem Explorando Lógica Fuzzy Valorada Intervalarmente*. Tese, Centro de Desenvolvimento Tecnológico. Universidade Federal de Pelotas, Pelotas. Disponível em: <https://guaiaca.ufpel.edu.br/handle/prefix/8733>. Acesso em: 11 março. 2025.
- Park, K. and Pai, V. S. (2006). Comon: a mostly-scalable monitoring system for planetlab. *ACM SIGOPS operating systems review*, 40(1):65–74.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2025a). `sklearn.metrics.accuracy_score` - scikit-learn 1.7.1 documentation. Acesso em: 27 agosto. 2025.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2025b). `sklearn.metrics.classification_report` - scikit-learn 1.7.1 documentation. Acesso em: 27 agosto. 2025.
- Prathiba, S. and Sankar, S. (2024). Energy-efficient resource allocation in cloud infrastructure using L3F-MGA and E-ANFIS. *Measurement: Sensors*, 31:100965.
- Ross, T. J. (2010). *Classical Relations and Fuzzy Relations*. John Wiley and Sons, Ltd, USA.
- Saatchi, R. (2024). Fuzzy logic concepts, developments and implementation. *Information*, 15(10).

- Sathyanarayanan, S. and Tantri, B. R. (2024). Confusion matrix-based performance evaluation metrics. *African Journal of Biomedical Research*, pages 4023–4031.
- Saxena, D. and Singh, A. K. (2024). A comprehensive survey on sustainable resource management in cloud computing environments.
- Shoeibi, A., Ghassemi, N., Khodatars, M., Moridian, P., Alizadehsani, R., Zare, A., Khosravi, A., Subasi, A., Acharya, U. R., and Gorriz, J. M. (2022). Detection of epileptic seizures on eeg signals using anfis classifier, autoencoders and fuzzy entropies. *Bio-medical Signal Processing and Control*, 73:103417.
- Singh, H. and Lone, Y. A. (2020). Deep neuro-fuzzy systems with python. *Apress, Berkeley*.
- Sun, Y., Wong, A. K., and Kamel, M. S. (2009). Classification of imbalanced data: A review. *International journal of pattern recognition and artificial intelligence*, 23(04):687–719.
- Takagi, T. and Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *IEEE transactions on systems, man, and cybernetics*, pages 116–132.
- Tan, P.-N., Steinbach, M., and Kumar, V. (2005). *Introduction to Data Mining, (First Edition)*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- Tien Bui, D., Khosravi, K., Li, S., Shahabi, H., Panahi, M., Singh, V. P., Chapi, K., Shirzadi, A., Panahi, S., Chen, W., and Bin Ahmad, B. (2018). New hybrids of anfis with several optimization algorithms for flood susceptibility modeling. *Water*, 10(9).
- Toosi, A. N., Kahani, M., and Monsefi, R. (2006). Network intrusion detection based on neuro-fuzzy classification. In *2006 International Conference on Computing & Informatics*, pages 1–5. IEEE.
- Uddin, M., Hamdi, M., Alghamdi, A., Alrizq, M., Memon, M. S., Abdelhaq, M., and Alsaqour, R. (2021). Server consolidation: A technique to enhance cloud data center power efficiency and overall cost of ownership. *International Journal of Distributed Sensor Networks*, 17(3):1550147721997218.
- Walia, N., Singh, H., and Sharma, A. (2015). ANFIS: Adaptive Neuro-Fuzzy Inference System - A Survey. *International Journal of Computer Applications*, 123(13).
- Wang, S.-C. (2003). *Artificial Neural Network*, pages 81–100. Springer US, Boston, MA.
- Wu, X., Wang, H., Tan, W., Wei, D., and Shi, M. (2020a). Dynamic allocation strategy of VM resources with fuzzy transfer learning method. *Peer-to-Peer Networking and Applications*, 13(6):2201 – 2213.
- Wu, X., Wang, H., Wei, D., and Shi, M. (2020b). ANFIS with natural language processing and gray relational analysis based cloud computing framework for real time energy efficient resource allocation. *Computer Communications*, 150:122–130.
- Zadeh, L. A. (1965). Fuzzy sets. *Information and control*, 8(3):338–353.
- Zadeh, L. A. (1988). Fuzzy logic. *Computer*, 21(4):83–93.
- Zadeh, L. A. (1994). Fuzzy logic, neural networks, and soft computing. *Communications of the ACM*, 37(3):77–84.