Linear Combinations of Bounded versus Unbounded Activation Functions: A Comparative Study on Noise Robustness

Rogério Ferreira de Moraes[1], **Natália Azeredo Viana da Silva**[1], **Nattan Gomes Silva Cortes**[1]

[1]Instituto Politécnico – Universidade Federal do Rio de Janeiro (UFRJ)
Rua Aloisio da Silva Gomes, 50 – Granja dos Cavaleiros
27.930-560 – Macaé – RJ – Brasil

rogeriofm@macae.ufrj.br, natalia.azeredo.eng@gmail.com
nattango132@gmail.com

Abstract. *Real tabular data suffer from measurement noise and missing attributes, yet the literature on trainable activations optimises accuracy on clean data, while the robustness literature addresses only fixed activations. We investigate this gap: does bounding the codomain of the primitives make the ensemble robust? On the 72 OpenML-CC18 tasks, under Gaussian, Uniform and Masking noise at five intensities, we compare trainable linear combinations that are bounded (C19, C8), unbounded (C22), mixed (C41), and ReLU. Without noise, C22 and C41 lead; under noise the picture reverses and bounded ensembles prevail ($d \approx 0.5$ under masking). The C41 mixture does not inherit the advantage: the benefit requires uniformly bounding every primitive.*

Resumo. *Dados tabulares reais sofrem ruído de medição e perda de atributos, mas a literatura de ativações treináveis otimiza acurácia em dados limpos, enquanto a de robustez trata apenas ativações fixas. Investigamos essa lacuna: a limitação do contradomínio das primitivas confere robustez ao ensemble? Nas 72 tarefas do OpenML-CC18, sob ruído Gaussiano, Uniforme e Masking em cinco intensidades, comparamos combinações lineares treináveis limitadas (C19, C8), não limitadas (C22), mistas (C41) e a ReLU. Sem ruído, C22 e C41 lideram; sob ruído o quadro se inverte e as limitadas prevalecem ($d \approx 0{,}5$ sob masking). A mistura C41 não herda a vantagem: o benefício exige limitação uniforme de todas as primitivas.*

1. Introdução

Multi-Layer Perceptrons (MLPs) consolidaram-se como ferramentas centrais em aprendizado supervisionado, cuja capacidade de mapear fronteiras de decisão complexas depende fundamentalmente das funções de ativação. Tradicionalmente, funções fixas como ReLU oferecem eficiência e mitigam problemas de gradiente, contudo, limitam a flexibilidade adaptativa da topologia. Para superar isso, a literatura propõe funções de ativação *treináveis* — ativações cujos parâmetros são otimizados junto aos pesos da rede. Elas se dividem em duas vertentes: as *paramétricas*, que aprendem poucos escalares de uma forma funcional fixa (p. ex. a PReLU, que aprende a inclinação do ramo negativo), e

as por *combinação linear*, que aprendem os pesos de uma mistura de várias primitivas [Li et al. 2022, Dubey et al. 2022, Apicella et al. 2021]. Este trabalho adota a segunda vertente.

Entretanto, o ganho de acurácia em cenários ideais não garante resiliência a corrupções reais em dados tabulares, como ruídos de medição ou perdas estruturais. O desempenho sob estresse é influenciado pela geometria da ativação: famílias não limitadas preservam atributos, mas podem amplificar *outliers*, enquanto funções limitadas atuam como filtros, contendo anomalias ao custo de restrições representacionais. Investigamos este *trade-off* comparando a robustez de combinações lineares de funções limitadas versus não limitadas. Avaliamos três perturbações complementares, escolhidas por representarem falhas recorrentes em dados tabulares: ruído **Gaussiano** (erro de medição de sensores), ruído **Uniforme** (quantização e arredondamento) e ***Feature Masking*** (perda de atributos por falha de coleta ou registro ausente). Quantificamos a resiliência pelo *Robustness Gap* (Gap_R), a queda de acurácia da estratégia ao passar do teste limpo para o teste perturbado — quanto menor, mais robusta a estratégia (Equação 5). Nossas contribuições são:

- Avaliação sistemática no *benchmark* OpenML-CC18 (72 *datasets*) [Bischl et al. 2021], contrastando *ensembles* limitados (famílias C19, C8), não limitados (C22), mistos (C41) e o *baseline* ReLU.
- Análise do *Robustness Gap* sob ruídos Gaussiano, Uniforme e *Feature Masking*.
- Demonstração estatística de que o confinamento do contradomínio das funções atua como defesa intrínseca contra a degradação estrutural do modelo em cenários de alta incerteza.

O restante deste artigo estrutura-se na fundamentação teórica (Seção 2), metodologia (Seção 3), resultados experimentais (Seção 4) e conclusões (Seção 5).

2. Trabalhos Relacionados

A literatura relacionada a este estudo interseciona três áreas principais: o desenvolvimento de ativações treináveis, a contenção empírica de falhas por limitação de amplitude e a robustez teórica via constante de Lipschitz.

Funções de ativações treináveis, diferente das funções fixas, são funções que são modificadas durante o treinamento, de forma que melhor se adaptam aos dados. Dentre essas funções, podemos destacar as parametrizáveis e os métodos baseados em combinar um conjunto de funções de ativação [Apicella et al. 2021, Sütfeld et al. 2018], precursores diretos da formulação aqui adotada; mais especificamente, adotamos a mesma de [de Moraes and Evangelista 2023] para combinar funções.

A literatura recente no entanto confirma que, embora a flexibilidade das ativações treináveis melhore a expressividade, ela introduz vulnerabilidades sob dados corrompidos se não houver um confinamento matemático adequado. [Ru et al. 2020] mostram que, sob *bit-flips* de memória, a ReLU clássica propaga integralmente pré-ativações anômalas, que passam a dominar as camadas seguintes e destroem a classificação. Impor um teto de saturação (*Clipped-ReLU* ou funções saturantes puras) corta a magnitude do erro na origem, melhorando substancialmente a tolerância a falhas com impacto quase nulo na acurácia em dados limpos.

[Schorn et al. 2021] propõem, para CNNs sujeitas a erros transitórios de memória, uma estrutura de segurança funcional baseada em supervisão de intervalos de ativação. A solução não é adotar uma ativação limitada genérica, mas monitorar as ativações originais e impor *thresholds* por meio de camadas de restrição especializadas, recuperando o desempenho sem retreinar a rede.

[Molina et al. 2020] estudam ativações treináveis e concluem que, embora a flexibilidade aumente a adaptabilidade, ela introduz grave instabilidade sob distribuições de entrada altamente dinâmicas e regimes intensos de atualização. A solução proposta é uma variante racional que restringe estruturalmente a escala máxima da saída (forçando o denominador a ser maior que o numerador) para preservar a estabilidade.

[Bhattacharya et al. 2026] apresenta a ArcGate, uma função de ativação paramétrica treinável que utiliza a arco-tangente para garantir estabilidade numérica. Ela é capaz de se adaptar de forma autônoma às necessidades de cada camada da rede e proporciona uma transição suave que protege o modelo contra ruídos e perturbações de sinal, mantendo um alto desempenho em tarefas de observação da Terra.

No contexto de mitigação de ruídos, a imposição de restrições geométricas às funções da rede tem se mostrado eficaz. [Pauli et al. 2021] restringem ativamente a sensibilidade da rede a perturbações usando a constante de Lipschitz do mapeamento entrada-saída como medida de resiliência: explorando a *slope-restriction* natural das ativações, formulam um Programa Semidefinido (SDP) para estimar seu limite superior e o integram ao treinamento via ADMM. Os resultados confirmam que limitar matematicamente a amplitude de variação das ativações melhora a resiliência a ruídos gaussianos e uniformes, com garantias teóricas e sem perda expressiva de acurácia.

3. Metodologia

3.1. *Ensembles* de Funções de Ativação Treináveis

Neste trabalho usamos 41 funções de ativação primitivas, organizadas nas famílias da Figura 1: **C22** (não limitadas), **C19** (limitadas), **C8** $\subset$ C19 (as estritamente crescentes), **C41** $=$ C22 $\cup$ C19 (mistura) e a ReLU isolada como linha de base (*Original*). As primitivas seguem [de Moraes and Evangelista 2023, Liew et al. 2016, Xu et al. 2016, LeCun et al. 1998, Mishra et al. 2017].

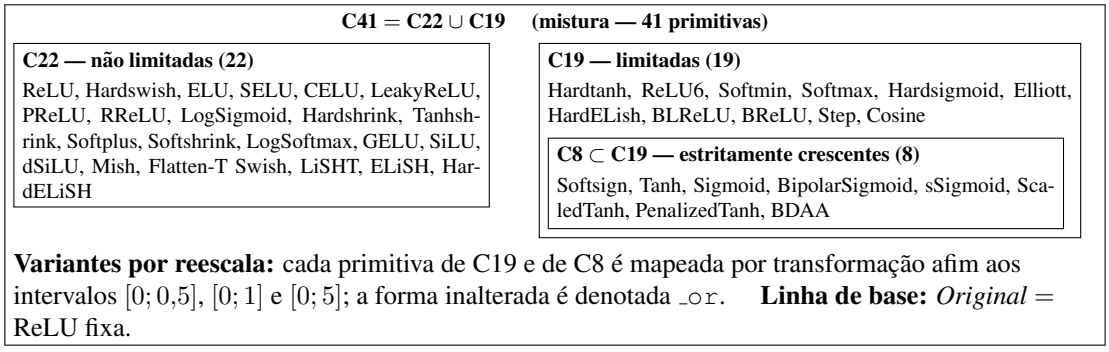


Figura 1. Composição das famílias de primitivas. C8 é subconjunto estrito de C19 (as 8 funções estritamente crescentes); C41 é a união de C22 e C19. As 11 funções listadas em C19 fora da caixa interna são as não estritamente crescentes.

Para reprodutibilidade, destacam-se quatro formulações específicas da C19: ScaledTanh implementada como $\sigma(A \cdot \tanh(B \cdot x))$, com parâmetros aprendíveis A e B inicializados em 1 (limitada em $(0, 1)$, adaptando [LeCun et al. 1998]); HardELish definida por $\mathrm{clip}((x + 1)/2,\, 0,\, 1)$; BipolarSigmoid como $\tanh(x/2)$; e Elliott como $x/(1 + |x|)$ [Elliott 1993]. A redundância matemática entre Elliott e Softsign (nativa do PyTorch) foi mantida para refletir a origem distinta das implementações.

A ativação do *ensemble* é uma combinação convexa de n primitivas $\{f_0, \ldots, f_{n-1}\}$:

$$g(x) = \sum_{i=0}^{n-1} \mathrm{softmax}(w)_i \, f_i(x), \qquad w \in \mathbb{R}^n, \tag{1}$$

onde o vetor de pesos w é aprendido e compartilhado entre as camadas. Como cada coeficiente $\mathrm{softmax}(w)_i \in (0, 1); i \in \{0, \cdots, n-1\}$, garante que $g(x)$ herde a limitação e a monotonicidade das funções base $f_i; i \in \{0, \cdots, n-1\}$, desde que todas sejam não decrescentes (ou não crescentes).

Para avaliar a resiliência a dados ruidosos, cada primitiva limitada sofreu uma transformação afim para mapear seu *range* natural aos intervalos $[0;\, 0{,}5]$, $[0;\, 1]$ e $[0;\, 5]$, gerando as variantes das famílias C19 e C8 (mantendo-se também as formas inalteradas, como C19_or). Este *design* visa testar a contenção do ruído por limitação de magnitude, o impacto de funções estritamente crescentes (C8), e verificar se a mistura de funções limitadas e não limitadas (C41) mitiga o impacto da perturbação frente ao uso exclusivo de não limitadas (C22).

3.2. Conjuntos de Dados e Arquitetura-Base

Os experimentos foram conduzidos sobre o *benchmark* OpenML-CC18 (*OpenML Curated Classification suite 2018*) [Bischl et al. 2021], composto por 72 tarefas de classificação de domínios variados. A curadoria da suíte impõe critérios explícitos: entre 500 e 100.000 instâncias, menos de 5.000 atributos, razão entre classe minoritária e majoritária de no mínimo $0{,}05$, e exclusão de bases artificiais, de séries temporais e de problemas com vazamento de informação (*data leakage*) ou solucionáveis trivialmente por um único atributo. Os conjuntos cobrem tanto problemas binários quanto multiclasse, com atributos numéricos e categóricos em proporções variadas, o que permite medir a resiliência em uma ampla gama de distribuições tabulares e sustenta a aplicação de testes de significância com $n = 72$ observações pareadas por comparação.

Na arquitetura da rede, adotamos uma *Multi-Layer Perceptron* (MLP) simples de três blocos ocultos ($128 \to 128 \to 64$) seguidos de uma camada linear de saída, ao invés de arquiteturas mais sofisticadas como *Transformers* tabulares. Cada bloco oculto segue estritamente a ordem LINEAR $\to$ BATCHNORM1D $\to$ *ativação* $\to$ DROPOUT$(0{,}1)$, de modo que a ativação recebe entradas já normalizadas e o *dropout* é aplicado após a não linearidade; essa ordem é idêntica em todas as estratégias comparadas. Manter uma arquitetura fixa e relativamente simples reduz os fatores de confusão arquiteturais e permite atribuir as diferenças observadas majoritariamente à função de ativação. Não se trata de um isolamento estrito — *BatchNorm*, *Dropout* e o compartilhamento do vetor w entre camadas também interagem com o comportamento das ativações —, mas de manter esses fatores constantes ao longo de toda a comparação. A profundidade de três blocos busca equilibrar capacidade representacional e custo de treinamento; o controle do sobreajuste

nas tarefas menores do CC18 não decorre da profundidade em si, mas da combinação de *dropout*, normalização por lote e parada precoce descrita adiante.

O processo de otimização dos pesos foi conduzido por meio do algoritmo Adam, utilizando uma taxa de aprendizado inicial fixa de $\eta = 10^{-3}$ e mini-lotes (*mini-batches*) estruturados com tamanho de lote igual a 128 instâncias.

Adotou-se parada precoce (*early stopping*) sobre a acurácia de validação, com paciência de 25 épocas e um mínimo de 10 épocas antes de habilitar o gatilho, medida que neutraliza flutuações espúrias da inicialização dos pesos. O teto de 1.000 épocas nunca foi atingido na prática.

Monitorou-se a acurácia de validação em vez da perda porque, sob precisão mista FP16, conjuntos de validação pequenos podem produzir *underflow* ou *overflow* no cálculo da perda e gerar valores indefinidos (NaN), ao passo que a acurácia, sendo uma contagem discreta de acertos, permanece numericamente estável.

Os coeficientes da combinação linear, representados pelo vetor de pesos w, são compartilhados globalmente entre as três camadas ocultas. Essa estratégia faz com que w não favoreça apenas uma camada, mas se adeque a todas, o que facilita também a análise dos resultados.

Custo computacional. O sobrecusto do *ensemble* é dominado pela avaliação das n primitivas: uma ativação combinada custa $\mathcal{O}(n)$ avaliações elementares contra $\mathcal{O}(1)$ da ReLU, para a mesma quantidade de multiplicações matriciais. Como as ativações operam sobre tensores de dimensão 128, 128 e 64, muito menores que as matrizes de pesos, esse termo não domina o custo assintótico. Em parâmetros o acréscimo é desprezível: o compartilhamento de w adiciona apenas n escalares (8 para C8, 41 para C41), contra $3n$ com vetores independentes por camada. A ordenação esperada de custo, em treino e inferência, é portanto ReLU $<$ C8 $<$ C19 $<$ C22 $<$ C41, aproximadamente linear em n; sua quantificação empírica não foi conduzida e constitui limitação reconhecida.

Os experimentos foram executados em uma estação de trabalho com processador Intel Core i5-13450HX, 32GB de RAM e GPU NVIDIA GeForce RTX 4050 (6GB VRAM), utilizando o framework PyTorch com aceleração via CUDA.

3.3. Injeção de Ruído

Antes da injeção de ruído, todas as *features* numéricas são padronizadas (z-score) usando média μ_j e desvio-padrão σ_j estimados *exclusivamente* no conjunto de treino, e as variáveis categóricas são codificadas via *one-hot*. Após esse pré-processamento, o conjunto de teste (*hold-out* de 10%) é perturbado por uma das três funções abaixo, com intensidade $\lambda \in \{0;\ 0{,}1;\ 0{,}25;\ 0{,}5;\ 1{,}0\}$. Para uma amostra $\mathbf{x} = (x_1, \ldots, x_d)$, a versão perturbada $\mathbf{x}' = (x'_1, \ldots, x'_d)$ é definida coordenada a coordenada por:

$$\text{Gaussiano:} \quad x'_j = x_j + \lambda\, s_j\, \varepsilon_j, \qquad \varepsilon_j \sim \mathcal{N}(0,1) \tag{2}$$

$$\text{Uniforme:} \quad x'_j = x_j + \lambda\, s_j\, u_j, \qquad u_j \sim \mathcal{U}(-1,+1) \tag{3}$$

$$\text{Masking:} \quad x'_j = b_j\, x_j, \qquad b_j \sim \text{Bernoulli}(1-\lambda) \tag{4}$$

onde s_j é o desvio-padrão da j-ésima *feature* no conjunto de teste, recalculado para preservar a escala por atributo nos dois ruídos aditivos. Assim, $\lambda = 0{,}25$ significa "um quarto

de desvio-padrão por *feature*" nos ruídos aditivos e "25% das *features* substituídas" no *masking*, cujo valor de substituição é $c = 0$ — equivalente, em dados padronizados, à imputação pela média do treino (e não a `NaN`). Sementes fixas por *fold* e por λ garantem reprodutibilidade.

3.4. Métricas de Avaliação e Testes Estatísticos

A Figura 2 resume o *pipeline* experimental completo. Os experimentos utilizaram protocolo de validação em dois níveis sobre as 72 tarefas do OpenML-CC18 [Bischl et al. 2021]. Cada conjunto foi particionado via *hold-out* estratificado (10% teste, 90% desenvolvimento), aplicando-se sobre o conjunto de desenvolvimento uma *10-Fold Cross-Validation* estratificada. Em cada *fold*, o MLP foi ajustado em 81% dos dados e avaliado em 9% para o critério de parada precoce (*early stopping*).

Após a otimização em ambientes limpos, o conjunto de teste foi submetido a três tipos de ruído (Gaussiano, Uniforme, *Masking*) em cinco intensidades ($\lambda \in \{0; 0{,}1; 0{,}25; 0{,}5; 1{,}0\}$). A resiliência foi quantificada pelo *Robustness Gap* (Gap_R):

$$\text{Gap}_R(\lambda) = \text{Acurácia}_{\lambda=0} - \text{Acurácia}_{\lambda} \tag{5}$$

Adotou-se $\lambda = 0{,}5$ como nível de referência para análises agregadas.

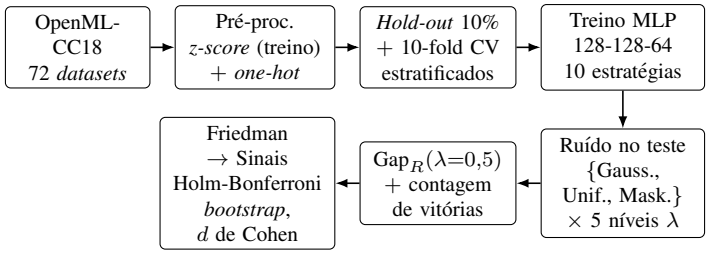


Figura 2. Visão geral do *pipeline* experimental. O ruído é aplicado *apenas* ao conjunto de teste, após o treinamento em dados limpos, de modo que a degradação medida reflete a sensibilidade do modelo já ajustado.

A significância estatística foi avaliada ao nível do conjunto de dados ($n = 72$), sendo que para cada um, foi calculado a média das acurácias dos 10 *folds* e essa média usada na análise estatística. Empregou-se o teste não-paramétrico de Friedman [Demšar 2006] para cada par (tipo, nível) de ruído, testando a equivalência das 10 estratégias. Rejeitada a hipótese nula, aplicaram-se testes *post-hoc* (Teste de Sinais) entre as variantes limitadas (C19, C8) e os *baselines* (C22, C41, Original), com correção de Holm-Bonferroni ($\alpha = 0{,}05$). Estimaram-se intervalos de confiança de 95% ($\text{CI}_{95\%}$) via *bootstrap* (2.000 repetições) e o d de Cohen para magnitudes de efeito.

Por fim, realizou-se uma contagem de vitórias. Formalmente, dado um conjunto de candidatos $\mathcal{P}$ (o *pool*), para cada par (conjunto de dados d, tipo de ruído t) declara-se vencedora a estratégia $s^\star(d,t) = \arg\min_{s\in\mathcal{P}} \text{Gap}_R(s,d,t;\lambda{=}0{,}5)$, isto é, a de menor degradação; quando $\mathcal{P}$ agrupa variantes sob um mesmo rótulo de família, esta é representada pela sua variante de menor *gap* naquele par. Empates são resolvidos pela ordem de listagem e cada configuração totaliza $72 \times 3 = 216$ pares. Como a contagem é, por construção, *dependente da composição de* $\mathcal{P}$ — remover um concorrente redistribui suas vitórias —, ela é usada aqui apenas como leitura descritiva; a inferência formal repousa

sobre as comparações pareadas da Subseção 4.4, que independem do *pool*. Reportamos quatro configurações de $\mathcal{P}$:

- *Pool* completo (10 estratégias);
- C19 vs. *baselines* (C8 excluído);
- C8 vs. *baselines* (C19 excluído);
- Melhor variante única de cada família limitada (C19, C8) vs. *baselines*.

Essa análise multinível avalia a robustez individual de cada família sem o viés de divisão de votos entre variantes similares.

4. Análise Experimental e Resultados

A apresentação segue a estrutura inferencial da Subseção 3.4: teste de Friedman, quantificação do *Robustness Gap* por família, contagem de vencedores em quatro configurações e, por fim, magnitudes de efeito via *bootstrap* e d de Cohen.

4.1. Teste Não-Paramétrico de Friedman

A Tabela 1 apresenta os resultados do teste não-paramétrico de Friedman aplicado a cada par ordenado (tipo, nível) de ruído, totalizando 15 configurações. A hipótese nula de equivalência distribucional das dez estratégias de ativação foi rejeitada em 14 das 15 combinações testadas, com $p < 10^{-4}$ na maior parte dos cenários. A única exceção observada foi a configuração (*masking*, $\lambda = 1{,}0$), em que o nível extremo de perturbação degrada uniformemente todas as estratégias e dissolve as diferenças observadas em níveis menores ($p = 0{,}169$). Este resultado confirma a presença de diferenças sistemáticas entre as famílias em praticamente todo o intervalo de operação considerado, autorizando a condução dos testes *post-hoc* subsequentes.

Tabela 1. Teste de Friedman por par (tipo, nível) de ruído, com $n = 72$ conjuntos de dados e $k = 10$ estratégias em todas as 15 configurações. A hipótese nula de equivalência é rejeitada em 14 das 15 ($\alpha = 0{,}05$); a única exceção, marcada com †, é (*masking*, $\lambda = 1{,}0$).

	Gaussiano		***Masking***		**Uniforme**	
λ	χ^2	valor-p	χ^2	valor-p	χ^2	valor-p
0,00	147,48	$2{,}9\times10^{-27}$	147,48	$2{,}9\times10^{-27}$	147,48	$2{,}9\times10^{-27}$
0,10	109,53	$1{,}8\times10^{-19}$	124,01	$2{,}0\times10^{-22}$	97,06	$6{,}2\times10^{-17}$
0,25	92,45	$5{,}2\times10^{-16}$	91,53	$8{,}0\times10^{-16}$	101,28	$8{,}7\times10^{-18}$
0,50	32,88	$1{,}4\times10^{-4}$	49,76	$1{,}2\times10^{-7}$	74,27	$2{,}2\times10^{-12}$
1,00	27,75	$1{,}1\times10^{-3}$	12,87	$0{,}169^{\dagger}$	28,50	$7{,}9\times10^{-4}$

4.2. Robustness Gap por Família

A Figura 3 apresenta o Gap_R médio por família para os três tipos de ruído. Em todas as configurações, as variantes com intervalo mínimo C8[0; 0,5] e C19[0; 0,5] ocupam as duas primeiras posições, indicando maior estabilidade representacional sob corrupção da entrada. Em contrapartida, o controle não-limitado C22 e a mistura C41 ocupam consistentemente as últimas posições do *ranking*, com gaps médios significativamente superiores aos das melhores variantes limitadas. A função ReLU (Original) ocupa posição intermediária em todos os ruídos.

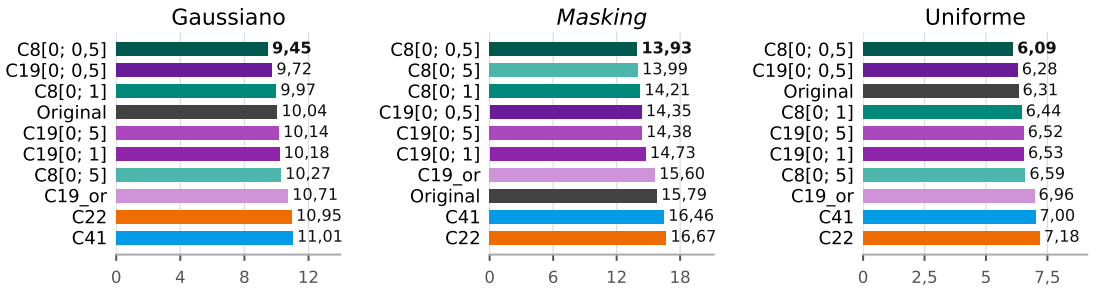


Figura 3. *Robustness Gap* médio por família, em pontos percentuais, ordenado do mais robusto (topo) ao menos robusto (base), para cada tipo de ruído sob $\lambda = 0{,}5$. Valores menores indicam maior robustez; em **negrito**, o menor valor de cada painel. Tons de uma mesma cor indicam variantes da mesma família.

A Tabela 2 consolida os valores exatos. A diferença entre a melhor variante limitada (C8[0; 0,5]) e o controle C22 varia de 1,1 (sob ruído Uniforme) a 2,7 pontos percentuais (sob *masking*), relevância estatística quantificada na Subseção 4.4; os desvios-padrão (não exibidos por concisão, de 14 a 16 pontos percentuais) mantêm-se aproximadamente constantes entre as famílias.

Tabela 2. *Robustness Gap* médio ($\overline{\text{Gap}_R}$), em pontos percentuais, por família e tipo de ruído, ao longo dos 72 conjuntos de dados do OpenML-CC18 com $\lambda = 0{,}5$. Valores menores indicam maior robustez. Em **negrito**, o menor valor de cada linha.

	Baselines			Família C19				Família C8		
Ruído	Orig.	C22	C41	_or	[0; 0,5]	[0; 1]	[0; 5]	[0; 0,5]	[0; 1]	[0; 5]
Gaussiano	10,04	10,95	11,01	10,71	9,72	10,18	10,14	**9,45**	9,97	10,27
Masking	15,79	16,67	16,46	15,60	14,35	14,73	14,38	**13,93**	14,21	13,99
Uniforme	6,31	7,18	7,00	6,96	6,28	6,53	6,52	**6,09**	6,44	6,59

4.3. Vencedores por Família e Análise de Vote-Splitting

Na configuração com todas as dez estratégias, a Original aparece como vencedora em 60 dos 216 cenários. Entretanto, conforme antecipado na Subseção 3.4, esta leitura é distorcida pelo *vote-splitting*: as variantes das famílias C19 e C8 ocupam regiões similares do espaço de robustez e dividem entre si as vitórias dos *datasets* onde uma família limitada seria a melhor escolha.

Para isolar a contribuição individual de cada família limitada, foram conduzidas três análises de *vote-splitting* progressivamente mais restritivas. A Tabela 3 apresenta a contagem de vitórias após a eliminação do *vote-splitting* entre as duas famílias limitadas. Em ambos os painéis, observa-se um padrão consistente: ao remover a família competidora, a variante com intervalo mínimo $[0;\ 0{,}5]$ emerge como a melhor representante limitada, com desempenho próximo ao da *baseline* ReLU no agregado e superior sob *masking*. Os controles não-limitado (C22) e misto (C41) permanecem no fim do *ranking* em ambos os painéis, reforçando a fragilidade desses *ensembles* sob perturbação.

A Tabela 4 mantém no *pool* apenas a melhor variante de cada família limitada junto aos três *baselines*, eliminando o *vote-splitting* entre famílias e dentro de cada

Tabela 3. Contagem de vitórias por família após eliminação do *vote-splitting* entre as famílias C19 e C8. Painel (a): família C8 removida do *pool* de candidatos, isolando o desempenho da família C19 contra os *baselines*. Painel (b): família C19 removida, isolando o desempenho da família C8. Em destaque, a variante limitada com maior contagem agregada em cada painel. Cada painel totaliza 216 cenários (72 *datasets* $\times$ 3 ruídos).

Família	**Total**	**Gauss.**	***Mask.***	**Unif.**
(a) Família C8 excluída — isolando C19				
Original	70	26	14	30
C19[0; 0,5]	53	18	19	16
C19[0; 5]	37	9	21	7
C19[0; 1]	24	12	5	7
C22	14	4	7	3
C19_or	11	1	4	6
C41	7	2	2	3
(b) Família C19 excluída — isolando C8				
Original	69	25	15	29
C8[0; 0,5]	63	24	19	20
C8[0; 5]	36	7	21	8
C8[0; 1]	23	7	10	6
C22	14	5	5	4
C41	11	4	2	5

família. O painel (b) revela um padrão expressivo: C8[0; 0,5], isolada de suas variantes irmãs e da família C19, supera a *baseline* ReLU no agregado dos três ruídos (99 vitórias contra 83 em 216 cenários), com domínio particularmente claro sob *masking* (41 contra 21 — quase o dobro). C19[0; 0,5] (painel (a)) fica próxima da ReLU no total (86 contra 92), mas também a supera sob *masking* (35 contra 21). Em ambos os painéis, C22 e C41 permanecem na parte inferior da contagem.

Como esta contagem é descritiva e sensível ao *pool*, a conclusão inferencial repousa sobre as comparações *pareadas* da Subseção 4.4, que independem de candidatos e confirmam o mesmo ordenamento — sustentando a leitura de que o *vote-splitting*, e não a ausência de efeito, explicava a aparente liderança da ReLU entre as dez estratégias.

4.4. Magnitudes de Efeito e Intervalos de Confiança

A Tabela 5 revela três achados centrais. Primeiro, sob *masking*, todas as seis variantes limitadas com intervalo explícito diferem significativamente de C22, com d de Cohen entre 0,51 e 0,58 (efeito médio-grande) e $p < 0{,}001$ após correção de Holm-Bonferroni. Segundo, sob ruído Gaussiano e Uniforme a significância restringe-se a um subconjunto pequeno (C19[0; 0,5], C19[0; 1] e C8[0; 1] sob Gaussiano; as duas primeiras sob Uniforme), com d entre 0,31 e 0,39 (efeito pequeno-médio); é notável que C19_or, sem reescala explícita do contradomínio, não atinge significância em nenhum ruído, indicando que a limitação efetiva exige um intervalo numérico controlado — não basta a primitiva ser teoricamente limitada. Terceiro, e particularmente relevante para a propriedade carregadora do efeito, a mistura C41 não difere significativamente de C22 em nenhum dos

Tabela 4. Contagem de vitórias considerando apenas uma variante limitada como representante da família, em disputa com os três *baselines* (Original, C22, C41). Esta configuração elimina simultaneamente o *vote-splitting* entre famílias (C19 vs. C8) e dentro de cada família (entre seus diferentes intervalos). Em destaque, a variante limitada; em **negrito**, os valores em que ela supera a *baseline* ReLU. Cada painel totaliza 216 cenários (72 *datasets* $\times$ 3 ruídos).

Família	**Total**	**Gauss.**	***Mask.***	**Unif.**
(a) Representante limitada: C19[0; 0,5]				
Original	92	32	21	39
C19[0; 0,5]	86	28	**35**	23
C22	19	6	9	4
C41	19	6	7	6
(b) Representante limitada: C8[0; 0,5]				
C8[0; 0,5]	**99**	**31**	**41**	27
Original	83	30	21	32
C22	19	6	6	7
C41	15	5	4	6

três ruídos (e exibe $d = -0{,}04$ sob Gaussiano). Este resultado é evidência empírica de que combinar funções limitadas e não-limitadas em um mesmo *ensemble* é insuficiente para colher o benefício de robustez: a propriedade decisiva é a limitação uniforme do contradomínio em todo o conjunto de primitivas.

Tabela 5. Tamanho do efeito (d de Cohen) das comparações pareadas entre cada variante limitada e o controle não-limitado C22, por tipo de ruído. Valores positivos indicam maior robustez da variante limitada. Significância após correção de Holm-Bonferroni: $^{*}p < 0{,}05$; $^{**}p < 0{,}01$; $^{***}p < 0{,}001$.

Comparação *vs.* C22	**Gaussiano**	***Masking***	**Uniforme**
C19_or	0,14	0,35	0,17
C19[0; 0,5]	$0{,}35^{*}$	$0{,}58^{***}$	$0{,}33^{*}$
C19[0; 1]	$0{,}37^{*}$	$0{,}56^{***}$	$0{,}31^{*}$
C19[0; 5]	0,33	$0{,}54^{***}$	0,29
C8[0; 0,5]	0,38	$0{,}51^{***}$	0,35
C8[0; 1]	$0{,}39^{**}$	$0{,}51^{***}$	0,29
C8[0; 5]	0,27	$0{,}53^{***}$	0,24
C41 (mistura)	$-0{,}04$	0,12	0,13

Consolidando, observa-se ainda que intervalos mais estreitos favorecem sistematicamente a robustez: nas três análises de *ranking* de *gap*, as variantes com intervalo $[0;\ 0{,}5]$ ocupam as duas primeiras posições, tanto na família C19 quanto na C8.

5. Conclusões e Trabalhos Futuros

Este trabalho demonstrou que, em combinações lineares treináveis de funções de ativação aplicadas a redes MLP sobre 72 tarefas tabulares do OpenML-CC18, *ensembles* puramente limitados são significativamente mais robustos a ruído de entrada que *ensembles* não-limitados (C22) ou mistos (C41), com efeito médio-grande sob *masking* ($d \approx 0{,}5$ após correção de Holm-Bonferroni). O subconjunto estritamente crescente C8, contendo apenas 8 primitivas, alcança desempenho equivalente ao C19 de 19 primitivas, evidenciando que a propriedade carregadora do efeito é a limitação uniforme do contradomínio, não a quantidade ou regularidade adicional das primitivas. A variante C8[0; 0,5] desponta como a mais robusta dentre todas as testadas. Trata-se, contudo, de uma recomendação condicionada ao escopo aqui avaliado — MLP de três blocos, dados tabulares do CC18 e perturbações aleatórias não-estruturadas nos três tipos e cinco intensidades descritos — e não de uma prescrição geral.

Limitações. Os achados foram estabelecidos sobre uma única família arquitetural e sobre dados tabulares. Não é possível afirmar, a partir destes experimentos, que a vantagem das combinações limitadas se preserva em arquiteturas com mecanismos de normalização ou de atenção distintos, nem sob perturbações estruturadas. A escolha por uma arquitetura fixa privilegiou a validade interna da comparação entre ativações, ao custo da validade externa entre arquiteturas. Tampouco foram medidos empiricamente os tempos de treino e inferência por estratégia.

Como trabalhos futuros, planeja-se: (i) ampliar o escopo arquitetural, contrastando a MLP com *Transformers* tabulares e com modelos convolucionais em tarefas de imagem, onde as ativações operam sobre representações intermediárias estruturalmente distintas, e incorporando a medição de tempo de parede e consumo de memória por estratégia; (ii) investigar a relação entre limitação do contradomínio e **robustez adversarial**, considerando perturbações estruturadas além do ruído aleatório aqui estudado; (iii) avaliar o impacto das primitivas limitadas sobre a **esparsidade e a interpretabilidade** dos pesos aprendidos, em particular das que não anulam valores negativos, contrastando-as com o comportamento esparsificante da ReLU; e (iv) conduzir **análise teórica** das condições sob as quais *ensembles* puramente limitados oferecem garantias formais de estabilidade representacional.

Agradecimentos

Natália Azeredo Viana da Silva agradece à Universidade Federal do Rio de Janeiro pela bolsa de iniciação científica concedida no âmbito do Programa Institucional de Bolsas de Iniciação Científica (PIBIC/UFRJ). Os autores utilizaram ferramentas de IA generativa Claude (Anthropic) como apoio à refatoração do código experimental e à revisão da escrita, e assumem integral responsabilidade por todo o conteúdo do artigo.

Referências

Apicella, A., Donnarumma, F., Isgrò, F., and Prevete, R. (2021). A survey on modern trainable activation functions. *Neural Networks*, 138:14–32.

Bhattacharya, A., Gole, S. D., Chaudhuri, S., Frery, A. C., and Banerjee, B. (2026). Arcgate: Adaptive arctangent gated activation. *arXiv preprint arXiv:2605.14518*.

Bischl, B., Casalicchio, G., Feurer, M., Gijsbers, P., Hutter, F., Lang, M., Mantovani, R. G., van Rijn, J. N., and Vanschoren, J. (2021). Openml benchmarking suites. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*.

de Moraes, R. F. and Evangelista, R. d. S. (2023). Heuristics to reduce linear combinations of activation functions to improve image classification. In *2023 36th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 169–174. IEEE.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30.

Dubey, S. R., Singh, S. K., and Chaudhuri, B. B. (2022). Activation functions in deep learning: A comprehensive survey and benchmark. *Neurocomputing*, 503:92–108.

Elliott, D. L. (1993). A better activation function for artificial neural networks. Technical Report ISR T.R. 93-8, Institute for Systems Research, University of Maryland.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Li, Z., Liu, F., Yang, W., Peng, S., and Zhou, J. (2022). A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects. *TNNLS*, 33:6999–7019.

Liew, S. S., Khalil-Hani, M., and Bakhteri, R. (2016). Bounded activation functions for enhanced training stability of deep neural networks on visual pattern recognition problems. *Neurocomputing*, 216:718–734.

Mishra, A., Chandra, P., Ghose, U., and Sodhi, S. S. (2017). Bi-modal derivative adaptive activation function sigmoidal feedforward artificial neural networks. *Applied Soft Computing*, 61:983–994.

Molina, A., Schramowski, P., and Kersting, K. (2020). Padé activation units: End-to-end learning of flexible activation functions in deep networks. In *International Conference on Learning Representations (ICLR)*.

Pauli, P., Gramlich, D., Berberich, J., Allgöwer, F., and Scherer, C. W. (2021). Training robust neural networks using lipschitz bounds. *IEEE Control Systems Letters*, 6:121–126.

Ru, Q., Hoang, L.-H., Hanif, M. A., and Shafique, M. (2020). Ft-clipact: Resilience analysis of deep neural networks and improving their fault tolerance using clipped activation. In *2020 IEEE 31st International Symposium on Software Reliability Engineering (ISSRE)*, pages 214–224. IEEE.

Schorn, C., Geissler, F., Qutub, S., and Dhamasia, A. (2021). Towards a safety case for hardware fault tolerance in convolutional neural networks using activation range supervision. In *International Conference on Computer Safety, Reliability, and Security (SAFECOMP)*. Springer.

Sütfeld, L. R., Brieger, F., Finger, H., Füllhase, S., and Pipa, G. (2018). Adaptive blending units: Trainable activation functions for deep neural networks. In *Proceedings of the SAI Intelligent Systems Conference (IntelliSys)*. arXiv:1806.10064.

Xu, B., Huang, R., and Li, M. (2016). Revise saturated activation functions. *arXiv preprint arXiv:1602.05980*.