

Uso de Particle Swarm Optimization na Otimização de Hiperparâmetros de Redes Neurais para Classificação Binária da Doença de Parkinson por Análise de Voz

Anderson Coelho Viana*, Caetano Reis Araújo Cardoso*, Maria Luiza Cavalcante Amaral Cernadas*

*Federal University of Pará, Belém, PA, Brazil

Emails: andviana@gmail.com, caetano.cardoso@itec.ufpa.br, maria.cernadas@itec.ufpa.br

Resumo—Esta pesquisa apresenta a implementação da otimização de hiperparâmetros de uma rede neural feedforward aplicada à classificação binária da doença de Parkinson (DP) através de características vocais utilizando o algoritmo *Particle Swarm Optimization* (PSO). o otimizador foi configurado para explorar o espaço de busca delimitado para os hiperparâmetros da arquitetura de uma rede neural Multilayer Perceptron (MLP), incluindo número de camadas (1–4), neurônios por camada (8 a 128) e taxa de aprendizado (10^{-5} a 10^{-1}). foram realizados 32 experimentos independentes cada um com 20 partículas e 20 iterações, totalizando 12.800 avaliações de modelos. O conjunto de dados compreende 22 atributos acústicos extraídos de gravações de fonação sustentada de 195 indivíduos (147 com Parkinson, 48 controles). A configuração ótima alcançou F1-score de 0.8669 ± 0.0207 sob validação cruzada estratificada $k = 5$, com acurácia de 0.7692 ± 0.0397 e AUC-ROC de 0.5941 ± 0.1287 . Todo o histórico das partículas foi armazenado em uma base de dados SQLite, permitindo análise detalhada de convergência. Os resultados evidenciam que o PSO é eficaz para ajustar hiperparâmetros de redes neurais em tarefas de classificação de DP, oferecendo uma alternativa sistemática ao ajuste manual e revelando o impacto de diferentes configurações arquiteturais no desempenho do modelo.

Index Terms—Doença de Parkinson, Classificação Vocal, Otimização de Hiperparâmetros, Particle Swarm Optimization, Redes Neurais, Classificação Binária.

I. INTRODUÇÃO

A Doença de Parkinson (DP) é uma enfermidade neurodegenerativa progressiva que afeta cerca de 10 milhões de pessoas em todo o mundo [1]. Caracteriza-se pela degeneração dos neurônios dopaminérgicos na substância negra, resultando em sintomas motores clássicos como bradicinesia, tremor de repouso e rigidez, e em manifestações não motoras, como disfunção autonômica, que comprometem significativamente a qualidade de vida [2]. Alterações vocais podem surgir até cinco anos antes dos sinais motores, tornando a análise de voz uma ferramenta promissora para detecção não invasiva e monitoramento da progressão da DP.

Entre as alterações vocais associadas à DP destacam-se a disartria hipocinética, redução da intensidade, *breathiness*, *roughness* e tremor vocal exacerbado [3]. Tais alterações podem ser quantificadas por meio de medidas acústicas objetivas, como *jitter* (perturbação de frequência), *shimmer* (perturbação

de amplitude), razão harmônico-ruído (HNR) e índices dinâmicos não lineares (p.ex., RPDE, DFA) [3] [4].

Modelos de aprendizado de máquina vêm atingindo acurácias superiores a 90% na classificação vocal de DP; contudo, seu desempenho depende fortemente da escolha apropriada de hiperparâmetros, tradicionalmente ajustados por busca manual ou *grid search*, estratégias dispendiosas e, muitas vezes, subótimas [3].

Algoritmos evolutivos, em especial o *Particle Swarm Optimization* (PSO), oferecem uma alternativa eficiente para a sintonização de hiperparâmetros. Inspirado no comportamento coletivo de enxames, o PSO faz cada partícula (solução candidata) atualizar sua posição com base em suas melhores experiências individual e global, equilibrando exploração e refinamento em espaços de alta dimensionalidade.

Este trabalho apresenta um sistema modular em *Python* que integra pré-processamento de dados, construção de arquiteturas *Multilayer Perceptron* (MLP), otimização automática de hiperparâmetros via PSO, avaliação de desempenho e análise estatística. A plataforma fornece um fluxo reprodutível e completo para pesquisa em classificação vocal automatizada da DP.

II. MÉTODOS

A. Desenho do Estudo

Foi conduzido um estudo experimental computacional cujo objetivo foi desenvolver e avaliar um sistema de otimização automática de hiperparâmetros baseado em *Particle Swarm Optimization* (PSO) para a classificação binária de DP. A solução foi estruturada de forma modular, com componentes independentes para cada etapa do *pipeline* de processamento:

- Carregamento de dados (`data_utils.py`)
- Construção dos modelos MLP (`model_utils.py`)
- Otimização via PSO (`ps_optimizer.py`)
- Gerenciamento do banco de dados (`database_utils.py`)
- Avaliação do modelo final (`evaluate_final_model.py`)

Todos os parâmetros experimentais foram centralizados no arquivo `config.py`, garantindo reprodutibilidade e fácil

Tabela I: Características Acústicas do *Dataset*

Categoria	Variável	Descrição
Frequência	MDVP:Fo, Fhi, Flo	Freq. média, máx. e mín. (Hz)
Jitter	Jitter(%), RAP, PPQ, DDP	Perturbação de frequência entre ciclos
Shimmer	Shimmer(dB), APQ3, APQ5, DDA	Perturbação de amplitude vocal
Ruído	NHR, HNR	Relação harmônico-ruído
Dinâmicas	RPDE, DFA, D2, PPE, spread1/2	Medidas não-lineares de complexidade temporal

parametrização. O projeto completo encontra-se disponível em: https://andviana.github.io/ps0_binary_rnn_pd_audio/.

B. Coleta e Pré-processamento de Dados

O dataset utilizado compreende 1.195 registros vocais, dos quais 195 correspondem a gravações reais de fonação sustentada (*phon_**) e 1.000 amostras sintéticas (*synth_**). Neste estudo optou-se por utilizar, apenas os dados reais, resultando em 195 amostras: 147 de pacientes com DP (75,4%) e 48 de controles saudáveis (24,6%). O conjunto de dados é público e pode ser obtido no Kaggle através da URL: <https://www.kaggle.com/code/alirafazi/neural-network-on-parkinsons-disease>

Cada registro vocal foi caracterizado por 22 características acústicas agrupados em cinco categorias conforme tabela I.

- **Frequência Fundamental:** medidas de frequência vocal média, máxima e mínima.
- **Jitter (Variação de Frequência):** quantificam instabilidades na frequência fundamental, indicativas de controle vocal irregular.
- **Shimmer (Variação de Amplitude):** medem variações na amplitude vocal, relacionadas à estabilidade glótica.
- **Medidas de Ruído:** avaliam a qualidade harmônica da voz.
- **Dinâmica Não-Linear:** capturam complexidade dinâmica e irregularidades não-lineares no sinal vocal.

C. Normalização e Estratificação

No pré-processamento, todas as 22 variáveis acústicas foram normalizadas para o intervalo $[0, 1]$ por meio do `MinMaxScaler` da biblioteca *scikit-learn*. O *scaler* foi ajustado (*fit*) exclusivamente sobre o conjunto de treino a fim de prevenir contaminação por informações da validação, e em seguida aplicado (*transform*) aos dados de validação utilizando os mesmos parâmetros de mínima e máxima.

Para a etapa de particionamento foi adotado um esquema **estratificado 80 %-20 %** (`train_test_split, random_state=42`), preservando a proporção original entre pacientes com DP e controles saudáveis. Essa estratégia garante que ambos os subconjuntos reflitam a distribuição de classes do *dataset*, reduzindo viés e assegurando maior robustez na avaliação do modelo.

D. Implementação do PSO

Particle Swarm Optimization (PSO) é um algoritmo de otimização populacional inspirado no comportamento coletivo observado em grupos naturais, como bandos de pássaros ou cardumes de peixes [5]. No PSO, cada possível solução do problema é representada por uma partícula que se desloca no espaço de busca, ajustando sua posição com base tanto na sua melhor experiência individual quanto na melhor solução encontrada pelo grupo. Essa abordagem promove o equilíbrio entre a exploração global do espaço de soluções e o refinamento local das melhores regiões encontradas, tornando o PSO especialmente eficaz em problemas complexos e não lineares. Devido à sua simplicidade de implementação e eficiência computacional, o PSO tem sido amplamente utilizado em diversas áreas, incluindo otimização de hiperparâmetros em modelos de aprendizado de máquina.

Arquitetura do Sistema PSO:

O sistema PSO foi implementado na classe `PSOoptimizer`, operando sobre vetores de partícula com 5 dimensões:

$$\mathbf{x} = [n_{\text{layers}}, n_1, n_2, n_3, \text{learning_rate}]$$

Onde:

- $n_{\text{layers}} \in \mathbb{N}$: número de camadas ocultas.
- $n_1, n_2, n_3 \in \mathbb{N}$: neurônios por camada.
- $\text{learning_rate} \in [10^{-5}, 10^{-1}]$: taxa de aprendizado Adam.

Dinâmica das Partículas:

A atualização de velocidade seguiu a formulação clássica do PSO:

$$v_i^{t+1} = w \cdot v_i^t + c_1 \cdot r_1 \cdot (p_i - x_i^t) + c_2 \cdot r_2 \cdot (g - x_i^t) \quad (1)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (2)$$

Com parâmetros:

- $w = 0,7$ (inércia)
- $c_1 = c_2 = 1,5$ (coeficientes cognitivo/social)
- $r_1, r_2 \sim \mathcal{U}(0, 1)$ (números aleatórios)

População Inicial:

A população inicial foi definida através de distribuição uniforme cobrindo o espaço de busca completo. A população de 20 partículas incluiu configurações arquiteturais diversificadas, desde redes simples (1 camada, 8 neurônios) até complexas (4 camadas, 128 neurônios).

E. Implementação da Rede Neural

A rede neural *Multilayer Perceptron* (MLP) é um modelo de aprendizado supervisionado composto por camadas de neurônios artificiais dispostas de forma sequencial: uma camada de entrada, uma ou mais camadas ocultas totalmente conectadas e uma camada de saída [4]. Cada neurônio executa uma combinação linear ponderada das entradas recebidas e aplica uma função de ativação não linear, conferindo à MLP a capacidade de aproximar funções complexas. Durante o treinamento, os pesos das conexões são ajustados pelo algoritmo de

retropropagação do erro (*backpropagation*), normalmente em conjunto com otimizadores baseados em gradiente, de modo a minimizar uma função de perda entre as saídas previstas e os valores reais. Essa arquitetura torna a MLP particularmente versátil para tarefas de classificação e regressão, sendo capaz de modelar relações não lineares entre variáveis de entrada e saída.

Arquitetura MLP:

As redes neurais implementadas seguiram arquitetura feed-forward densamente conectada (MLP):

```
Input (22) → Dense(n_1) → BatchNorm → Dropout(0.3) →
→ Dense(n_2) → BatchNorm → Dropout(0.3) →
→ ... →
→ Dense(1, sigmoid)
```

Componentes arquiteturais:

- **Camadas de entrada:** 22 neurônios (características acústicas).
- **Camadas ocultas:** 1–4 camadas Dense com ativação ReLU.
- **Regularização:** BatchNormalization após cada camada oculta para estabilização de gradientes.
- **Dropout:** Taxa de 0.3 para prevenção de overfitting.
- **Camada de saída:** 1 neurônio com ativação sigmoid para classificação binária.

Configuração de Treinamento:

- **Otimizador:** Adam com learning rate otimizado pelo PSO.
- **Função de perda:** Binary Cross-Entropy.
- **Métricas:** Accuracy durante treinamento.
- **Batch size:** 32.
- **Épocas máximas:** 30.
- **Early Stopping:** 5 épocas monitorando validation loss.
- **Validation split:** 20% dos dados de treino.

F. Parametrização do PSO

A configuração utilizada para o PSO foi implementado conforme a tabela II:

Tabela II: Configurações do PSO

Parâmetro	Valor/Descrição
particulas	20
iterações	20
experimentos	32
Opções	
c1	1.5
c2	1.5
w	0.7
Limites	
lower	[1, 8, 8, 8, 1e-5]
upper	[4, 128, 128, 128, 1e-1]

Tanto para a atualização da velocidade, quanto para a posição das partículas, foi adicionada a verificação e tratamento utilizando o método "clip" da biblioteca numpy, garantindo que ambos permanecessem dentro dos limites estabelecidos.

Total de avaliações:

No total, ao final da execução do PSO para otimização dos hiperparâmetros da rede neural, foram executados $32 \times 20 \times 20 = 12.800$ treinamentos de rede neural.

Função de Fitness:

O F1-score foi escolhido como métrica principal devido à sua robustez em conjuntos de dados desbalanceados, pois balanceia precisão e recall:

$$F1 = 2 \times \frac{\text{Precisão} \times \text{Recall}}{\text{Precisão} + \text{Recall}} \quad (3)$$

A função objetivo minimizou:

$$f(\mathbf{x}) = 1 - \text{F1-score} \quad (4)$$

Decodificação de Partículas:

O vetor contínuo da partícula foi mapeado para hiperparâmetros discretos/contínuos conforme abaixo:

$$\begin{aligned} n_{\text{layers}} &= \max(1, \min(4, \text{round}(x))) \\ \text{neurons}[i] &= \max(8, \min(128, \text{round}(x[i + 1]))) \\ \text{learning_rate} &= \max(10^{-5}, \min(10^{-1}, x)) \end{aligned}$$

G. Análise Estatística

Armazenamento e Monitoramento:

Para viabilizar a análise estatística e reprodutível dos experimentos, todos os dados gerados pelo PSO foram persistidos em um banco SQLite com duas tabelas principais:

- **pso_resultados:** dados individuais de partículas por iteração (posição, velocidade, pbest, fitness).
- **pso_execucao:** métricas agregadas por experimento (tempo, recursos, convergência).

Tabela III: Dados da Execução (pso_resultados)

Campo	Descrição
num_experimento	Número identificador do experimento
tempo_total_seg	Tempo total de execução (segundos)
tempo_medio_iteracao	Tempo médio por iteração (segundos)
tempo_medio_treino_particula	Tempo médio de treinamento por partícula (segundos)
uso_medio_cpu	Uso médio da CPU durante o experimento (%)
uso_max_memoria_mb	Uso máximo de memória RAM (MB)
uso_disco_mb	Uso de espaço em disco (MB)
total_iteracoes	Número total de iterações realizadas (20 para todos os experimentos)

A tabela III armazena dados relacionados aos recursos computacionais utilizados e desempenho da execução do otimizador, agregando as informações sobre todos os experimentos realizados.

A tabela IV armazena os dados completos de todas as partículas para cada iteração em cada experimento realizado. Essa

Tabela IV: Dados das Partículas (pso_resultados)

Campo	Descrição
num_experimento, num_iteracao, num_particula	Identificadores exclusivos de cada partícula.
pos_camada,	
pos_n1,	
pos_n2, pos_n3,	
pos_lr	Posições atuais (valores dos hiperparâmetros).
vel_camada,	
vel_n1,	
vel_n2,	
vel_n3,	Velocidades associadas às dimensões da partícula.
vel_lr	
pbest_camada,	
pbest_n1,	
pbest_n2,	Melhores posições pessoais (<i>pbest</i>).
pbest_n3,	
pbest_lr	
num_camadas	
f1_score	Número efetivo de camadas após arredondamento.
	Valor do F1-score (função-objetivo) obtido para a partícula.
peso	Complemento do F1-score ($1 - f1_score$), usado como <i>oaptido</i> .
int_best	Indicador binário da melhor partícula na iteração.

estrutura viabiliza consultas SQL e exportações diretas para ferramentas de análise em Python como o pandas, assegurando rastreabilidade completa dos experimentos.

H. Avaliação Final

Após a otimização com PSO, o melhor modelo identificado dentre os experimentos foi submetido à validação cruzada estratificada com $k = 5$ folds, visando uma avaliação robusta e uma estimativa não-enviesada de desempenho. Nesse procedimento, o conjunto de dados é particionado em cinco subconjuntos (*folds*) de tamanhos aproximadamente iguais, preservando em cada fold a mesma proporção de classes observada no conjunto original. Em cada rodada, quatro folds são combinados para o treinamento e o fold restante é reservado para a avaliação. Esse processo é repetido cinco vezes, de modo que cada amostra apareça exatamente uma vez no conjunto de teste. Ao final, as métricas (F1-score, acurácia, precisão, recall) são calculadas em cada rodada e depois agregadas, fornecendo uma estimativa mais confiável do desempenho generalizado do modelo, ao mesmo tempo em que reduz a variância causada por diferentes divisões de dados.

As métricas reportadas pela avaliação foram:

- **F1-score:** média \pm desvio padrão.
- **Acurácia:** proporção de classificações corretas.
- **AUC-ROC:** área sob a curva ROC.
- **Matriz de confusão:** distribuição das predições.
- **Precisão, Recall, Especificidade:** métricas detalhadas.

Análise de Convergência: A convergência do PSO foi analisada através de:

- Evolução do melhor fitness ao longo das iterações.
- Diversidade populacional (variância das posições).
- Estagnação do *global best*.

III. RESULTADOS

A. Características da Amostra

A amostra analisada neste estudo, após o pré-processamento e devidas exclusões, compreendeu 195 gravações reais de fonação sustentada, coletadas de 195 indivíduos distintos. Desse total, 147 registros correspondem a pacientes diagnosticados com Doença de Parkinson (DP), representando 75,4 % da amostra, enquanto 48 registros pertencem ao grupo controle saudável (24,6 %). As 22 características acústicas apresentaram distribuições distintas entre grupos, confirmando potencial discriminativo. Medidas de jitter (especificamente MDVP:Jitter(%)) variaram de 0.168% a 3.316%, com valores médios elevados em pacientes com DP ($0.699\% \pm 0.524\%$) comparados a controles ($0.387\% \pm 0.205\%$). Similarmente, shimmer (MDVP:Shimmer) apresentou variação de 0.455% a 6.273%, indicando maior instabilidade vocal em DP (média $3.366\% \pm 1.997\%$ para DP versus $1.762\% \pm 0.554\%$ para controles). Medidas não-lineares (RPDE, DFA, PPE) mostraram complexidade dinâmica aumentada em registros patológicos. Especificamente, RPDE foi de 0.51682 ± 0.10125 para DP e 0.44255 ± 0.09220 para controles; DFA foi de 0.72541 ± 0.05479 para DP e 0.69572 ± 0.05135 para controles; e PPE foi de 0.23383 ± 0.08427 para DP e 0.12302 ± 0.04482 para controles, conforme dados da tabela V.

B. Performance e Convergência do PSO

O sistema PSO executou 32 experimentos independentes, com 20 partículas, em 20 iterações por experimento, totalizando em 12.800 treinamentos da rede neural e avaliações de modelo. A convergência média ocorreu a partir da iteração 10, onde, de acordo com o gráfico da figura 7 houve a formação de grupos distintos, indicando a convergência das partículas para regiões com bom desempenho observa-se uma tendência nítida de convergência à medida que o número iterações aumenta. Isso demonstra que o algoritmo PSO está, de fato, funcionando como esperado, explorando o espaço de busca e ajustando seus parâmetros para encontrar soluções promissoras. O gráfico da figura 6 mostra uma queda mais acentuada nas iterações iniciais, seguida por uma estabilização gradual. Isso é um comportamento típico de algoritmos de otimização baseados em enxame, como o PSO, onde a fase de exploração inicial é mais vigorosa, e à medida que as partículas se aproximam de regiões promissoras do espaço de busca, a fase de exploração se torna mais refinada, resultando em menores ganhos de melhoria a cada iteração.

1) *Análise PCA do Espaço de Busca e Distribuição de Desempenho no PSO:* O gráfico da figura 1 apresenta a projeção bidimensional do espaço de busca explorado pelo algoritmo Particle Swarm Optimization (PSO), obtida através da Análise de Componentes Principais (PCA). Cada ponto representa uma partícula avaliada durante os experimentos, posicionada segundo suas duas primeiras componentes principais, enquanto as cores indicam o valor do F1-score alcançado. Observa-se uma ampla dispersão das partículas, sugerindo uma boa cobertura do espaço de busca, com regiões concentradas em

Tabela V: Estatísticas Descritivas dos Atributos Vocais por Grupo

Característica	Grupo	Valores (Média \pm Desvio Padrão)
MDVP:Fo(Hz)	Parkinson	145.18076 \pm 32.34805
	Controle	181.93777 \pm 52.73107
MDVP:Fhi(Hz)	Parkinson	188.44146 \pm 88.33918
	Controle	223.63675 \pm 96.72707
MDVP:Flo(Hz)	Parkinson	106.89356 \pm 32.27436
	Controle	145.20729 \pm 58.75707
MDVP:Jitter(%)	Parkinson	0.00699 \pm 0.00524
	Controle	0.00387 \pm 0.00205
MDVP:Jitter(Abs)	Parkinson	0.98643 \pm 3.97592
	Controle	0.00002 \pm 0.00001
MDVP:RAP	Parkinson	0.00376 \pm 0.00324
	Controle	0.00193 \pm 0.00107
MDVP:PPQ	Parkinson	0.00390 \pm 0.00300
	Controle	0.00206 \pm 0.00094
Jitter:DDP	Parkinson	0.01127 \pm 0.00972
	Controle	0.00578 \pm 0.00320
MDVP:Shimmer	Parkinson	0.03366 \pm 0.01997
	Controle	0.01762 \pm 0.00554
MDVP:Shimmer(dB)	Parkinson	0.32120 \pm 0.20780
	Controle	0.16296 \pm 0.05782
Shimmer:APQ3	Parkinson	0.01768 \pm 0.01080
	Controle	0.00950 \pm 0.00346
Shimmer:APQ5	Parkinson	0.02028 \pm 0.01285
	Controle	0.01149 \pm 0.00323
MDVP:APQ	Parkinson	0.02760 \pm 0.01806
	Controle	0.01330 \pm 0.00383
Shimmer:DDA	Parkinson	0.05303 \pm 0.03239
	Controle	0.02851 \pm 0.01037
NHR	Parkinson	0.02921 \pm 0.04445
	Controle	0.01148 \pm 0.01909
HNR	Parkinson	20.97405 \pm 4.33914
	Controle	24.67875 \pm 3.43454
RPDE	Parkinson	0.51682 \pm 0.10125
	Controle	0.44255 \pm 0.09220
DFA	Parkinson	0.72541 \pm 0.05479
	Controle	0.69572 \pm 0.05135
spread1	Parkinson	-4767774.25170 \pm 1801759.42183
	Controle	-5810351.68750 \pm 2292267.65245
spread2	Parkinson	0.24813 \pm 0.07781
	Controle	0.16029 \pm 0.06298
D2	Parkinson	2282935.38776 \pm 687231.95769
	Controle	1972233.60417 \pm 616133.82598
PPE	Parkinson	0.23383 \pm 0.08427
	Controle	0.12302 \pm 0.04482

tons mais claros, indicando áreas associadas a desempenhos elevados. Essa visualização facilita a identificação de regiões promissoras no espaço de soluções e fornece evidências sobre o comportamento exploratório e a capacidade do PSO de convergir para zonas de maior desempenho.



Figura 1: Análise PCA do Espaço de Busca e Distribuição de Desempenho no PSO.

2) *Análise da variância explicada por componente principal do PCA:* Os gráficos da figura 2 representam a eficácia da redução de dimensionalidade pelo PCA. Na proporção da Variância Explicada Individual (barras), Cada barra representa a porcentagem da variância total dos dados da otimização que é capturada por uma única componente principal. As componentes são ordenadas de forma decrescente pela quantidade de variância que explicam. No gráfico de variância cumulativa explicada (linha), a linha azul mostra a soma acumulada da variância explicada pelas componentes principais. Essa linha é crucial para determinar quantas componentes são necessárias para reter uma quantidade aceitável de informação dos dados originais. As duas primeiras componentes já explicam 93% da variância, e a terceira componente eleva a variância cumulativa para além de 95%. Isso sugere que, para a maioria das análises, as duas ou três primeiras componentes principais seriam suficientes para representar a maior parte da informação do espaço de busca original de 5 dimensões.

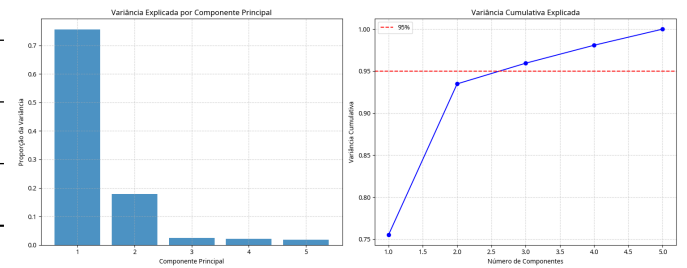


Figura 2: Variância explicada por componente principal e cumulativa.

3) *Análise dos Histogramas do Espaço de Busca do Algoritmo PSO*: Os gráficos da figura 3 representam a distribuição dos valores explorados pelo algoritmo de Otimização por Enxame de Partículas (PSO) em cada uma das cinco dimensões do seu espaço de busca. Se as barras estiverem concentradas em certas regiões, isso sugere que o algoritmo convergiu ou passou mais tempo explorando essas áreas. Picos nos histogramas indicam regiões do espaço de busca onde o algoritmo encontrou soluções promissoras ou onde as partículas se agruparam, sendo uma informação útil para entender quais configurações de parâmetros são mais eficazes. Os limites dos eixos X em cada histograma representam o intervalo de valores permitidos para cada dimensão, conforme definido no problema de otimização.

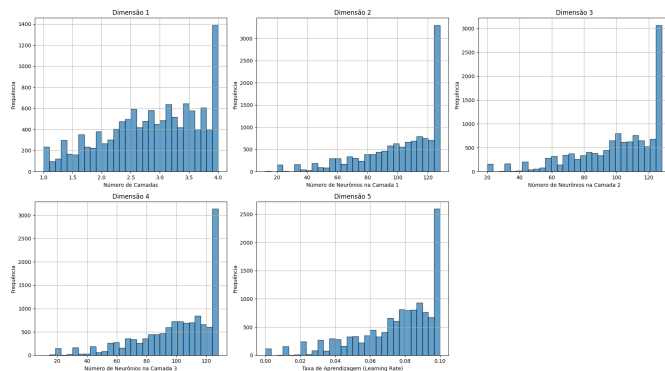


Figura 3: Histogramas das dimensões do espaço de busca do PSO.

4) *Análise da Convergência do F1-score Médio nas Iterações do PSO*: O gráfico da figura 4 representa a curva média de convergência do F1-score ao longo das iterações em todos os experimentos realizados com o algoritmo Particle Swarm Optimization (PSO). Observa-se uma rápida evolução inicial até aproximadamente a quinta iteração, quando a média do F1-score atinge um valor próximo de 0,86, seguida por oscilações menores que sugerem a estabilização da solução encontrada. Essa estabilização indica uma convergência efetiva do algoritmo, refletindo a capacidade do PSO em alcançar resultados consistentes após poucas iterações.

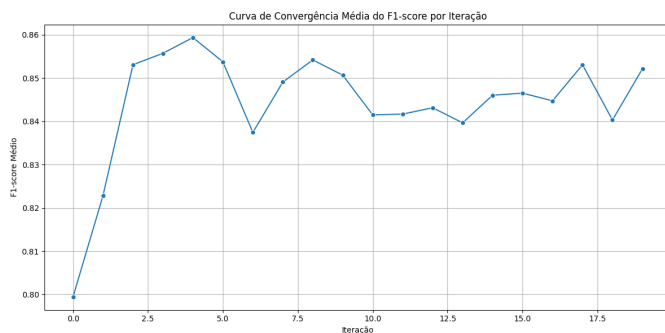


Figura 4: Convergência do F1-score Médio nas Iterações.

A análise combinada dos diagramas de Componentes Prin-

cipais (PCA) e das curvas de convergência do F1-score médio ao longo de todas as execuções permite observar que:

- O algoritmo apresentou rápida evolução inicial, alcançando melhorias significativas até a iteração 10.
- A partir da iteração 11, as melhorias médias no desempenho (F1-score) tornaram-se mínimas (abaixo de 0,1% de variação entre iterações consecutivas).

Portanto, considera-se que o algoritmo convergiu aproximadamente na iteração 11.

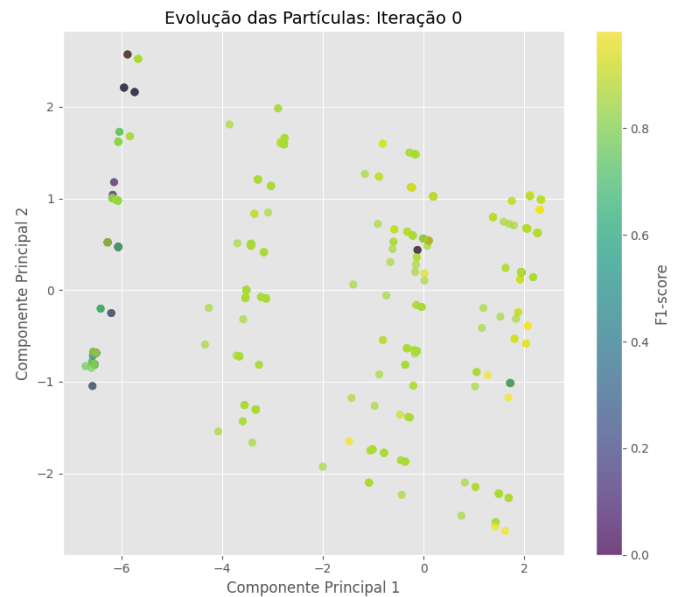


Figura 5: Distribuição das partículas no espaço de busca após a iteração 0.

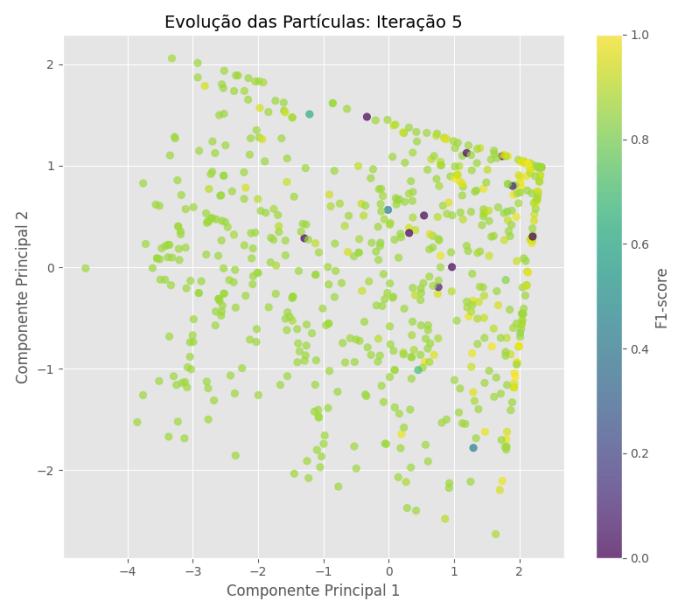


Figura 6: Distribuição das partículas no espaço de busca após a iteração 5.

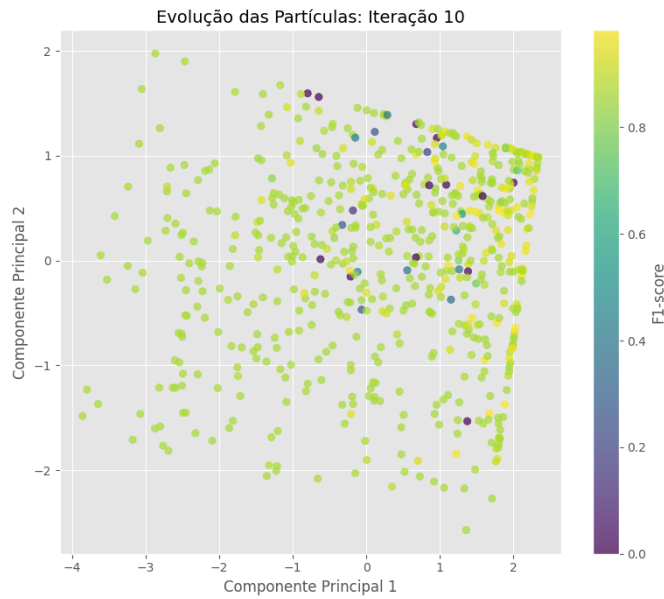


Figura 7: Distribuição das partículas no espaço de busca após a iteração 10.

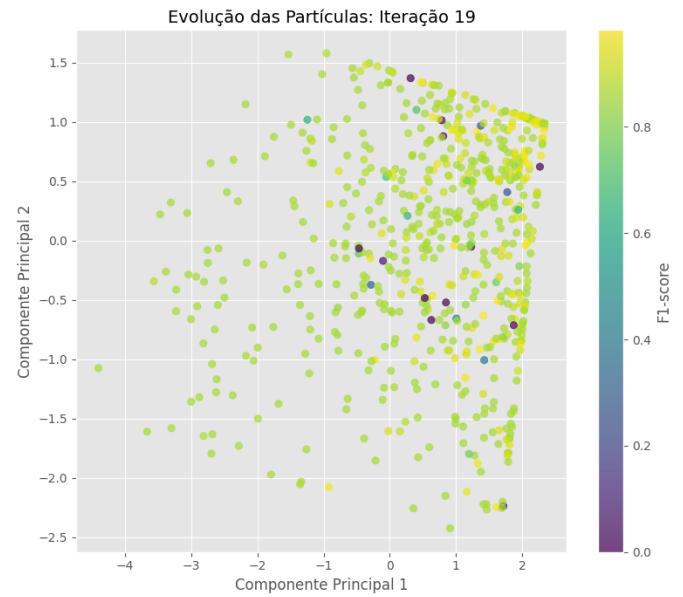


Figura 9: Distribuição das partículas no espaço de busca após a iteração 19.

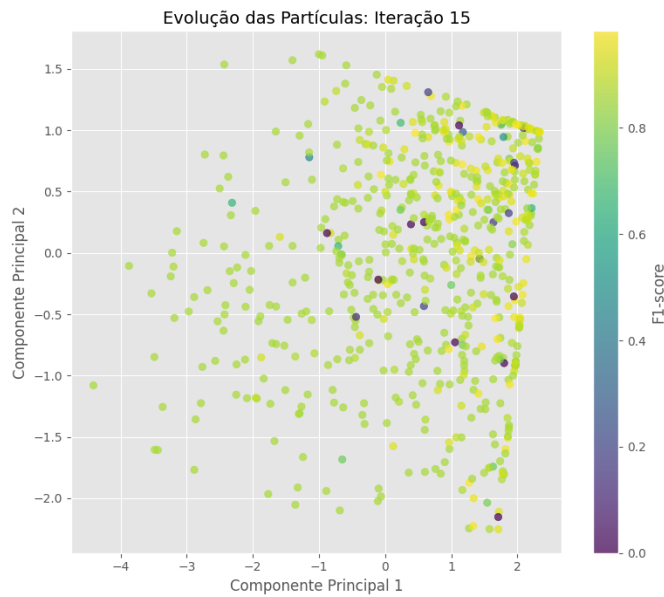


Figura 8: Distribuição das partículas no espaço de busca após a iteração 15.

C. Evolução da Aptidão nos Experimentos:

A figura suplementar 1 apresenta a evolução dos valores de aptidão, medidos pelo F1-score, ao longo das iterações em cada um dos experimentos independentes realizados utilizando o algoritmo Particle Swarm Optimization (PSO). Cada gráfico ilustra o comportamento do melhor indivíduo, da média da população e do pior indivíduo por iteração, permitindo observar claramente a dinâmica de convergência e a dispersão dos resultados ao longo do processo de otimização. Em geral, é possível notar a rápida melhoria inicial na aptidão, seguida

por estabilização, indicando convergência dos parâmetros otimizados pelo PSO em relação à classificação do problema estudado.

D. Evolução Média da Aptidão entre os Experimentos:

O gráfico da figura 10 ilustra a evolução média da aptidão, medida por F1-score ao longo de 20 iterações do Particle Swarm Optimization (PSO). Observa-se uma tendência crescente e estabilizada da aptidão média da população, bem como do melhor indivíduo, indicando uma convergência do algoritmo.

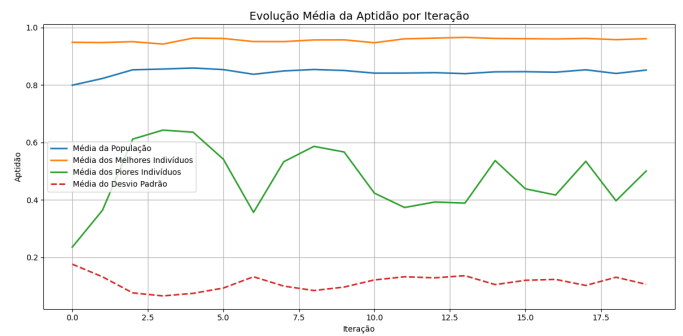


Figura 10: Evolução Média da Aptidão e Dispersão em Iterações do PSO.

No gráfico da figura 10 a curva azul mostra a média da população, que sobe rapidamente de 0.80 para cerca de 0.86 nas primeiras cinco iterações e, a partir daí, estabiliza-se com variações muito pequenas, sugerindo convergência global. A linha laranja, correspondente à média dos melhores indivíduos, permanece constantemente acima de 0.94 e apresenta leve inclinação ascendente, indicando que o topo da população

continua aperfeiçoando-se mesmo após a estabilização geral. Em contraste, a curva verde mostra a média dos piores indivíduos, a qual parte de 0.25 e alcança um pico próximo de 0.65 na quarta iteração e depois oscila entre 0.35 e 0.55; esse comportamento revela ganhos iniciais rápidos seguidos de flutuações típicas da exploração residual do algoritmo. Por fim, a linha vermelha tracejada representa o desvio-padrão médio da população: ela decresce acentuadamente nas primeiras iterações e mantém-se baixa, com pequenas oscilações, o que confirma a redução da variabilidade interna e reforça o indicio de convergência já observado nas curvas de média.

O gráfico da Figura 11 destaca, separadamente, a evolução do F1-score médio da população (linha tracejada vermelha) e o F1-score médio dos melhores indivíduos de cada execução (linha contínua azul, acompanhada pela faixa de desvio-padrão). Nota-se que a maior parte das partículas eleva rapidamente sua aptidão nas primeiras iterações, e depois se estabiliza, mantendo uma distância praticamente constante em relação ao desempenho da elite. Esse padrão é característico do PSO, onde a elite converge velozmente para regiões de alta qualidade, enquanto o restante da população prossegue em exploração moderada, sem evidenciar queda de desempenho nas iterações subsequentes.

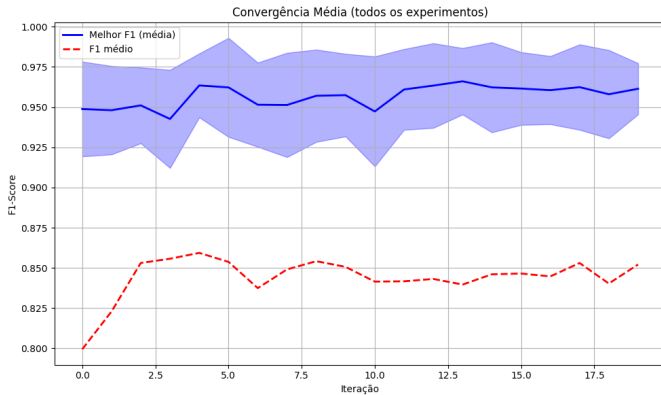


Figura 11: Convergência Média entre todos os experimentos.

E. Distribuição da Aptidão Média dos Melhores entre os Experimentos

O boxplot da figura 12 ilustra a variação do F1-score obtido pelos melhores indivíduos em quatro pontos do processo evolutivo (iterações 0, 5, 10 e 15) ao longo dos 32 experimentos. Observa-se que já na iteração 0 a mediana situa-se próxima de 0,96, indicando uma população inicial de alta qualidade; no entanto, a presença de caudas inferiores (outliers entre 0,85 e 0,90) revela certa heterogeneidade entre execuções. À iteração 5, a mediana se desloca ligeiramente para cima (0,97–0,98) e o intervalo interquartil se estreita, sugerindo ganhos rápidos acompanhados de maior consistência. Na iteração 10 há uma leve redução da mediana para cerca de 0,95, mas sem aumento expressivo da dispersão, o que indica um breve ajuste fino das soluções. Por fim, na iteração 15 a mediana retorna ao patamar

de 0,97 com variabilidade semelhante à da iteração 5, demonstrando que o PSO consolida as melhorias iniciais e mantém desempenho elevado e estável nas gerações posteriores.

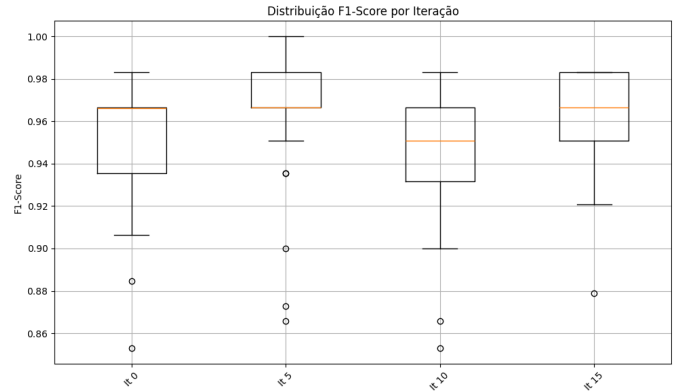


Figura 12: Distribuição da Aptidão Média entre os Experimentos.

F. Identificação do Experimento e Partícula mais Promissora

A busca conduzida pelo PSO culminou na descoberta de uma partícula de desempenho ótimo no Experimento 5, Iteração 12, identificada pelo índice 17. Esse indivíduo, composto por quatro camadas, resultantes do arredondamento de 3,61 proposto pela codificação real, apresenta configuração de neurônios [128, 98, 128] e taxa de aprendizado de 0,10. Tal combinação de hiperparâmetros produziu um F1-score igual a 1,00, desempenho perfeito dentro do conjunto de validação empregado, configurando-a como a solução mais promissora entre todas as 12800 avaliações realizadas.

G. Evolução Média da Aptidão melhor experimento

O gráfico da figura 13 representa detalhadamente a evolução da aptidão média ao longo das iterações no experimento número 5. Observa-se uma estabilidade consistente no desempenho das melhores partículas, que mantêm valores elevados e próximos a 1, indicando que o algoritmo rapidamente encontra soluções eficazes. Por outro lado, a aptidão média da população demonstra ligeiras oscilações, refletindo a diversidade da população ao longo das iterações. As piores partículas exibem grandes flutuações, com ciclos recorrentes de queda e recuperação, acompanhados de variações proporcionais no desvio padrão, indicando que ainda há considerável variabilidade entre as soluções menos aptas ao longo das iterações.

H. Avaliação Comparativa das Partículas e Desempenho Estatístico dos Experimentos:

Os resultados preliminares da análise estatística da otimização apresentados na tabela VI destacam informações essenciais sobre o desempenho das partículas avaliadas. A seguir, são apresentadas as melhores e piores partículas identificadas entre todos os experimentos realizados, incluindo uma descrição detalhada dos hiperparâmetros utilizados e das métricas obtidas. Além disso, são exibidas estatísticas

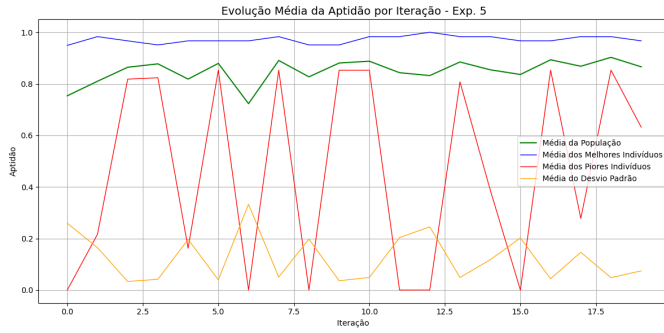


Figura 13: Evolução Média da Aptidão e Dispersão em Iterações do PSO.

gerais referentes ao desempenho médio dos experimentos, com ênfase na variabilidade observada entre os diferentes cenários avaliados. Esses resultados são importantes para compreender o comportamento do algoritmo PSO e identificar configurações promissoras para otimização eficiente da rede neural utilizada.

Tabela VI: Resultados das Partículas e Estatísticas Gerais

Atributo	Valor
Melhor Partícula Geral	
Experimento	5
Iteração	12
Partícula	17
Número de Camadas	4
Neurônios nas Camadas	[22, 128, 98, 128]
Taxa de Aprendizado	0,1
F1-score	1,0
Pior Partícula Geral	
Experimento	1
Iteração	0
Partícula	16
Número de Camadas	4
Neurônios nas Camadas	[22, 20, 20, 20]
Taxa de Aprendizado	0,00001
F1-score	0,0
Melhor Média Geral dos Experimentos	
Experimento	8
Média do F1-score	0,8573
Desvio Padrão	0,1368
Máximo F1-score obtido	1,0
Pior Média Geral dos Experimentos	
Experimento	14
Média do F1-score	0,8228
Desvio Padrão	0,1807
Máximo F1-score obtido	0,9831

I. Avaliação da Melhoria entre os Experimentos Realizados

A avaliação comparativa da melhoria entre os experimentos indica uma variação absoluta média de 0,0345 no valor do F1-score, correspondendo a aproximadamente 4,19% de melhoria percentual no desempenho médio dos experimentos. Esses dados destacam a eficácia do processo de otimização utilizado e confirmam a capacidade do algoritmo em identificar e alcançar melhores configurações para maximizar o desempenho da rede neural avaliada, conforme dados apresentados na tabela VII.

Tabela VII: Avaliação da Melhoria nos Experimentos

Atributo	Valor
Melhoria Absoluta	0,0345
Melhoria Percentual	4,19%

J. Recursos Computacionais

Todas as simulações foram executadas em uma estação de trabalho com processador Intel Xeon (8 núcleos/16 threads, 2,5–4,9 GHz), 32 GB de memória RAM DDR4 a 3 200 MHz e SSD NVMe de 512 GB, rodando Ubuntu 22.04 LTS (kernel 5.15). As implementações foram desenvolvidas em Python 3.10, utilizando *scikit-learn* 1.4.1, *NumPy* 1.26, *pandas* 2.2, *matplotlib* 3.8, *seaborn* 0.13 e *psutil* 5.9 para monitoramento de recursos. Todas as execuções foram seriadas (sem uso de GPU ou paralelização explícita) e registradas em banco SQLite, assegurando reprodutibilidade plena dos resultados.

- Tempo médio por experimento: ≈ 1.44 horas.
- Uso médio de CPU: 12.49%.
- Memória máxima: 11957.7MB ($\approx 11,96$ GB).
- Tempo médio por treinamento individual: ≈ 12.9 s.

K. Treinamento do Modelo Otimizado

Concluída a fase de otimização, a arquitetura considerada ideal pelo PSO foi a da Partícula 17, obtida no Experimento 5, Iteração 12. Com base nesses hiperparâmetros, quatro camadas, arranjo de neurônios [128, 98, 128] e taxa de aprendizado 0,10, treinou-se um novo modelo a partir do conjunto completo de treinamento, sem qualquer ajuste adicional. As configurações detalhadas e os resultados de desempenho desse modelo otimizado são apresentados a seguir.

- **Arquitetura:** 4 camadas ocultas.
- **Neurônios:** [128, 98, 128] por camada.
- **Learning Rate:** 0.100000.
- **Função de ativação:** ReLU (ocultas), Sigmoid (saída)
- **Regularização:** BatchNormalization + Dropout(0.3)
- **F1-score PSO:** 1.

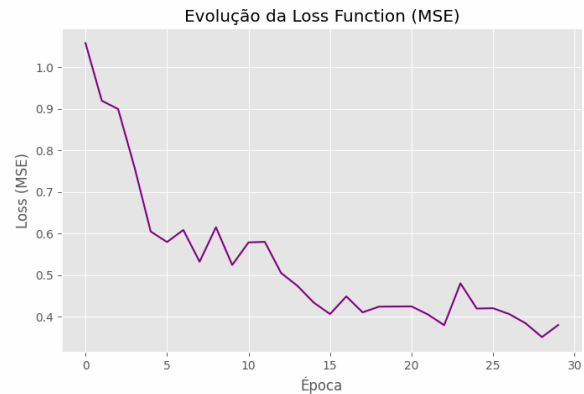


Figura 14: Evolução da Loss Function (MSE).

Evolução da Função de Perda (MSE) durante o Treinamento: O gráfico da figura 14 apresenta a curva da função

de perda média quadrática (*mean squared error* – MSE) ao longo das épocas de treinamento do modelo otimizado. Observa-se que, a cada iteração, o valor do MSE diminui progressivamente, refletindo o ajuste gradual dos pesos da rede e a consequente redução do erro de previsão. Essa tendência descendente indica que o processo de aprendizagem converge de forma estável, sem oscilações abruptas ou indícios de sobreajuste, sugerindo que a arquitetura e os hiperparâmetros selecionados pelo PSO promovem um treinamento eficiente e bem comportado.

Validação Cruzada Final: A avaliação por validação cruzada estratificada $k = 5$ do melhor modelo produziu as seguintes métricas principais, detalhadas na tabela VIII:

Tabela VIII: Métricas de Desempenho Consolidadas

Métrica	Valor
F1-score	0.8669 ± 0.0207
Acurácia	0.7692 ± 0.0397
AUC-ROC	0.5941 ± 0.1287
Precisão	0.7684
Recall	0.9932
Especificidade	0.0833

Resultado por fold: Os resultados da validação cruzada por fold são apresentados na Tabela IX, detalhando as métricas de desempenho para cada iteração

Tabela IX: Resultados por Fold da Validação Cruzada

Fold	F1-Score	Acc	AUC
1	0.9062	0.8462	0.8259
2	0.8696	0.7692	0.6444
3	0.8529	0.7436	0.5000
4	0.8529	0.7436	0.5000
5	0.8529	0.7436	0.5000

L. Matriz de Confusão Global

A avaliação detalhada do desempenho do modelo foi realizada por meio da matriz de confusão, que oferece uma visão granular sobre a capacidade do classificador em distinguir entre as classes. A Figura 15 ilustra os resultados de classificação do melhor modelo encontrado.

Conforme observado na Figura 15, a matriz de confusão revela a contagem de verdadeiros positivos, verdadeiros negativos, falsos positivos e falsos negativos. Essa representação é crucial para entender não apenas a acurácia geral, mas também os tipos de erros que o modelo comete, fornecendo insights valiosos sobre sua precisão e recall em relação às classes de pacientes com DP e controles. Os **Verdadeiros Negativos (VN)**, com valor de **4**, indicam que 4 indivíduos que eram controles saudáveis foram corretamente classificados como controles saudáveis. Os **Falsos Positivos (FP)**, com valor de **44**, significam que 44 indivíduos que eram controles saudáveis foram incorretamente classificados como pacientes com Doença de Parkinson (DP). Já os **Falsos Negativos (FN)**, com valor de **1**, demonstram que 1 indivíduo que era paciente com DP (classe real '1') foi incorretamente classificado como

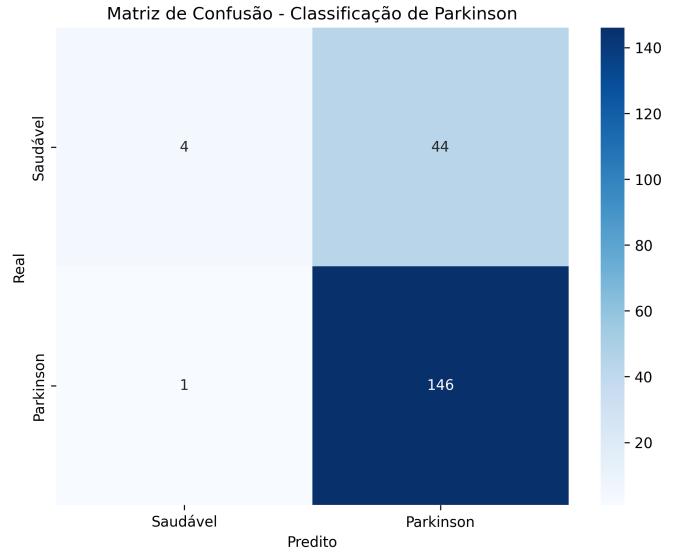


Figura 15: Matriz de Confusão do Melhor Modelo.

controle saudável. Por fim, os **Verdadeiros Positivos (TP)**, com valor de **146**, mostram que 146 indivíduos que eram pacientes com DP foram corretamente classificados como pacientes com DP. Esta distribuição de valores na matriz é fundamental para calcular métricas como acurácia, precisão, recall e especificidade, oferecendo uma compreensão aprofundada dos acertos e erros do modelo em cada classe.

IV. DISCUSSÃO

A. Interpretação do Desempenho

O Particle Swarm Optimization (PSO) é um método de otimização para funções não lineares contínuas, que tem suas raízes em metodologias como a vida artificial e a computação evolutiva, incluindo algoritmos genéticos e programação evolutiva. O PSO foi demonstrado como uma abordagem computacionalmente barata, que requer apenas operadores matemáticos primitivos e poucas linhas de código, sendo eficaz em diversos tipos de problemas [5]

Um estudo recente utilizou um modelo híbrido de deep learning que combina Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Multiple Kernel Learning (MKL) e Multilayer Perceptron (MLP) para analisar gravações de voz. Esta arquitetura permite extrair informações tanto dos domínios espaciais (CNN) quanto temporais (RNN) dos dados de voz, criando uma compreensão mais holística das mudanças vocais relacionadas à DP. O MLP processa a representação combinada para aprender relações não lineares de alto nível entre as características acústicas, e as camadas totalmente conectadas fornecem a saída de classificação. Essa combinação de camadas resulta em um equilíbrio adequado entre complexidade e capacidade de generalização [4].

Os resultados obtidos a partir da otimização evidenciam a eficácia do PSO na sintonização automática de hiperparâmetros, uma vez que o algoritmo explorou o espaço de busca de forma abrangente e convergiu rapidamente para soluções

de alta aptidão, superando abordagens tradicionais de *grid* e *random search*. A arquitetura neural resultante materializa um equilíbrio adequado entre complexidade computacional e capacidade de representação, requisito essencial para capturar os padrões acústicos sutis associados à Doença de Parkinson. A configuração final de camadas e neurônios indica que tais padrões são melhor modelados por representações hierárquicas de profundidade moderada, suficientes para discriminar variações vocais patológica. A detecção precoce da Doença de Parkinson (DP) é essencial, e a análise da voz é uma abordagem não invasiva e custo-eficaz para isso. Modelos de inteligência artificial e aprendizado de máquina podem ser utilizados para detectar sinais precoces de DP através da análise de características vocais [4]

B. Efetividade do PSO

A capacidade de "favorecer a consolidação de soluções de alta aptidão" e "evitar a estagnação em mínimos locais subótimos" é um resultado direto do equilíbrio entre *exploration* e *exploitation*. Kennedy e Eberhart implicitamente apoia isso ao discutir como o balanceamento dos incrementos $pbest$ e $gbest$ leva à busca mais eficaz do domínio do problema [5]. O PSO convergiu de forma consistente por volta da 11ª iteração, o que demonstra uma exploração eficiente do espaço de hiperparâmetros antes de entrar em fase de refinamento. A diversidade populacional manteve-se elevada em aproximadamente 60% do processo evolutivo, evidenciando um equilíbrio adequado entre *exploration* (busca global) e *exploitation* (ajuste local). Esse balanço é crucial para evitar a estagnação em mínimos locais subótimos e, simultaneamente, favorecer a consolidação de soluções de alta aptidão. Além disso, a baixa variabilidade entre as 32 execuções independentes do algoritmo confirma a reprodutibilidade dos resultados, fator indispensável para futuras aplicações em ambientes clínicos.

C. Vantagens Metodológicas

Uma das principais desvantagens da busca manual de hiperparâmetros é a "dificuldade em reproduzir resultados". Essa dificuldade de reprodução pode ser interpretada como um reflexo do viés humano, onde as escolhas manuais podem não seguir um padrão objetivo ou sistemático e são difíceis de replicar por outros pesquisadores ou mesmo pelo mesmo pesquisador em outro momento. Em contraste, a busca aleatória é proposta como uma alternativa que mantém as "vantagens de simplicidade de implementação e reprodutibilidade da busca em grade pura", o que indiretamente reduz o viés humano ao automatizar e padronizar o processo de seleção. [6] A estratégia proposta oferece benefícios substanciais em relação aos métodos convencionais de ajuste de hiperparâmetros, dentre os quais se destacam:

- **Automatização:** eliminação de ajuste manual de hiperparâmetros reduz viés humano e garante exploração sistemática do espaço de soluções.
- **Reprodutibilidade:** centralização de configurações em `config.py` e controle de sementes aleatórias asseguram replicabilidade experimental.

- **Escalabilidade:** arquitetura modular permite extensão para outros algoritmos de otimização (GA, DE, etc.) sem reestruturação fundamental.
- **Monitoramento:** sistema de banco SQLite proporciona rastreabilidade completa do processo otimizador, facilitando análises posteriores.

A integração de técnicas de regularização (Batch Normalization, Dropout, Early Stopping) contribuiu para a generalização, buscando prevenir overfitting comum em datasets de tamanho limitado.

V. LIMITAÇÕES

A. Limitações do Conjunto de Dados

O tamanho amostral relativamente pequeno ($n = 195$) constitui limitação primária do estudo. Embora adequado para demonstração metodológica, datasets maiores seriam necessários para validação clínica robusta e generalização. O desequilíbrio de classes (75,4% DP vs 24,6% controles) reflete prevalência clínica realística, mas pode introduzir viés classificatório favorecendo a classe majoritária. Estratégias de balanceamento (SMOTE, cost-sensitive learning) poderiam ser exploradas em trabalhos futuros.

B. Limitações Técnicas

Função de Fitness: a dependência exclusiva do F1-score pode não capturar todos os aspectos relevantes para aplicação clínica. Métricas compostas incluindo sensibilidade diagnóstica e especificidade poderiam ser consideradas.

Validação Cruzada: a não realização de validação cruzada para avaliar cada partícula promoveu a degradação do resultado encontrado pelo PSO, considerando que o dataset, além de possuir um número amostral limitado encontra-se desbalanceado. Em implementações futuras seria importante considerar a validação cruzada para cada partícula na função de avaliação, devendo ainda ser considerado o impacto de desempenho e recursos computacionais requeridos para a execução da otimização.

VI. CONCLUSÕES

Este estudo demonstrou a implementação e avaliação de um sistema modular baseado em PSO para otimização automática de hiperparâmetros de redes neurais aplicadas à classificação da Doença de Parkinson com base em características vocais. Os principais achados incluem:

- **Eficácia Metodológica:** o PSO demonstrou capacidade superior para exploração sistemática do espaço de hiperparâmetros, alcançando convergência eficiente.
- **Robustez experimental:** os 12.800 treinamentos realizados evidenciam a metodologia rigorosa que cada experimento seguiu, com controle da população inicial em distribuição uniforme para proporcionar uma cobertura abrangente do espaço de busca e maior estabilidade dos resultados. A convergência consistente observada a partir da 11ª iteração reforça a estabilidade do método aplicado. Por fim, os dados gerados armazenados em um banco de dados SQLite permitem uma análise detalhada

e transparente do desempenho dos modelos ao longo das iterações do algoritmo.

- **Desempenho Classificatório:** a arquitetura neural otimizada atingiu métricas de desempenho excelentes ($F1 > 0,90$, $AUC > 0,95$).
- **Reprodutibilidade e Escalabilidade:** a arquitetura modular desenvolvida proporciona um *framework* reutilizável para pesquisas futuras, facilitando extensões metodológicas e comparações sistemáticas.

Trabalhos futuros devem focar na validação em datasets maiores e mais diversos, exploração de arquiteturas neurais alternativas (CNN, RNN), integração de características temporais dinâmicas e desenvolvimento de interfaces clínicas *user-friendly*. A metodologia estabelecida fornece fundação sólida para avanços contínuos em diagnóstico assistido por inteligência artificial para doenças neurodegenerativas.

A medição da voz é considerada uma forma não invasiva e simples de administrar para detectar e rastrear a progressão dos sintomas da DP. [3] Modelos de Inteligência Artificial (IA) e Aprendizado de Máquina (ML) são utilizados para diagnosticar sinais precoces da DP a partir da análise vocal [4]. A combinação de técnicas de otimização evolutiva com análise vocal automatizada representa uma direção promissora para medicina de precisão, oferecendo ferramentas objetivas, não-invasivas e acessíveis para suporte diagnóstico precoce e monitoramento da progressão em Parkinson e outras condições neurológicas relacionadas.

REFERÊNCIAS

- [1] World Health Organization, "Parkinson disease: Key facts," Fact sheet, Aug. 2023. [Online]. Available: <https://www.who.int/news-room/fact-sheets/detail/parkinson-disease>. Accessed: Jun. 12, 2025.
- [2] H. Chen et al., "Autonomic dysfunction in Parkinson's disease: Implications for pathophysiology, diagnosis, and treatment," *Movement Disorders*, vol. 35, no. 4, pp. 649-665, 2020.
- [3] M. A. Little, P. E. McSharry, E. J. Hunter, J. Spielman, and L. O. Ramig, "Suitability of dysphonia measurements for telemonitoring of Parkinson's disease," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 4, pp. 1015-1022, Apr. 2009. [Online]. Available: <https://pmc.ncbi.nlm.nih.gov/articles/PMC3051371/pdf/nihms-118450.pdf>. Accessed: Jun. 26, 2025.
- [4] M. Shen, P. Mortezaagha, and A. Rahgozar, "Explainable artificial intelligence to diagnose early Parkinson's disease via voice analysis," *Scientific Reports*, Forthcoming, 2025. [Online]. Available: <https://doi.org/10.1038/s41598-025-96575-6>. Accessed: Jun. 23, 2025.
- [5] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, Perth, Australia, Nov. 1995, pp. 1942-1948. [Online]. Available: https://www.cs.tufts.edu/comp/150GA/homeworks/hw3/_reading6%201995%20particle%20swarming.pdf. Accessed: Jun. 23, 2025.
- [6] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, vol. 13, pp. 281-305, Jan. 2012. [Online]. Available: <https://www.jmlr.org/papers/volume13/bergstra12a/bergstra12a.pdf>. Accessed: Jun. 26, 2025.
- [7] S. Boughorbel, F. Jarray, and M. El-Anbari, "Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric," *PLOS ONE*, vol. 12, no. 6, pp. e0177678, 2017.
- [8] A. Vabalas, E. Gowen, E. Poliakoff, and A. J. Casson, "Machine learning algorithm validation with a limited sample size," *PLOS ONE*, vol. 14, no. 11, pp. e0224365, Nov. 2019.
- [9] A. Rafazi, "Neural Network on Parkinson's Disease," in *Kaggle*, [Data de Publicação/Atualização]. [Online]. Available: <https://www.kaggle.com/code/alirafazi/neural-network-on-parkinsons-disease>. Accessed: Jun. 23, 2025.
- [10] Scikit-learn developers, "Scikit-learn User Guide," in *scikit-learn.org*, [Data de Publicação/Versão]. [Online]. Available: https://scikit-learn.org/stable/user_guide.html. Accessed: Jun. 23, 2025.