



# **Relatório de Resultados**

**TÓPICOS EM INTELIGÊNCIA ARTIFICIAL II**

**Universidade Federal Fluminense**

Camila Ferreira

# 1. Datasets Escolhidos:

## Dataset 1:

### Estimation of Obesity Levels Dataset

Fonte: [UCI Machine Learning Repository - Dataset 544](#)

#### Descrição

Este dataset foi criado com o objetivo de prever os níveis de obesidade com base em hábitos alimentares e condições físicas dos indivíduos. Ele foi coletado por meio de questionários preenchidos por pessoas no México, Peru e Colômbia. A base contém variáveis relacionadas a estilo de vida, como frequência de exercícios, consumo de alimentos e histórico de peso.

#### Informações do Dataset

- **Tamanho:** 2111 instâncias
- **Atributos:** 17 atributos

#### Atributo Alvo (Target)

- NObeyesdad: Nível de obesidade do indivíduo
  - Normal\_Weight
  - Overweight\_Level\_I
  - Overweight\_Level\_II
  - Obesity\_Type\_I
  - Insufficient\_Weight
  - Obesity\_Type\_II
  - Obesity\_Type\_III

| Atributo                       | Tipo       | Descrição                                   |
|--------------------------------|------------|---|
| Gender                         | Categórica | Gênero do indivíduo                         |
| Age                            | Numérica   | Idade                                       |
| Height                         | Numérica   | Altura em metros                            |
| Weight                         | Numérica   | Peso em kg                                  |
| family_history_with_overweight | Booleana   | Histórico familiar de sobrepeso             |
| FAVC                           | Booleana   | Consome alimentos calóricos com frequência? |
| FCVC                           | Numérica   | Frequência de consumo de vegetais           |
| NCP                            | Numérica   | Número de refeições principais por dia      |
| CAEC                           | Categórica | Consumo de alimentos entre as refeições     |
| SMOKE                          | Booleana   | Fuma  |

| Atributo | Tipo       | Descrição   |
|----------|------------|---|
| CH2O     | Numérica   | Consumo diário de água (litros)                   |
| SCC      | Booleana   | Monitora ingestão calórica?                       |
| FAF      | Numérica   | Frequência de atividade física (horas por semana) |
| TUE      | Numérica   | Tempo usando tecnologia por dia (horas)           |
| CALC     | Categórica | Frequência de consumo de álcool                   |
| MTRANS   | Categórica | Meio de transporte mais utilizado                 |

## Dataset 2:

### Contraceptive Method Choice Dataset

Fonte: [UCI Machine Learning Repository - Dataset 30](#)

#### Descrição

Este dataset foi desenvolvido com o objetivo de prever o método contraceptivo utilizado por mulheres com base em informações socioeconômicas e demográficas. A base é derivada de um estudo realizado na Indonésia e é composta por dados coletados de mulheres casadas que participavam do programa nacional de planejamento familiar. O foco principal é auxiliar na análise de padrões que influenciam a escolha de métodos contraceptivos.

#### Informações do Dataset

- **Tamanho:** 1473 instâncias
- **Atributos:** 9 atributos

#### Atributo Alvo (Target)

- **Contraceptive Method Used:** Método contraceptivo utilizado pela mulher
  - 1: Nenhum
  - 2: Curto prazo (pílulas, preservativos, etc.)
  - 3: Longo prazo (DIU, esterilização, etc.)

| Atributo          | Tipo       | Descrição  |
|-------------------|------------|--|
| Wife_age          | Numérica   | Idade da esposa (em anos)                        |
| Wife_education    | Categórica | Grau de escolaridade da esposa (1=baixo, 4=alto) |
| Husband_education | Categórica | Grau de escolaridade do marido (1=baixo, 4=alto) |
| Num_children      | Numérica   | Número de filhos vivos                           |
| Wife_religion     | Booleana   | Religião da esposa (0=Islâmica, 1=Outra)         |

| Atributo           | Tipo       | Descrição                                    |
|--------------------|------------|--|
| Wife_working       | Booleana   | A esposa trabalha? (0=Não, 1=Sim)            |
| Husband_occupation | Categórica | Ocupação do marido (valores de 1 a 4)        |
| Standard_of_living | Categórica | Nível de vida (1=baixo, 4=alto)              |
| Media_exposure     | Booleana   | Exposição à mídia (0=Não exposta, 1=Exposta) |

## Dataset 3:

### Heart Disease Dataset

Fonte: [UCI Machine Learning Repository - Dataset 45](#)

#### Descrição

Este dataset foi criado com o objetivo de prever a presença de doenças cardíacas em pacientes com base em diversas características clínicas e demográficas. A base contém dados coletados de quatro bancos hospitalares diferentes, sendo a versão mais utilizada aquela que consolida as informações em um único conjunto limpo e padronizado. Os atributos incluem variáveis como idade, sexo, pressão arterial, nível de colesterol e resultados de exames cardíacos.

#### Informações do Dataset

- **Tamanho:** 303 instâncias (na versão Cleveland, a mais usada)
- **Atributos:** 13 atributos

#### Atributo Alvo (Target)

- **Grau de doença cardíaca presente no indivíduo :** valores de 0 a 4, onde 0 significa ausência da doença

| Atributo | Tipo       | Descrição   |
|----------|------------|---|
| age      | Numérica   | Idade do paciente   |
| sex      | Binária    | Sexo (1 = masculino; 0 = feminino)                        |
| cp       | Categórica | Tipo de dor no peito (0 a 3)                              |
| trestbps | Numérica   | Pressão arterial em repouso (mm Hg)                       |
| chol     | Numérica   | Nível sérico de colesterol (mg/dl)                        |
| fbs      | Binária    | Glicemia em jejum > 120 mg/dl (1 = verdadeiro; 0 = falso) |
| restecg  | Categórica | Resultados do eletrocardiograma em repouso                |
| thalach  | Numérica   | Frequência cardíaca máxima atingida                       |
| exang    | Binária    | Angina induzida por exercício (1 = sim; 0 = não)          |

| Atributo | Tipo       | Descrição  |
|----------|------------|--|
| oldpeak  | Numérica   | Depressão do segmento ST induzida por exercício em relação ao repouso      |
| slope    | Categórica | Inclinação do segmento ST no esforço máximo                                |
| ca       | Numérica   | Número de vasos principais coloridos por fluoroscopia (0 a 3)              |
| thal     | Categórica | Resultado do teste de tálio (3 = normal; 6 = defeito fixo; 7 = reversível) |

## Dataset 4:

### Letter Recognition Dataset

Fonte: [UCI Machine Learning Repository - Dataset 59](#)

#### Descrição

Este dataset foi criado com o objetivo de classificar letras do alfabeto inglês maiúsculas com base em diversas características extraídas de imagens. Os dados foram gerados a partir de imagens de letras capturadas e processadas para extrair atributos relacionados à geometria e distribuição de pixels. O conjunto de dados é frequentemente usado em estudos de reconhecimento óptico de caracteres (OCR) e aprendizado supervisionado.

#### Informações do Dataset

- **Tamanho:** 20.000 instâncias
- **Atributos:** 16 atributos preditivos + 1 alvo (letra)

#### Atributo Alvo (Target)

- Letra do alfabeto (A a Z)
  - Cada instância é rotulada com uma letra maiúscula de 'A' até 'Z'

| Atributo | Tipo       | Descrição   |
|----------|------------|---|
| lettr    | Categórica | Letra correspondente à instância (A a Z)          |
| x-box    | Numérica   | Posição horizontal da caixa delimitadora da letra |
| y-box    | Numérica   | Posição vertical da caixa delimitadora da letra   |
| width    | Numérica   | Largura da caixa delimitadora                     |
| high     | Numérica   | Altura da caixa delimitadora                      |
| onpix    | Numérica   | Número de pixels dentro da caixa                  |
| x-bar    | Numérica   | Centro de massa horizontal                        |
| y-bar    | Numérica   | Centro de massa vertical                          |
| x2bar    | Numérica   | Variância horizontal                              |

| Atributo | Tipo     | Descrição                        |
|----------|----------|----------------------------------|
| y2bar    | Numérica | Variância vertical               |
| xybar    | Numérica | Correlação entre x e y           |
| x2ybr    | Numérica | Correlação de segunda ordem x*y  |
| xy2br    | Numérica | Correlação de segunda ordem y*x  |
| x-ege    | Numérica | Entropia do perfil horizontal    |
| xegvy    | Numérica | Entropia da variância horizontal |
| y-ege    | Numérica | Entropia do perfil vertical      |
| yegvx    | Numérica | Entropia da variância vertical   |

## Dataset 5:

### Dermatology Dataset

Fonte: [UCI Machine Learning Repository - Dataset 33](#)

#### Descrição

Este dataset foi criado com o objetivo de auxiliar no diagnóstico de doenças de pele eritemato-esquamosas, que apresentam sintomas semelhantes como eritema (vermelhidão) e descamação, tornando o diagnóstico diferencial um desafio clínico. O conjunto de dados combina informações clínicas e histopatológicas (obtidas por biópsias) para classificar seis tipos diferentes de doenças dermatológicas.

#### Informações do Dataset

- **Tamanho:** 366 instâncias
- **Atributos:** 34 atributos

#### Atributo Alvo (Target)

- **class:** Tipo de doença dermatológica
  - 1 – Psoríase (*psoriasis*)
  - 2 – Dermatite seborreica (*seborreic dermatitis*)
  - 3 – Líquen plano (*lichen planus*)
  - 4 – Pitiríase rosada (*pityriasis rosea*)
  - 5 – Dermatite crônica (*chronic dermatitis*)
  - 6 – Pitiríase rubra pilar (*pityriasis rubra pilaris*)

| Atributo         | Tipo     | Descrição                            |
|------------------|----------|--------------------------------------|
| erythema         | Numérica | Grau de eritema (0 a 3)              |
| scaling          | Numérica | Grau de descamação (0 a 3)           |
| definite_borders | Numérica | Presença de bordas definidas (0 a 3) |

| Atributo                             | Tipo     | Descrição  |
|--------------------------------------|----------|--|
| itching                              | Numérica | Grau de coceira (0 a 3)                                |
| koebner_phenomenon                   | Numérica | Fenômeno de Koebner (0 a 3)                            |
| polygonal_papules                    | Numérica | Pápulas poligonais (0 a 3)                             |
| follicular_papules                   | Numérica | Pápulas foliculares (0 a 3)                            |
| oral_mucosal_involvement             | Numérica | Envolvimento da mucosa oral (0 a 3)                    |
| knee_and_elbow_involvement           | Numérica | Envolvimento de joelhos/cotovelos (0 a 3)              |
| scalp_involvement                    | Numérica | Envolvimento do couro cabeludo (0 a 3)                 |
| family_history                       | Booleana | Histórico familiar de doença (0 = não, 1 = sim)        |
| melanin_incontinence                 | Numérica | Incontinência de melanina (0 a 3)                      |
| eosinophils_infiltrate               | Numérica | Presença de eosinófilos no infiltrado (0 a 3)          |
| PNL_infiltrate                       | Numérica | Presença de polimorfonucleares no infiltrado (0 a 3)   |
| fibrosis_of_papillary_dermis         | Numérica | Fibrose da derme papilar (0 a 3)                       |
| exocytosis                           | Numérica | Grau de exocitose (0 a 3)                              |
| acanthosis                           | Numérica | Grau de acantose (0 a 3)                               |
| hyperkeratosis                       | Numérica | Grau de hiperqueratose (0 a 3)                         |
| parakeratosis                        | Numérica | Grau de paraqueratose (0 a 3)                          |
| clubbing_of_rete_ridges              | Numérica | Deformação das cristas da epiderme (0 a 3)             |
| elongation_of_rete_ridges            | Numérica | El alongamento das cristas da epiderme (0 a 3)         |
| thinning_of_suprapapillary_epidermis | Numérica | Afinamento da epiderme suprapapilar (0 a 3)            |
| spongiform_pustule                   | Numérica | Presença de pústula espongiforme (0 a 3)               |
| munro_microabcess                    | Numérica | Presença de microabscesso de Munro (0 a 3)             |
| focal_hypergranulosis                | Numérica | Hipergranulose focal (0 a 3)                           |
| disappearance_granular_layer         | Numérica | Desaparecimento da camada granulosa (0 a 3)            |
| vacuolisation_basal_layer            | Numérica | Vacuolização/dano na camada basal (0 a 3)              |
| spongiosis                           | Numérica | Grau de espongiose (0 a 3)                             |
| saw_tooth_retes                      | Numérica | Cristas epidérmicas em forma de dente de serra (0 a 3) |
| follicular_horn_plug                 | Numérica | Plugue córneo folicular (0 a 3)                        |
| perifollicular_parakeratosis         | Numérica | Paraqueratose perifolicular (0 a 3)                    |

| Atributo                            | Tipo     | Descrição                                   |
|-------------------------------------|----------|---|
| inflammatory_mononuclear_infiltrate | Numérica | Infiltrado inflamatório mononuclear (0 a 3) |
| band_like_infiltrate                | Numérica | Infiltrado em faixa (0 a 3)                 |
| age                                 | Numérica | Idade do paciente                           |

## 2. Tratamento de dados

Para os experimentos realizados com **Multi-Layer Perceptron (MLP)** e **WiSARD**, o pré-processamento dos dados seguiu etapas similares em diversos aspectos, com adaptações específicas para atender às necessidades de cada modelo.

- **Identificação de tipos de dados:** Em ambos os casos, as colunas do conjunto de dados foram separadas entre **numéricas** e **categóricas**, sendo que variáveis binárias (com valores como "yes"/"no") foram tratadas como **categóricas**.
- **Codificação de variáveis categóricas:** Tanto no MLP quanto no WiSARD, as variáveis categóricas foram transformadas utilizando o **OneHotEncoder**, convertendo cada categoria em um vetor binário. Isso garantiu que as categorias fossem representadas de forma apropriada e compatível com os dois modelos.
- **Normalização de dados numéricos:**
  - No **MLP**, os atributos numéricos foram padronizados utilizando o **StandardScaler**, para que tivessem média 0 e desvio padrão 1, o que facilita o treinamento de redes neurais.
  - No **WiSARD**, os atributos numéricos foram normalizados para o intervalo **[0, 1]** com o **MinMaxScaler**, preparando os valores para a etapa de codificação binária.
- **Codificação dos atributos numéricos:**
  - No **MLP**, os dados numéricos padronizados foram diretamente utilizados na rede.
  - No **WiSARD**, os dados normalizados passaram por um **thermometer encoding**, transformando cada valor em uma sequência de bits que representa sua posição relativa no intervalo, com o número de bits definido pelo parâmetro  $n\_bits$ .
- **Codificação do rótulo (target):** Em ambos os modelos, o rótulo da variável alvo ( $y$ ) foi transformado em valores inteiros utilizando o **LabelEncoder**, garantindo o formato esperado pelos classificadores.

Essas etapas permitiram um pré-processamento consistente entre os dois modelos, mantendo as variáveis categóricas tratadas da mesma maneira e adaptando o tratamento dos atributos numéricos conforme as exigências de cada técnica.

## 3. Implementação das Redes Neurais

### 3.1. Multi-Layer Perceptron (MLP)



A arquitetura do modelo baseia-se em uma *Multi-Layer Perceptron (MLP)* implementada com *PyTorch*. A classe *MLP* define uma rede neural totalmente conectada, com múltiplas camadas ocultas, função de ativação customizável e regularização via *Dropout*.

A classe *MLPClassifierTorch* encapsula a MLP dentro de uma interface compatível com *scikit-learn*, permitindo seu uso em *Pipelines* e integração com métodos de validação e otimização como *RandomizedSearchCV*.

### 3.1.1. Otimização de Hiperparâmetros

A seleção dos hiperparâmetros foi realizada por meio de *RandomizedSearchCV*, com validação cruzada *StratifiedKFold*. O espaço de busca incluiu:

- ***hidden\_sizes***: tamanhos das camadas ocultas, por exemplo: (64,), (128,), (128, 64), (256, 128).
  - ***dropout\_rate***: taxa de dropout, amostrada de uma distribuição contínua entre 0.1 e 0.5.
  - ***learning\_rate***: taxa de aprendizado, amostrada entre 0.0001 e 0.01.
  - ***activation\_fn***: função de ativação (ReLU, LeakyReLU, Tanh).
  - ***weight\_decay***: regularização L2.
  - ***max\_epochs*, *early\_stopping*, *patience*, *verbose***: parâmetros de controle do treinamento.
- 

## 3.2. WiSARD

A arquitetura do modelo baseia-se em um **WiSARD (Wilkie, Stonham and Aleksander Recognition Device)** implementado com o pacote *torchwnn* integrado ao *PyTorch*. A classe *WiSARDClassifierTorch* encapsula o classificador WiSARD em uma interface compatível com *scikit-learn*, permitindo seu uso em *Pipelines* e integração com métodos de validação e otimização como *RandomizedSearchCV*.

O classificador permite a configuração do tamanho das tuplas utilizadas e do mecanismo de bleaching, que contribui para o tratamento de ambiguidades em classificações multiclasse.

### 3.2.1. Otimização de Hiperparâmetros

A seleção dos hiperparâmetros foi realizada por meio de *RandomizedSearchCV*, com validação cruzada *StratifiedKFold*. O espaço de busca incluiu:

- ***tuple\_size***: tamanho das tuplas utilizadas nos discriminadores, por exemplo: 8, 9, 10, 11.
- ***bleaching***: uso ou não do mecanismo de bleaching (*True* ou *False*).

Outros parâmetros do *RandomizedSearchCV* incluíram o número de iterações (*n\_iter*), o número de folds na validação cruzada (*cv*), a métrica de avaliação (*scoring*) e a execução paralela (*n\_jobs=-1*).

---

## 3.2 Avaliação dos Modelos

O melhor modelo identificado pela busca aleatória é avaliado nos conjuntos de treino e teste. As principais métricas utilizadas são:

- **Acurácia** global
- **Relatório de classificação** com precisão, recall e F1-score por classe
- **Matriz de confusão**, visualizada com *seaborn*

A matriz de confusão permite identificar padrões de erro entre classes reais e previstas, sendo útil especialmente em problemas multiclasse.

Exemplo de geração das métricas:

```
from sklearn.metrics import classification_report, confusion_matrix

print(classification_report(y_test, y_pred, target_names=target_names))

sns.heatmap(confusion_matrix(y_test, y_pred),
            annot=True, fmt='d', cmap='Blues',
            xticklabels=target_names, yticklabels=target_names)
```

---

### 3.3 Balanceamento de Classes com SMOTE

Durante a análise dos dados, foi identificado que o *Dataset 2*, o *Dataset 3* e o *Dataset 5* apresentava um desbalanceamento significativo entre as classes da variável alvo. Para resolver esse problema, utilizamos o *SMOTE* (Synthetic Minority Over-sampling Technique, uma técnica de oversampling que cria novas instâncias sintéticas da(s) classe(s) minoritária(s).

---

#### Etapas Realizadas no SMOTE

##### 1. Limpeza dos dados:

- Remoção de instâncias com valores ausentes utilizando `.dropna()`.

##### 2. Separação da variável alvo (y) correspondente às instâncias válidas.

##### 3. Aplicação do SMOTE:

- Utilizado com `random_state=42` para reprodutibilidade.
- Gerou amostras sintéticas para balancear as classes.

```
from imblearn.over_sampling import SMOTE

X_cleaned_df = X_processed_df.dropna()
y_cleaned = y[X_processed_df.notna().all(axis=1)]

smote = SMOTE(random_state=42)
X_resampled, y_resampled = smote.fit_resample(X_cleaned_df, y_cleaned)
```

---

## 4. Resultados:

---

# Dataset 1:

Wizard:

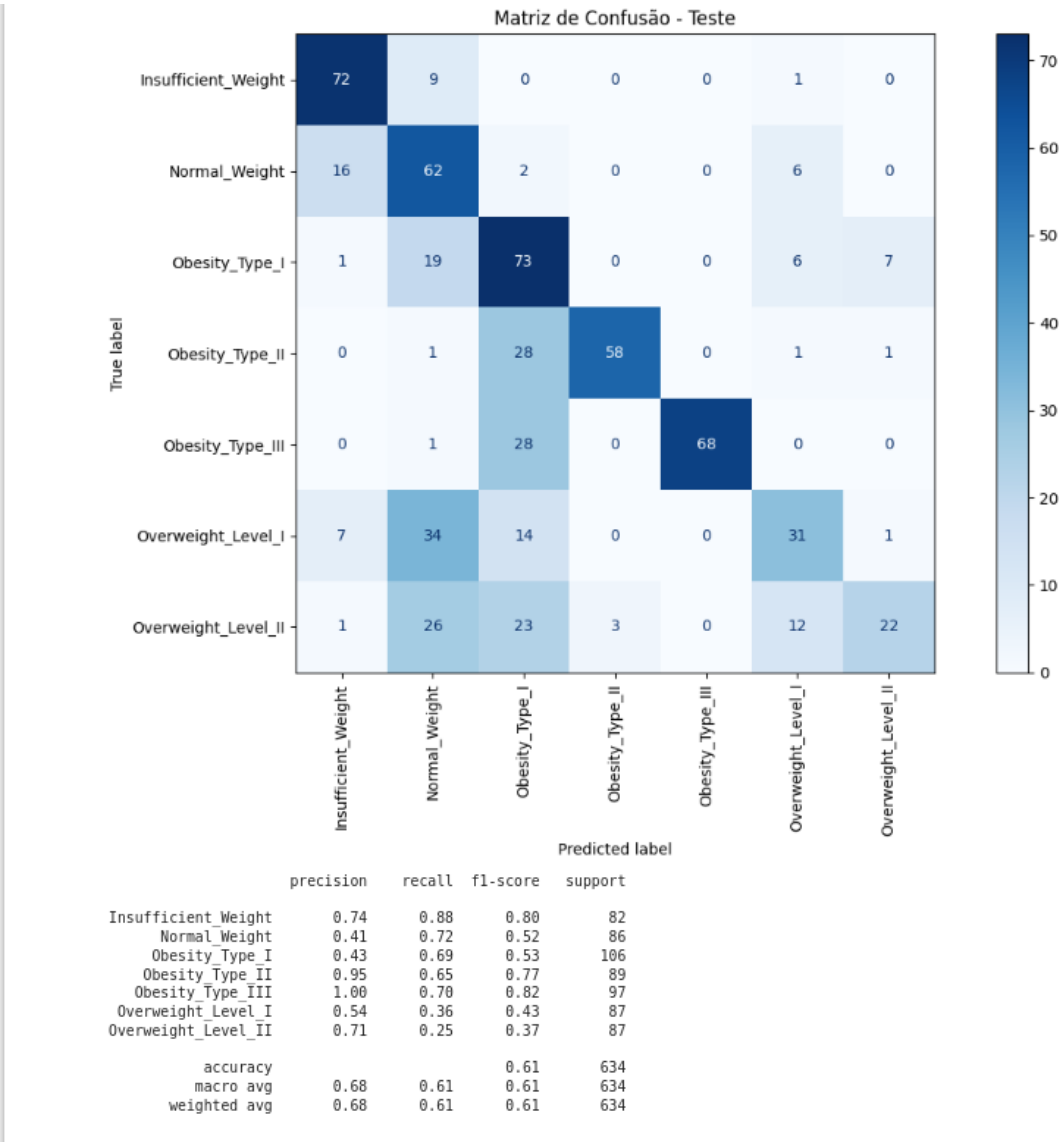
## 1. Métricas de treinamento :

`sklearn.metrics(y_train, y_train_pred, classes, cv)`

Melhores parâmetros encontrados: {'wisard\_\_tuple\_size': 11, 'wisard\_\_bleaching': True}  
Melhor acurácia: 0.660786237679821

|                     | precision | recall | f1-score | support |
|---------------------|-----------|--------|----------|---------|
| Insufficient_Weight | 0.84      | 1.00   | 0.91     | 190     |
| Normal_Weight       | 0.59      | 0.87   | 0.70     | 201     |
| Obesity_Type_I      | 0.53      | 0.89   | 0.67     | 245     |
| Obesity_Type_II     | 0.94      | 0.66   | 0.77     | 208     |
| Obesity_Type_III    | 1.00      | 0.69   | 0.82     | 227     |
| Overweight_Level_I  | 0.80      | 0.61   | 0.69     | 203     |
| Overweight_Level_II | 1.00      | 0.43   | 0.60     | 203     |
| accuracy            |           |        | 0.74     | 1477    |
| macro avg           | 0.81      | 0.74   | 0.74     | 1477    |
| weighted avg        | 0.81      | 0.74   | 0.74     | 1477    |

## 2. Matriz Confusão dos Testes :



MLP:

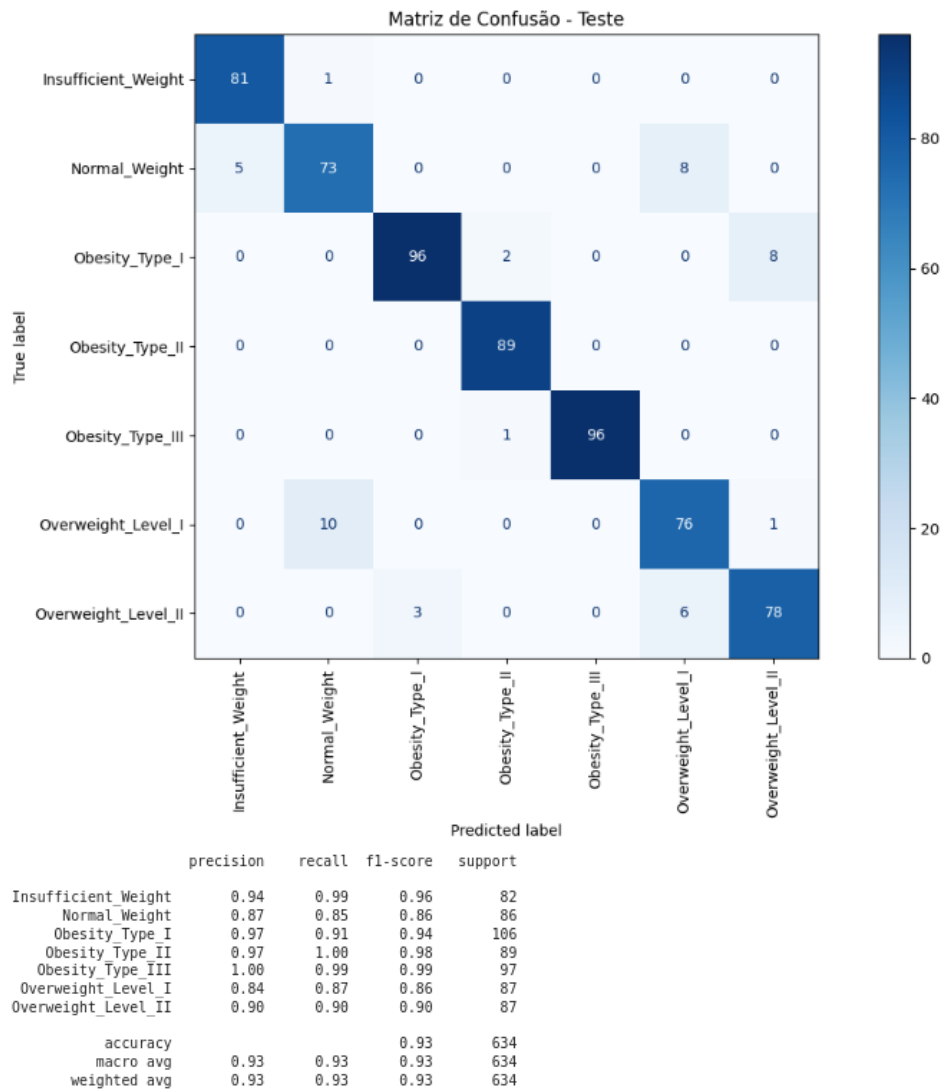
1. Métricas de treinamento :

Melhores parâmetros encontrados: {'mlp\_activation\_fn': <class 'torch.nn.modules.activation.Tanh'>, 'mlp\_dropout\_rate': np.float64(0.24092738738669997), 'mlp\_early\_stopping': True, 'mlp\_hidden\_sizes': (128, 64), 'mlp\_learning\_rate': np.float64(0.007028903586919395), 'mlp\_max\_epochs': 143, 'mlp\_patience': 10, 'mlp\_verbose': False, 'mlp\_weight\_decay': np.float64(0.00629942846779884)}

Acurácia média na CV: 0.9438181401740723

|                     | precision | recall | f1-score | support |
|---------------------|-----------|--------|----------|---------|
| Insufficient_Weight | 0.92      | 0.99   | 0.96     | 190     |
| Normal_Weight       | 0.98      | 0.91   | 0.94     | 201     |
| Obesity_Type_I      | 0.99      | 0.96   | 0.98     | 245     |
| Obesity_Type_II     | 0.96      | 0.99   | 0.98     | 208     |
| Obesity_Type_III    | 1.00      | 1.00   | 1.00     | 227     |
| Overweight_Level_I  | 0.97      | 0.97   | 0.97     | 203     |
| Overweight_Level_II | 0.98      | 0.97   | 0.97     | 203     |
| accuracy            |           |        | 0.97     | 1477    |
| macro avg           | 0.97      | 0.97   | 0.97     | 1477    |
| weighted avg        | 0.97      | 0.97   | 0.97     | 1477    |

2. Matriz Confusão dos Testes :



Dataset 2:

Wizard:

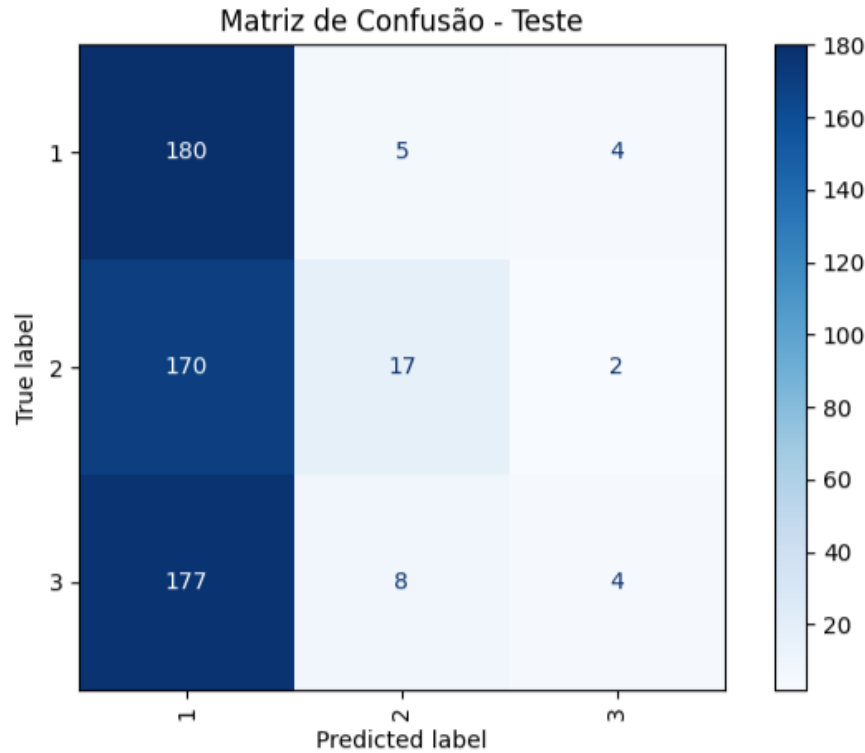
1. Métricas de treinamento :

Melhores parâmetros encontrados: {'wisard\_tuple\_size': 9, 'wisard\_bleaching': False}

Acurácia média na CV: 0.3477272727272727

|              | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 1            | 0.36      | 1.00   | 0.53     | 440     |
| 2            | 0.77      | 0.12   | 0.21     | 440     |
| 3            | 1.00      | 0.06   | 0.12     | 440     |
| accuracy     |           |        | 0.39     | 1320    |
| macro avg    | 0.71      | 0.39   | 0.29     | 1320    |
| weighted avg | 0.71      | 0.39   | 0.29     | 1320    |

2. Matriz Confusão dos Testes :



|              | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 1            | 0.34      | 0.95   | 0.50     | 189     |
| 2            | 0.57      | 0.09   | 0.16     | 189     |
| 3            | 0.40      | 0.02   | 0.04     | 189     |
| accuracy     |           |        | 0.35     | 567     |
| macro avg    | 0.44      | 0.35   | 0.23     | 567     |
| weighted avg | 0.44      | 0.35   | 0.23     | 567     |

MLP:

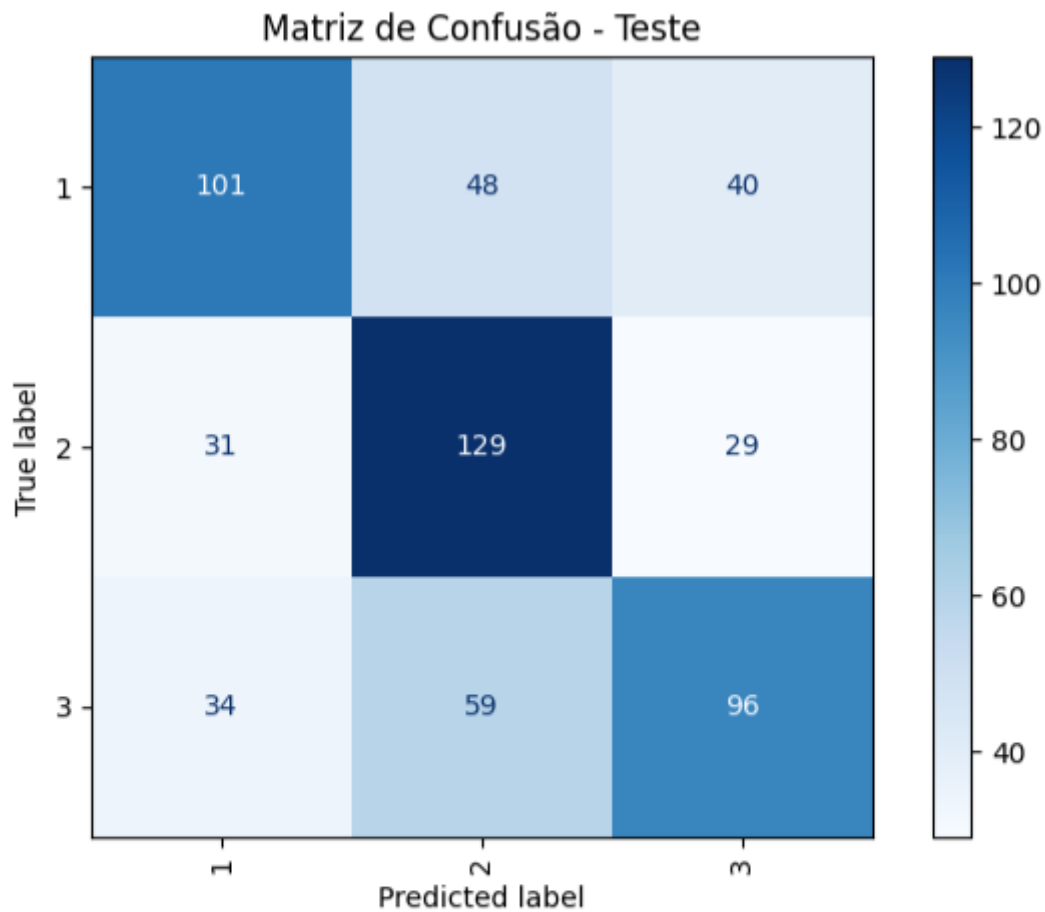
1. Métricas de treinamento :

Melhores parâmetros encontrados: {'mlp\_activation\_fn': <class 'torch.nn.modules.activation.ReLU'>, 'mlp\_dropout\_rate': np.float64(0.12180188587721688), 'mlp\_early\_stopping': True, 'mlp\_hidden\_sizes': (128, 64), 'mlp\_learning\_rate': np.float64(0.009504585843529142), 'mlp\_max\_epochs': 75, 'mlp\_patience': 10, 'mlp\_verbose': False, 'mlp\_weight\_decay': np.float64(0.009158643902204486)}

Acurácia média na CV: 0.5727272727272726

|              | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 1            | 0.73      | 0.60   | 0.66     | 440     |
| 2            | 0.60      | 0.70   | 0.65     | 440     |
| 3            | 0.57      | 0.59   | 0.58     | 440     |
| accuracy     |           |        | 0.63     | 1320    |
| macro avg    | 0.64      | 0.63   | 0.63     | 1320    |
| weighted avg | 0.64      | 0.63   | 0.63     | 1320    |

2. Matriz Confusão dos Testes :



|              | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 1            | 0.61      | 0.53   | 0.57     | 189     |
| 2            | 0.55      | 0.68   | 0.61     | 189     |
| 3            | 0.58      | 0.51   | 0.54     | 189     |
| accuracy     |           |        | 0.57     | 567     |
| macro avg    | 0.58      | 0.57   | 0.57     | 567     |
| weighted avg | 0.58      | 0.57   | 0.57     | 567     |

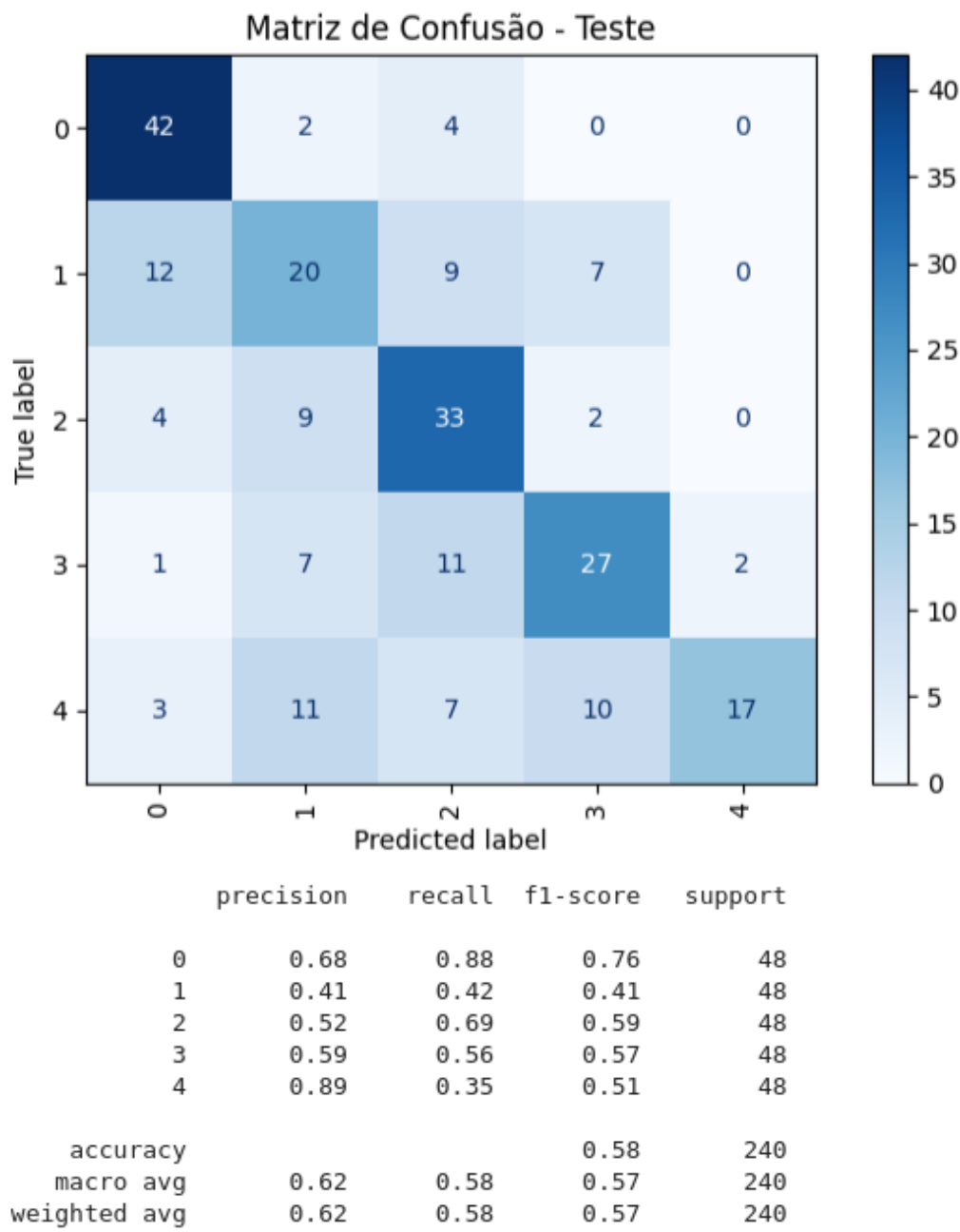
Dataset 3:

Wizard:

1. Métricas de treinamento :

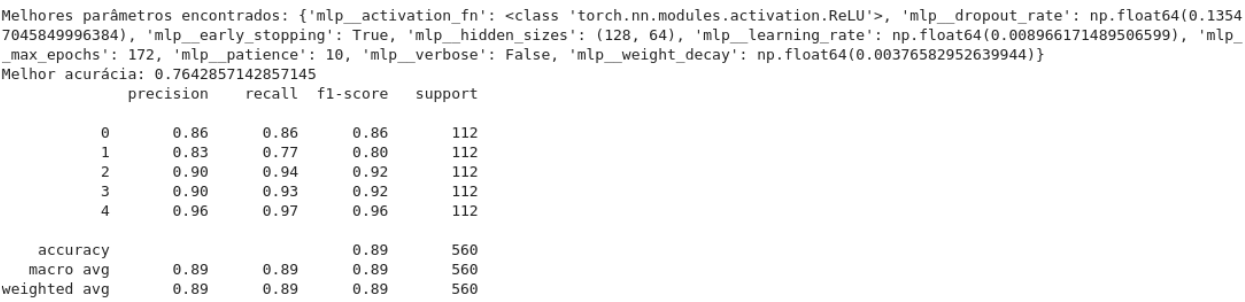
| Melhores parâmetros encontrados: {'wisard_tuple_size': 11, 'wisard_bleaching': False} |           |        |          |         |
|---|-----------|--------|----------|---------|
| Acurácia média na CV: 0.6017671975926245  |           |        |          |         |
|   | precision | recall | f1-score | support |
| 0   | 0.82      | 1.00   | 0.90     | 112     |
| 1   | 0.74      | 0.82   | 0.78     | 112     |
| 2   | 0.81      | 0.90   | 0.86     | 112     |
| 3   | 0.87      | 0.79   | 0.83     | 112     |
| 4   | 1.00      | 0.65   | 0.79     | 112     |
| accuracy  |           |        | 0.83     | 560     |
| macro avg   | 0.85      | 0.83   | 0.83     | 560     |
| weighted avg  | 0.85      | 0.83   | 0.83     | 560     |

2. Matriz Confusão dos Testes :

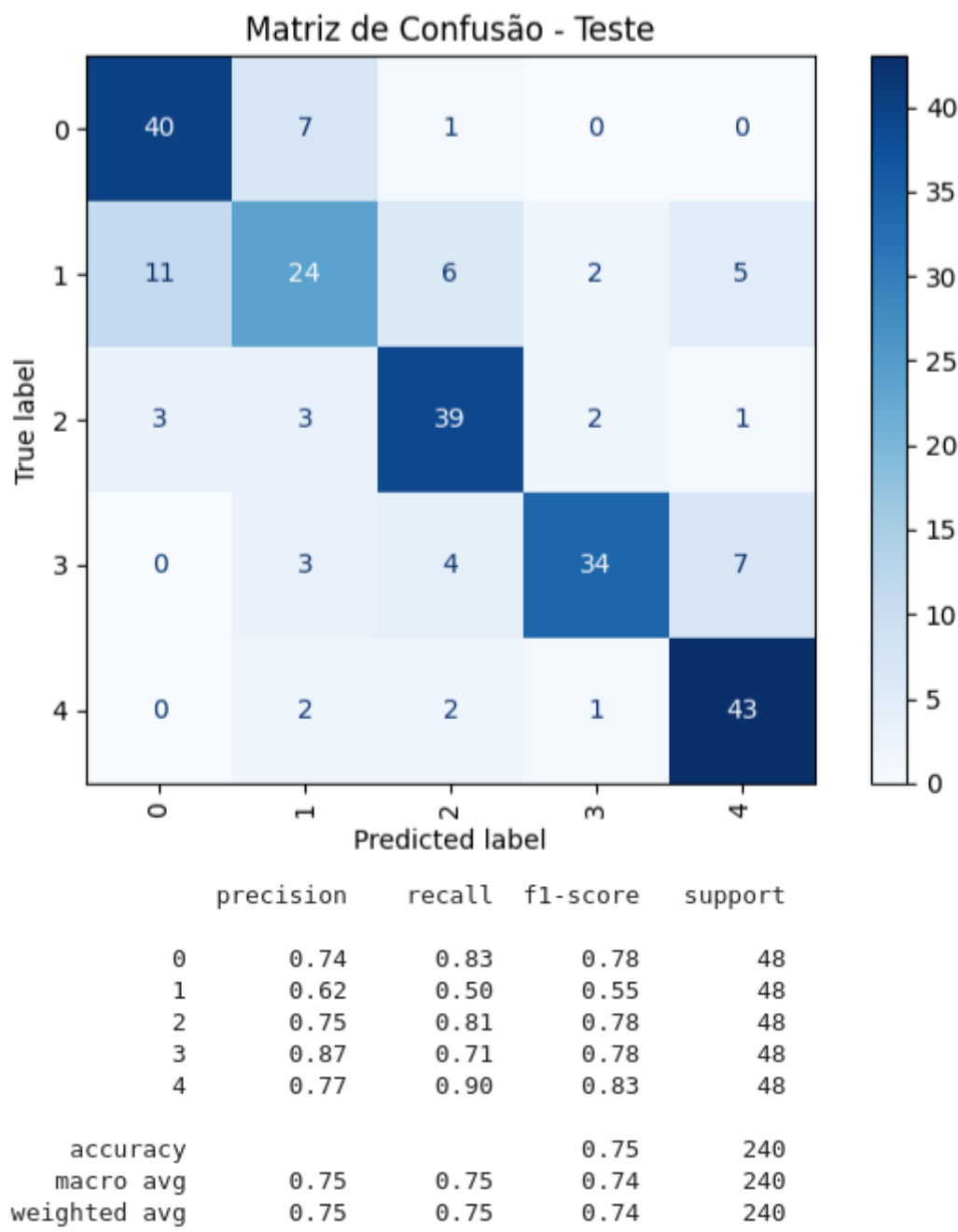


MLP:

1. Métricas de treinamento :



2. Matriz Confusão dos Testes :



Dataset 4:

Wizard:

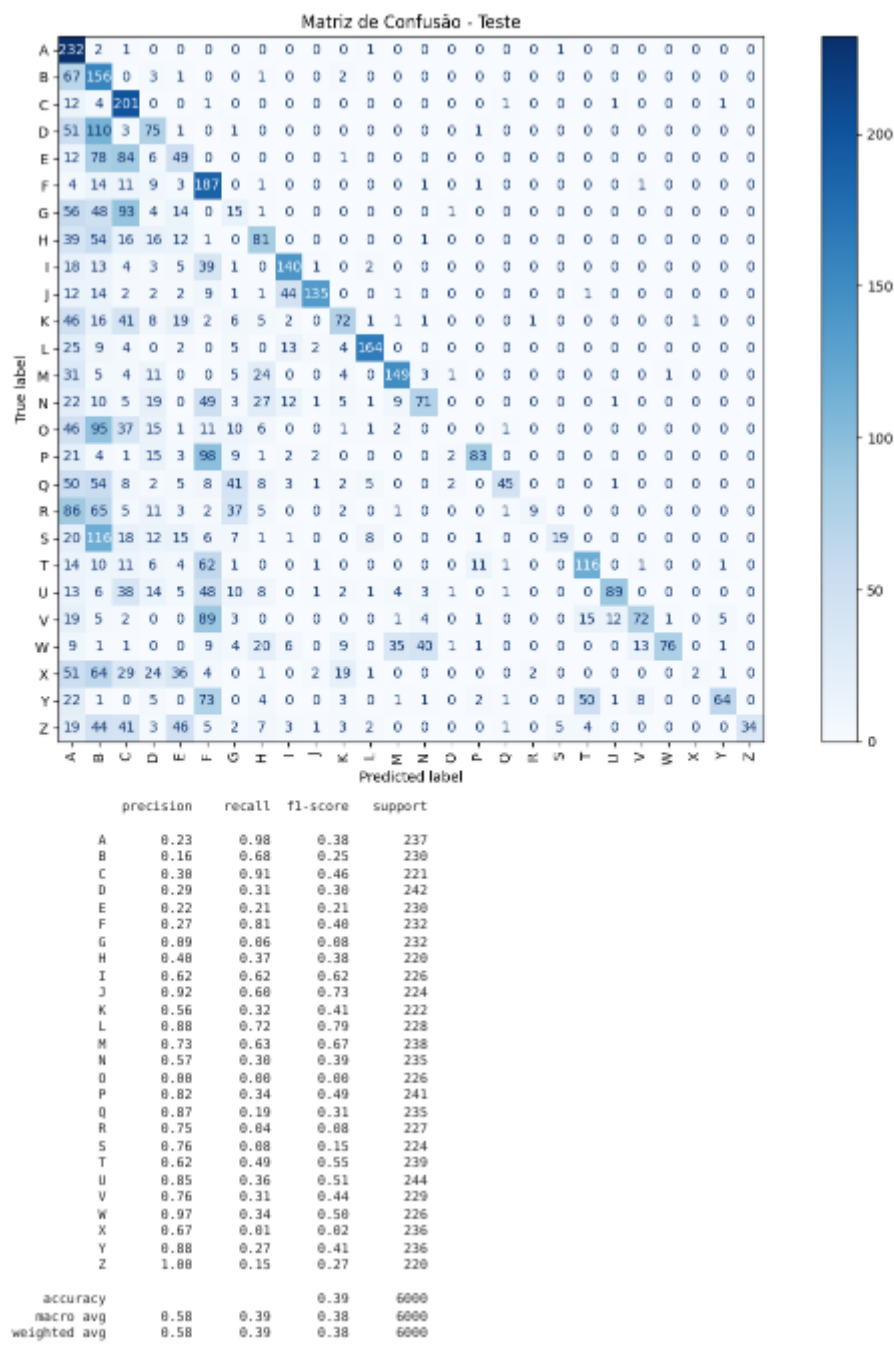


1. Métricas de treinamento :

Melhores parâmetros encontrados: {'wisard\_tuple\_size': 11, 'wisard\_bleaching': True}  
Melhor acurácia: 0.43471431056008397

|              | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| A            | 0.23      | 1.00   | 0.38     | 552     |
| B            | 0.16      | 0.70   | 0.26     | 536     |
| C            | 0.31      | 0.90   | 0.47     | 515     |
| D            | 0.32      | 0.38   | 0.35     | 563     |
| E            | 0.24      | 0.26   | 0.25     | 538     |
| F            | 0.26      | 0.79   | 0.39     | 543     |
| G            | 0.10      | 0.06   | 0.07     | 541     |
| H            | 0.39      | 0.34   | 0.36     | 514     |
| I            | 0.68      | 0.71   | 0.69     | 529     |
| J            | 0.93      | 0.61   | 0.74     | 523     |
| K            | 0.57      | 0.31   | 0.40     | 517     |
| L            | 0.93      | 0.70   | 0.80     | 533     |
| M            | 0.72      | 0.57   | 0.64     | 554     |
| N            | 0.58      | 0.33   | 0.42     | 548     |
| O            | 0.53      | 0.02   | 0.04     | 527     |
| P            | 0.79      | 0.33   | 0.47     | 562     |
| Q            | 0.91      | 0.18   | 0.29     | 548     |
| R            | 0.83      | 0.05   | 0.10     | 531     |
| S            | 0.85      | 0.13   | 0.22     | 524     |
| T            | 0.67      | 0.45   | 0.54     | 557     |
| U            | 0.86      | 0.38   | 0.53     | 569     |
| V            | 0.81      | 0.37   | 0.51     | 535     |
| W            | 1.00      | 0.33   | 0.49     | 526     |
| X            | 0.57      | 0.01   | 0.03     | 551     |
| Y            | 0.99      | 0.28   | 0.43     | 550     |
| Z            | 1.00      | 0.21   | 0.34     | 514     |
| accuracy     |           |        | 0.40     | 14000   |
| macro avg    | 0.62      | 0.40   | 0.39     | 14000   |
| weighted avg | 0.62      | 0.40   | 0.39     | 14000   |

2. Matriz Confusão dos Testes :



MLP:

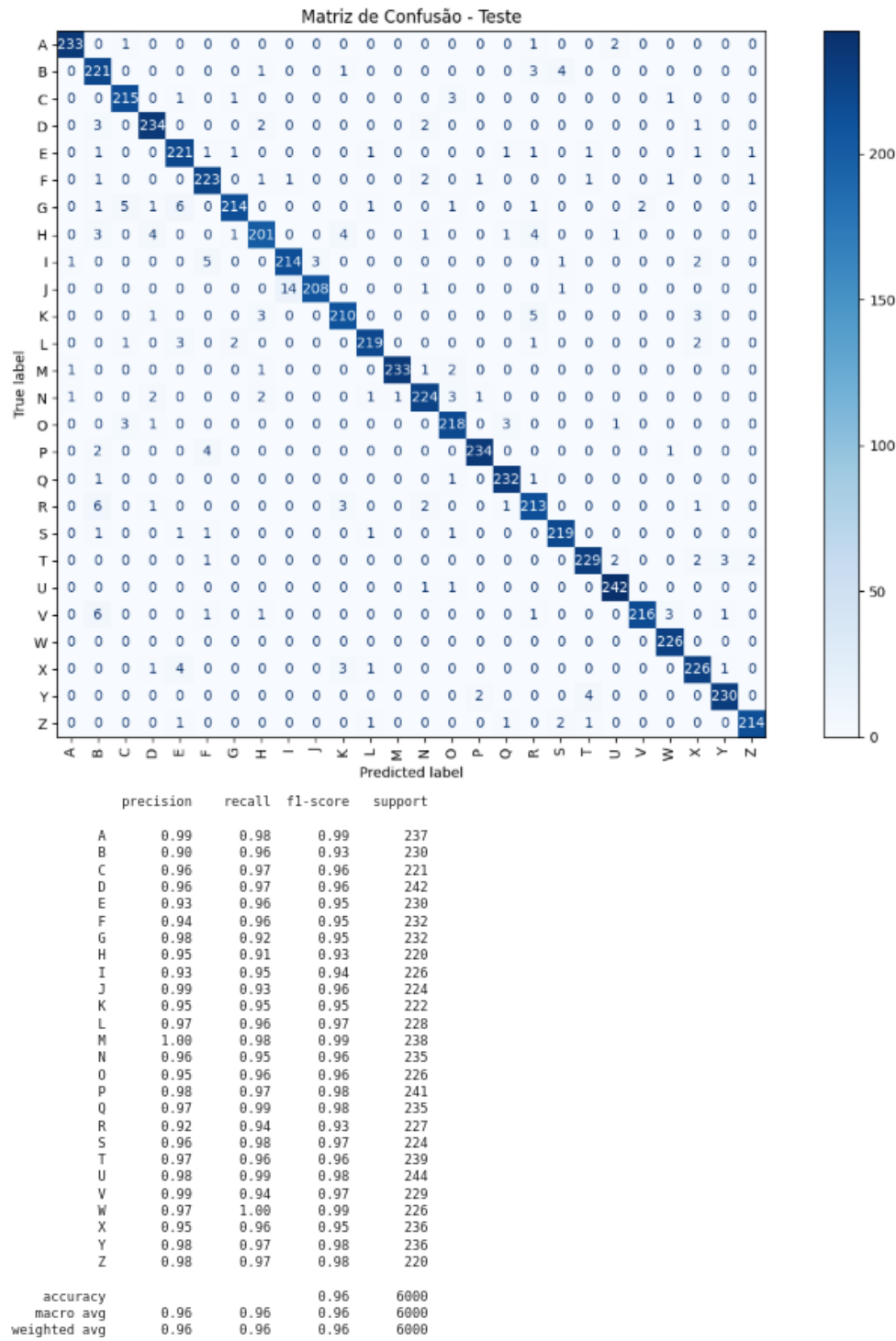
1. Métricas de treinamento :

Melhores parâmetros encontrados: {'mlp\_activation\_fn': <class 'torch.nn.modules.activation.ReLU'>, 'mlp\_dropout\_rate': np.float64(0.156732301008724), 'mlp\_early\_stopping': True, 'mlp\_hidden\_sizes': (256, 128), 'mlp\_learning\_rate': np.float64(0.005044203047025815), 'mlp\_max\_epochs': 252, 'mlp\_patience': 10, 'mlp\_verbose': False, 'mlp\_weight\_decay': np.float64(0.0005687115467735005)}

Melhor acurácia: 0.9485714285714284

|              | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| A            | 1.00      | 0.99   | 0.99     | 552     |
| B            | 0.95      | 0.97   | 0.96     | 536     |
| C            | 0.99      | 0.98   | 0.98     | 515     |
| D            | 0.95      | 0.99   | 0.97     | 563     |
| E            | 0.97      | 0.98   | 0.97     | 538     |
| F            | 0.97      | 0.97   | 0.97     | 543     |
| G            | 0.99      | 0.96   | 0.97     | 541     |
| H            | 0.96      | 0.93   | 0.94     | 514     |
| I            | 0.98      | 0.98   | 0.98     | 529     |
| J            | 0.99      | 0.96   | 0.97     | 523     |
| K            | 0.97      | 0.97   | 0.97     | 517     |
| L            | 0.99      | 0.98   | 0.99     | 533     |
| M            | 0.99      | 0.99   | 0.99     | 554     |
| N            | 0.99      | 0.98   | 0.98     | 548     |
| O            | 0.97      | 0.98   | 0.98     | 527     |
| P            | 0.99      | 0.96   | 0.97     | 562     |
| Q            | 0.97      | 0.99   | 0.98     | 548     |
| R            | 0.94      | 0.96   | 0.95     | 531     |
| S            | 0.98      | 0.99   | 0.99     | 524     |
| T            | 0.98      | 0.99   | 0.98     | 557     |
| U            | 0.99      | 0.99   | 0.99     | 569     |
| V            | 0.99      | 0.97   | 0.98     | 535     |
| W            | 0.98      | 0.99   | 0.99     | 526     |
| X            | 0.99      | 0.99   | 0.99     | 551     |
| Y            | 0.98      | 0.99   | 0.98     | 550     |
| Z            | 0.99      | 0.97   | 0.98     | 514     |
| accuracy     |           |        | 0.98     | 14000   |
| macro avg    | 0.98      | 0.98   | 0.98     | 14000   |
| weighted avg | 0.98      | 0.98   | 0.98     | 14000   |

2. Matriz Confusão dos Testes :



Dataset 5:

Wizard:

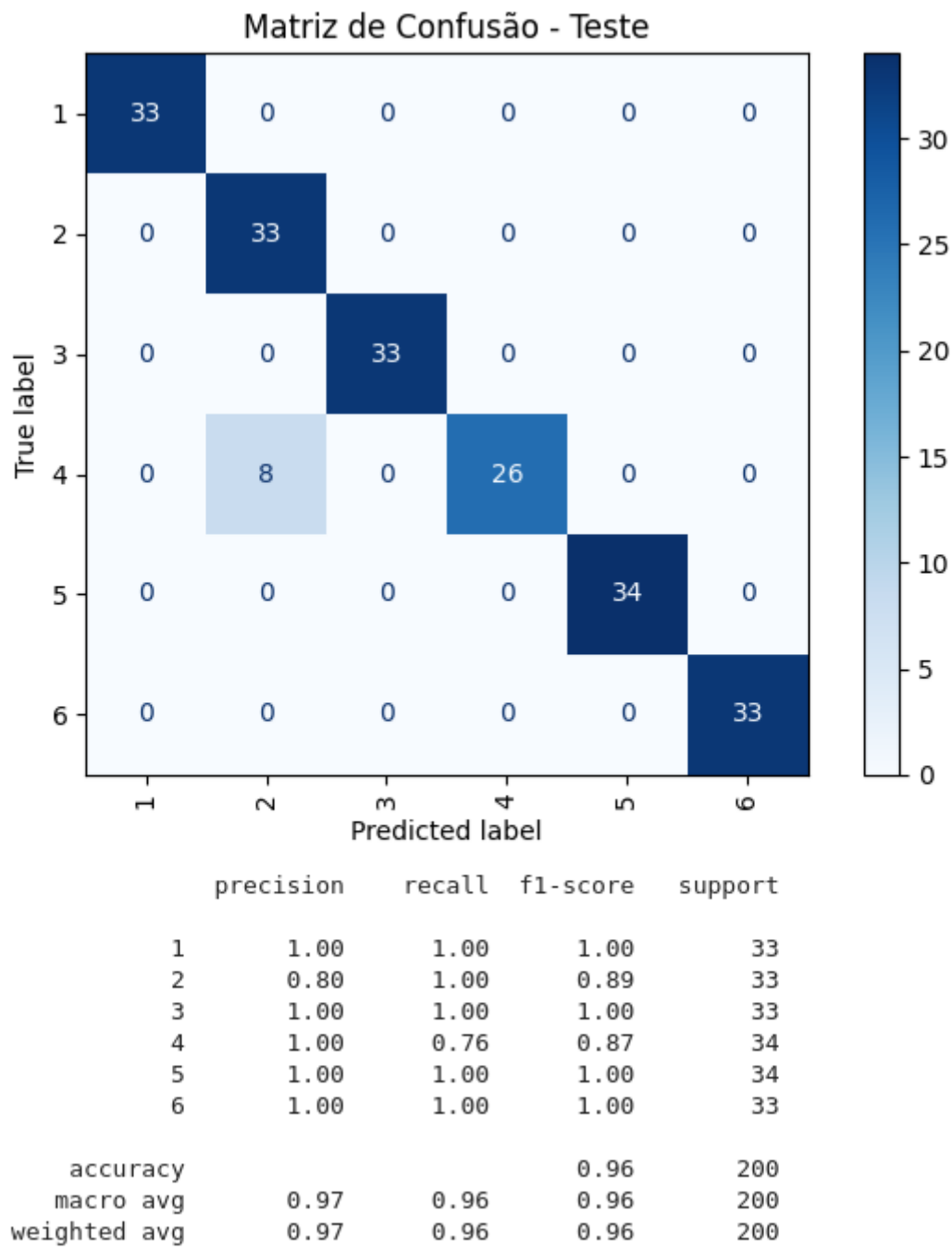
1. Métricas de treinamento :

Melhores parâmetros encontrados: {'wisard\_tuple\_size': 11, 'wisard\_bleaching': True}

Melhor acurácia: 0.9678108629721534

|              | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 1            | 1.00      | 1.00   | 1.00     | 78      |
| 2            | 0.80      | 1.00   | 0.89     | 78      |
| 3            | 1.00      | 1.00   | 1.00     | 78      |
| 4            | 1.00      | 0.74   | 0.85     | 77      |
| 5            | 1.00      | 1.00   | 1.00     | 77      |
| 6            | 1.00      | 1.00   | 1.00     | 78      |
| accuracy     |           |        | 0.96     | 466     |
| macro avg    | 0.97      | 0.96   | 0.96     | 466     |
| weighted avg | 0.97      | 0.96   | 0.96     | 466     |

2. Matriz Confusão dos Testes :



MLP:

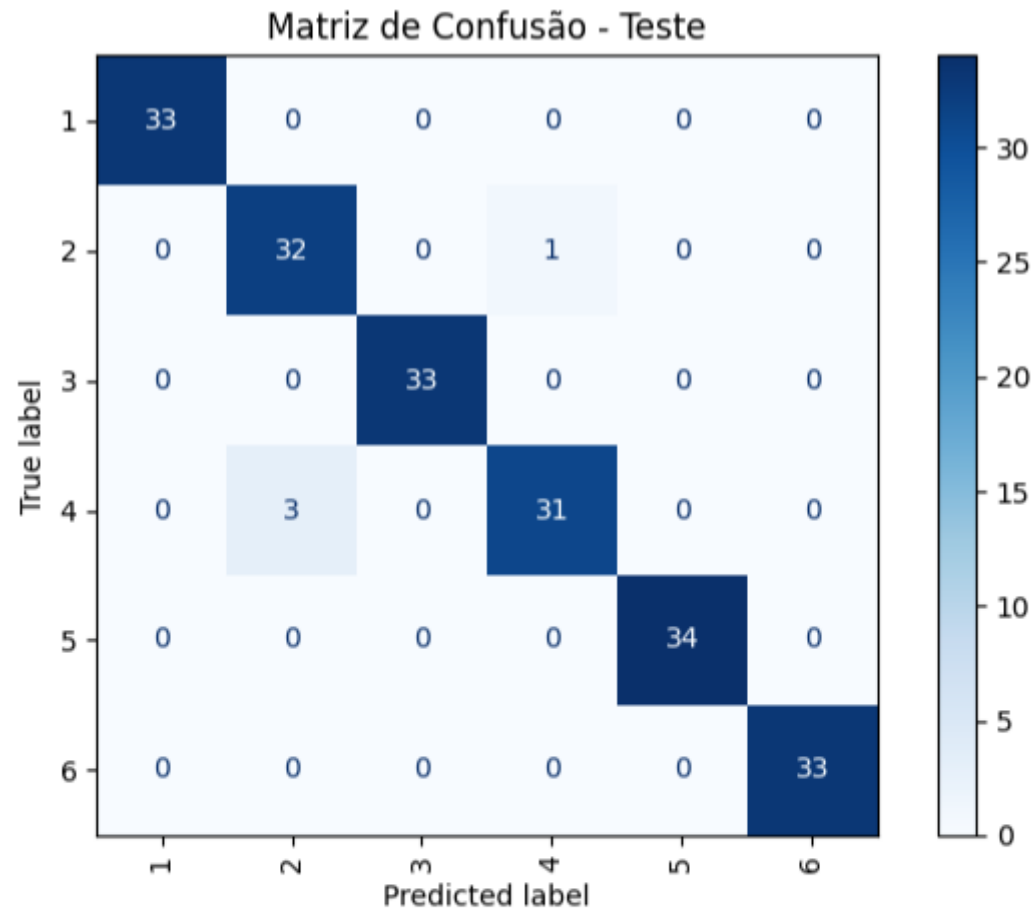
1. Métricas de treinamento :

Melhores parâmetros encontrados: {'mlp\_activation\_fn': <class 'torch.nn.modules.activation.LeakyReLU'>, 'mlp\_dropout\_rate': np.float64(0.19983689107917987), 'mlp\_early\_stopping': True, 'mlp\_hidden\_sizes': (128, 64), 'mlp\_learning\_rate': np.float64(0.006024145688620425), 'mlp\_max\_epochs': 160, 'mlp\_patience': 10, 'mlp\_verbose': False, 'mlp\_weight\_decay': np.float64(0.008609404067363205)}

Melhor acurácia: 0.9914436055822466

|              | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 1            | 1.00      | 1.00   | 1.00     | 78      |
| 2            | 0.99      | 1.00   | 0.99     | 78      |
| 3            | 1.00      | 1.00   | 1.00     | 78      |
| 4            | 1.00      | 0.99   | 0.99     | 77      |
| 5            | 1.00      | 1.00   | 1.00     | 77      |
| 6            | 1.00      | 1.00   | 1.00     | 78      |
| accuracy     |           |        | 1.00     | 466     |
| macro avg    | 1.00      | 1.00   | 1.00     | 466     |
| weighted avg | 1.00      | 1.00   | 1.00     | 466     |

2. Matriz Confusão dos Testes :



|              | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 1            | 1.00      | 1.00   | 1.00     | 33      |
| 2            | 0.91      | 0.97   | 0.94     | 33      |
| 3            | 1.00      | 1.00   | 1.00     | 33      |
| 4            | 0.97      | 0.91   | 0.94     | 34      |
| 5            | 1.00      | 1.00   | 1.00     | 34      |
| 6            | 1.00      | 1.00   | 1.00     | 33      |
| accuracy     |           |        | 0.98     | 200     |
| macro avg    | 0.98      | 0.98   | 0.98     | 200     |
| weighted avg | 0.98      | 0.98   | 0.98     | 200     |