

Estudante: Estevão Augusto da Fonseca Santos

Professor: Ahmed Ali Abdalla Esmin

Métricas do MLPClassifier

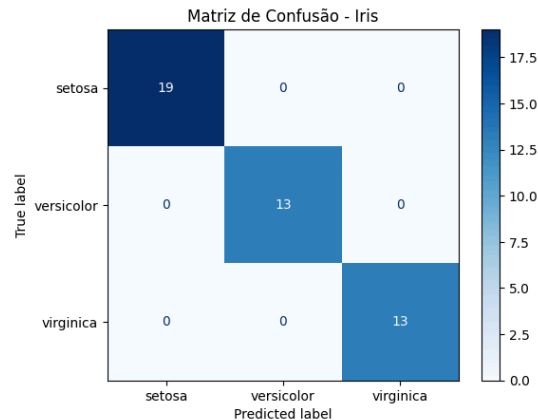
Abaixo vemos os datasets utilizados no projeto, as quais o MLPClassifier demonstrou bons resultados. No Iris tendo classificado todos os registros corretamente, diferente do Wine, que teve um pequeno erro de classificação.

Dataset Iris

```
===== CLASSIFICAÇÃO USANDO MLP - Iris =====
Acurácia: 1.0000
Tempo de Treinamento: 0.2075 segundos
Tempo de Predição: 0.0002 segundos

Relatório de Classificação:
```

	precision	recall	f1-score	support
0	1.000	1.000	1.000	19
1	1.000	1.000	1.000	13
2	1.000	1.000	1.000	13
accuracy			1.000	45
macro avg	1.000	1.000	1.000	45
weighted avg	1.000	1.000	1.000	45

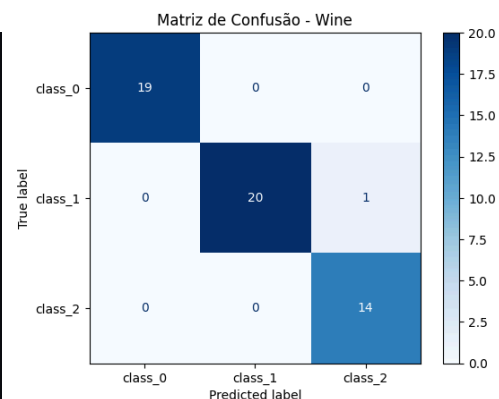


Dataset Wine

```
===== CLASSIFICAÇÃO USANDO MLP - Wine =====
Acurácia: 0.9815
Tempo de Treinamento: 0.2030 segundos
Tempo de Predição: 0.0002 segundos

Relatório de Classificação:
```

	precision	recall	f1-score	support
0	1.000	1.000	1.000	19
1	1.000	0.952	0.976	21
2	0.933	1.000	0.966	14
accuracy			0.981	54
macro avg	0.978	0.984	0.980	54
weighted avg	0.983	0.981	0.982	54



Comparação entre MLPClassifier e KNN

O KNN é um classificador mais simples. Ele não “aprende” ou “treina” de verdade, apenas guarda os dados de treino e classifica uma nova amostra olhando para as mais próximas. O cálculo da distância entre uma amostra e os dados de treinamento é simples e rápido, contudo ineficaz em grandes quantidades de dados.

Já o MLPClassifier (Perceptron Multicamadas), essa é uma IA que realmente aprende. Ela ajusta pesos internos durante o treinamento para reconhecer padrões. Isso faz com que o treino seja mais demorado, mas as previsões sejam mais rápidas. Seu uso é recomendado para problemas complexos com datasets grandes.

Em suma, KNN é melhor para aplicações simples e rápidas, cuja quantidade de dados é menor. Diferentemente do MLPClassifier, que é mais robusto, completo, e busca resolver problemas complexos a partir de grandes datasets.