

Relatório Trabalho Prático I
CI209 -Inteligência Artificial

1. Introdução

Neste trabalho para a disciplina de Inteligência Artificial (CI209), turma 2020-2021, foi implementado o Algoritmo Perceptron para a classificar conjuntos de dados de duas classes. Para os testes foi disponibilizado pela professora as bases de dados MNIST e XOR simplificadas, com menos amostras que em suas versões originais. Ao final da implementação, o classificador perceptron criado teve o seu desempenho comparado com o algoritmo *MLP (Multi Layer Perceptron)* disponível na biblioteca *scikit-learn*. O presente relatório tem como objetivo explicar aspectos da implementação do algoritmo e fornecer uma discussão sobre os resultados obtidos em instâncias de teste.

2. Perceptron

O perceptron é um classificador linear simples que foi implementado para imitar o comportamento do neurônio biológico. Ele é capaz de resolver problemas linearmente separáveis, por meio de um conjunto de pesos. O algoritmo é dividido em duas etapas: treino e teste. Durante a etapa de treino o vetor de pesos do perceptron é atualizado e ajustado de acordo com os dados de treinamento. Enquanto durante a etapa de teste, os pesos são utilizados para determinar a classe de uma entrada. Para a implementação deste trabalho foi utilizado a função abaixo para a predição:

$$p = f(\sum w_i * x_i + w_0)$$

Sendo $f(x) = \{1 \text{ se } x \geq 0; 0 \text{ se } x < 0\}$. Sendo w , o vetor de pesos; x , o dado de entrada e w_0 , o bias.

A atualização dos pesos foi feita utilizando a função abaixo:

$$w_i = w_{old} + e * l * x_i$$

Sendo w_i , o novo valor do peso; w_{old} , o valor do peso antigo; e , o erro da amostra; l , a taxa de aprendizado; e x_i uma característica da amostra sendo calculada.

3.Resultados e Discussão

O perceptron implementado para o trabalho foi testado em duas bases: a base XOR e a MINIST. A base XOR possui 2000 amostras, sendo 1500 utilizadas para treino e as restantes para teste. Enquanto a base MNIST possui 4000 amostras, sendo 3000

utilizadas para treino e o restante para teste. Um gráfico com a distribuição das duas pode ser visto na Figura 1.

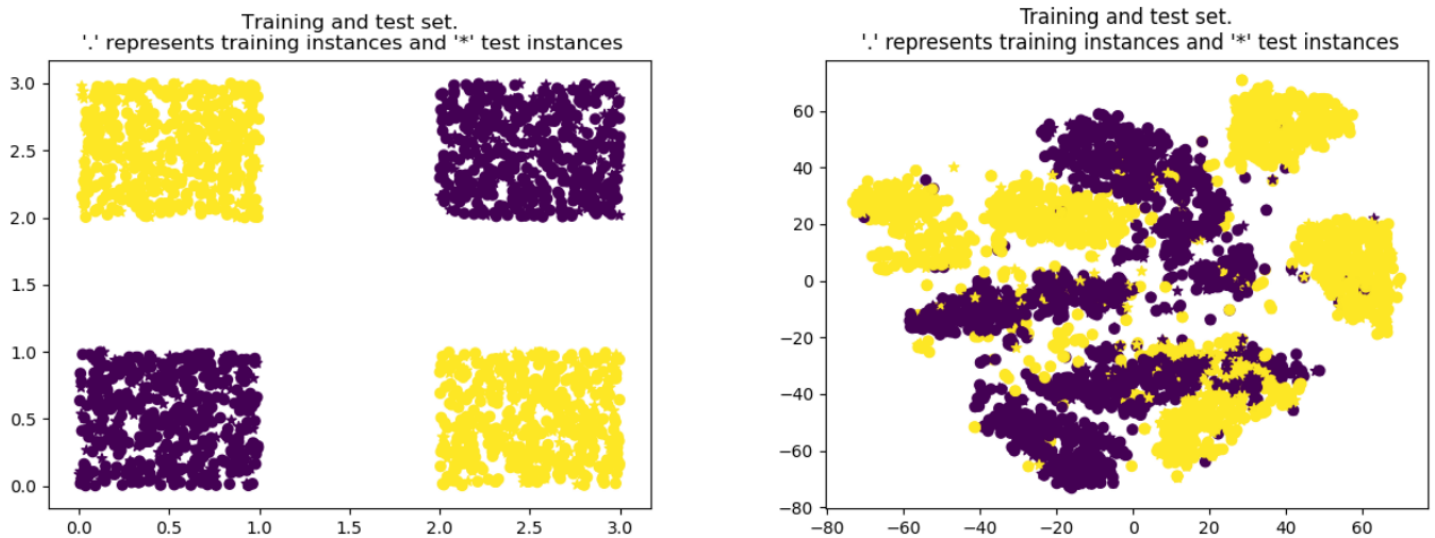


Figura 1: Distribuição dos Datasets usados para treinamento e teste. Em amarelo dados da classe 0 e em roxo dados da classe 1. A esquerda base XOR e a direita base MNIST.

Na Figura 1 é possível compreender a distribuição dos dados e já ter uma ideia de como o perceptron irá se sair em cada uma. A primeira, base XOR, apresenta dados não separáveis linearmente, por mais que tentemos, em um plano 2d não é possível traçar uma reta que separa as classes em quadrantes diferentes. A segunda base apresenta dados misturados mas que podem ser divididos por uma linha capaz de separar as classes.

Essa análise prévia dos dados disponibilizados pode nos dar um diagnóstico prévio de qual classificador se sairá melhor em qual base. Ao testar os classificadores este viés esperado foi confirmado. Para avaliar e comparar o desempenho dos classificadores foi medido a acurácia e a pontuação F1 na etapa de teste. Além disso, também foi utilizado uma matriz de confusão. A Figura 2 apresenta a matriz de confusão de ambos os classificadores para as duas bases e a Tabela 1 apresenta a acurácia e F1 Score dos mesmos para as duas bases.

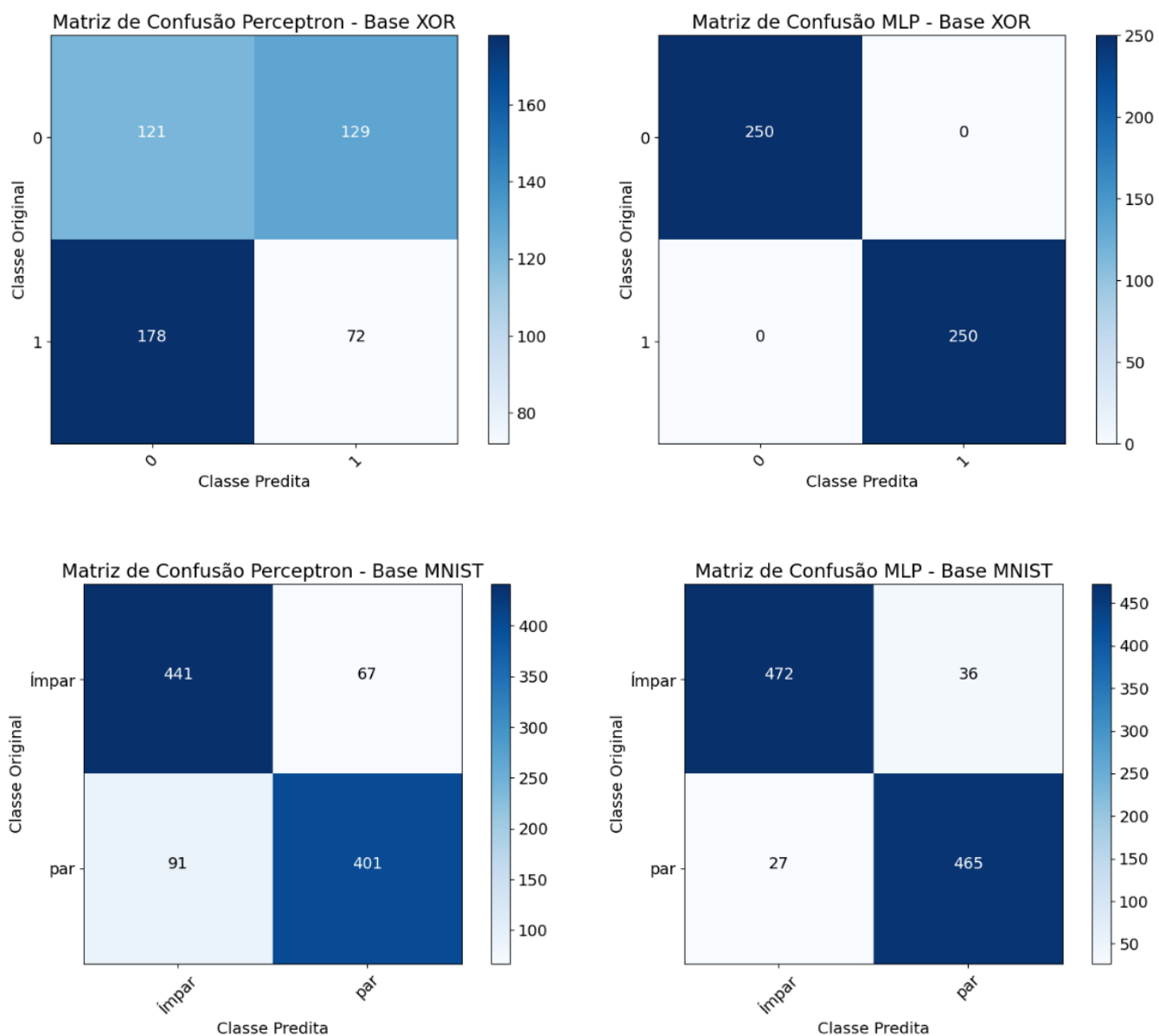


Figura 2: Matriz de confusão dos classificadores obtidos ao treinar e testar a base XOR.

Tabela 1: Acurácia e F1 Score em porcentagem dos classificadores nas duas bases.

Classificador	Base XOR		Base MNIST	
	<u>Acurácia</u>	<u>F1 Score</u>	<u>Acurácia</u>	<u>F1 Score</u>
<i>Perceptron</i>	38,60	31,93	84,20	83,54
<i>MLP</i>	100,0	100,0	93,70	93,66

Durante a etapa de testes foram utilizados os valores default fornecidos pela biblioteca scikit-learn e os parâmetros padrão selecionados para o Perceptron, disponíveis na implementação da biblioteca.

Ao analisar a Figura 2 e a Tabela 1, em ambos os testes o classificador MLP teve desempenho melhor do que o perceptron. Isso se dá principalmente por ser uma rede neural muito mais complexa do que o perceptron, muitas vezes construída a partir de camadas destes classificadores simples. Por mais que os resultados da MLP tenham sido melhores, o Perceptron teve um desempenho muito bom e quase próximo ao atingido pela MLP na base de dados MNIST. Entretanto, seu desempenho ficou muito abaixo do esperado ao analisar a base de dados XOR. Como explicado anteriormente, isso se dá pela distribuição de dados dessa base que não permite com que o Perceptron tenha um bom desempenho.

Podemos concluir com o trabalho que independente do classificador utilizado é sempre proveitoso estudar os dados a serem classificados previamente. Assim se torna possível entender a natureza do problema e escolher o classificador mais adequado para a situação.