

1ª Lista de Exercícios Computacionais - Resolução

Lucas F. P. Costa
Universidade Tecnológica Federal do Paraná
Ponta Grossa, Paraná
luccos@alunos.utfpr.edu.br

I. INTRODUÇÃO

Redes Perceptron e Adaline são algoritmos classificadores e representam cada qual a configuração de uma rede neural artificial. Ambos funcionam como discriminantes lineares, ou seja, conseguem resolver problemas de classificação cujos dados podem ser separados por uma fronteira linear [2].

Neste trabalho, objetivou-se a solução de duas questões. A primeira aborda o comportamento de portas lógicas e a aplicação das redes para sua classificação, e a segunda trata de um problema possivelmente real, sobre o desenvolvimento de uma solução relativa ao campo de estudo da química, no qual se analisa o processo de destilação fracionada do petróleo.

O trabalho está organizado de forma a responder cada questão, fazendo a descrição do problema, desenvolvimento e resultados encontrados. Ao final, as conclusões gerais são sumarizadas.

II. QUESTÃO 1

A. Descrição

Considere as portas lógicas AND, OR e XOR. Monte a tabela verdade de cada uma e faça a classificação das mesmas utilizando como modelos o Adaline e Perceptron. Para cada caso, determine um critério de parada e apresente o erro de treinamento em forma de um gráfico do MSE \times iteração. Discuta se os modelos conseguiram alcançar 100% de acerto e as razões para os resultados. Para cada caso, execute a rede pelo menos 10 vezes [3].

B. Desenvolvimento

As portas lógicas mencionadas podem ser representadas cada qual por uma tabela verdade. As Tabelas I, II e III mostram a construção para cada caso.

Tabela I
FUNÇÃO LÓGICA AND

x_1	x_2	y
0	0	-1
0	1	-1
1	0	-1
1	1	1

Tomando por base as tabelas, pode-se utilizar x_1 e x_2 como dados de entrada para as redes Adaline e Perceptron. Além disso, é necessário adicionar um terceiro valor de entrada relativa ao *bias*, neste caso sempre igual a 1. Para tanto, a entrada pode ser escrita como um vetor de três posições, expresso da seguinte forma:

Tabela II
FUNÇÃO LÓGICA OR

x_1	x_2	y
0	0	-1
0	1	1
1	0	1
1	1	1

Tabela III
FUNÇÃO LÓGICA XOR

x_1	x_2	y
0	0	-1
0	1	1
1	0	1
1	1	-1

$$\mathbf{x} = [x_0, x_1, x_2] = [1, x_1, x_2]. \quad (1)$$

Cada rede retorna como saída a classe relativa à resposta das portas lógicas, porém usando os valores 1 para saída positiva e -1 para negativa. Para fazer o treinamento das redes, também é preciso um vetor com os valores desejados de saída \mathbf{d} , possibilitando o cálculo do erro a cada iteração. O vetor \mathbf{d} tem, portanto, tamanho igual a quantidade de dados de treino e seus valores são as classes de cada entrada. Exemplificando, o vetor \mathbf{d} para a porta lógica AND, como ilustrado na Tabela I, poderia ser representado como:

$$\mathbf{d} = [-1, -1, -1, 1]. \quad (2)$$

C. Resultados

Para realização dos testes, em ambas as redes, a taxa de aprendizado η selecionada é igual a 0,1 e o critério de parada é quando se atinge o total de 50 iterações. Durante a fase de treinamento, os pesos das redes são inicializados com valores aleatórios entre 0 e 1. Cada rede foi executada 10 vezes.

A demonstração dos resultados está dividida primeiramente apresentando a rede Perceptron, seguida da Adaline.

1) *Perceptron*: Os resultados obtidos durante o treinamento do Perceptron para cada uma das três portas lógicas é ilustrado na Figura 1, que exibe o Erro Quadrático Médio (MSE, do inglês *Mean Square Error*) por iteração.

É possível observar que para os casos AND e OR, foi possível alcançar um valor de erro igual a 0, ou seja, para os

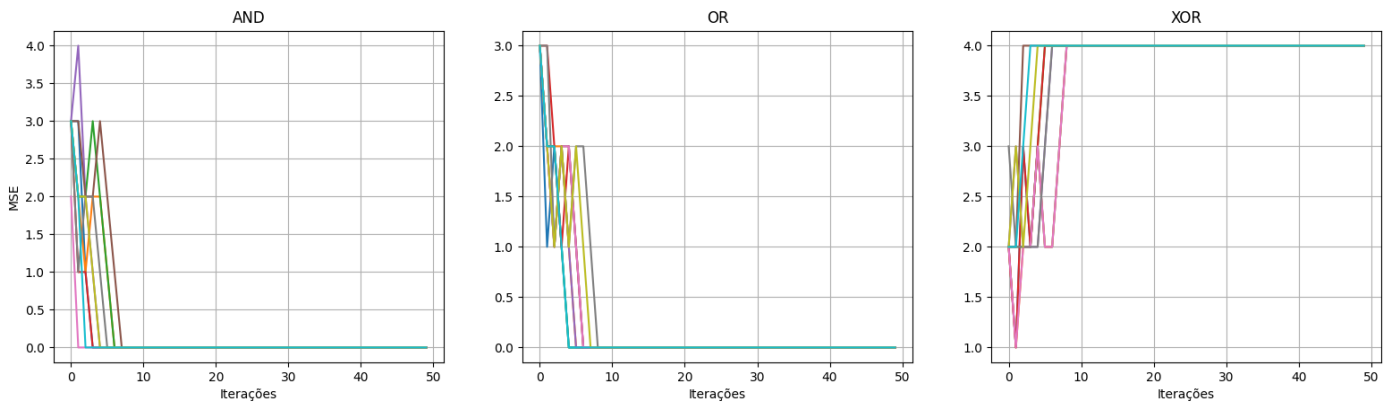


Figura 1. Erro Quadrático Médio para cada iteração durante o treinamento da rede Perceptron.

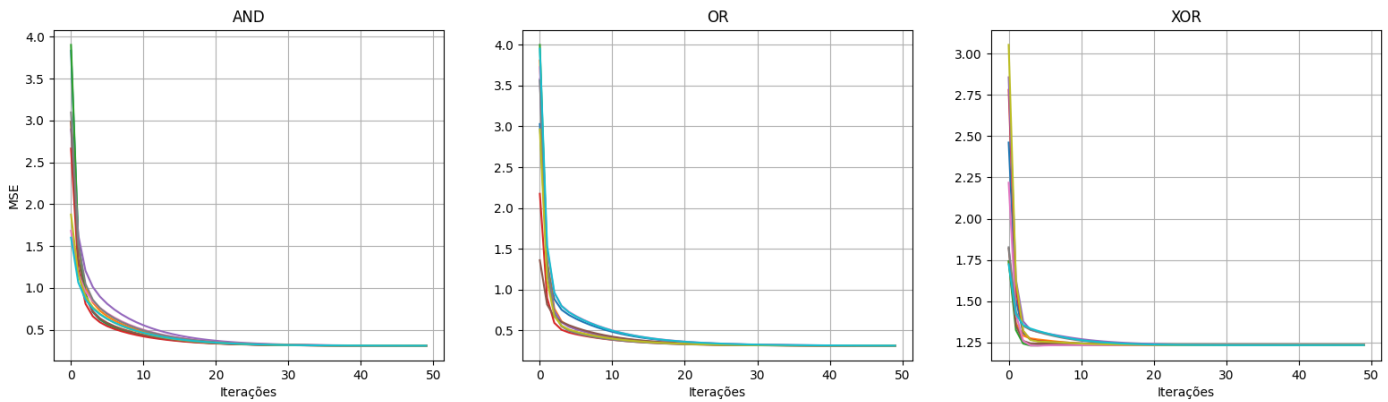


Figura 2. Erro Quadrático Médio em cada iteração no treino da rede Adaline.

dados apresentados, a rede é capaz de classificar corretamente com 100% de certeza. O mesmo não acontece para o caso XOR, onde o erro cresce até estabilizar com valor igual a 4, o que é esperado, já que a rede Perceptron não é capaz de fazer essa classificação, pois este caso não é linearmente separável.

Interessante notar que para alguns testes dos casos AND e OR houve aumento do erro entre uma iteração e outra, logo no início. Isso pode ser explicado pelo valor escolhido de η possivelmente alto, o que fez com que os passos dados para encontrar a solução foram muito grandes, sendo que esse comportamento é mais evidente no caso AND.

2) *Adaline*: A Figura 2 demonstra o MSE para cada iteração dos casos AND, OR e XOR para o treino da rede Adaline.

Percebe-se a característica marcante de diferença entre essa rede e a Perceptron na curva mais suave em todos os casos, resultado do fato de que neste método, para correção dos pesos, usa-se o erro real em cada iteração, antes de passar pela função de ativação, o que resulta em valores de erro reais e não inteiros.

Também é possível verificar que, para todos os casos, houve melhoria do erro inicial com tendência de redução e estabilização. Os valores finais (da última iteração) de MSE para cada caso estão na Tabela IV. Embora para o caso XOR também tenha ocorrido redução, o valor de erro é muito alto,

levando a uma classificação incorreta de entradas, assim como esperado.

Tabela IV
FUNÇÃO LÓGICA XOR

Porta lógica	MSE
AND	0,3086
OR	0,3093
XOR	1,2340

Diferentemente do observado na rede Perceptron, o comportamento da Adaline não indica problemas de passos muito longos na tentativa de encontro da solução, o que leva a crer que η escolhida para este teste foi adequado. Além disso, ao final das iterações, todas as curvas se encontram no mesmo ponto, o que significa que a quantidade de iterações está correta. Porém, não foi possível alcançar valor de erro igual a 0 para os casos AND e OR, devido a forma padrão de operação da rede em questão.

III. QUESTÃO 2

A. Descrição

Resolva computacionalmente utilizando Perceptron e Adaline as questões do projeto prático 3.6 (pág. 70) do livro de Silva, Spatti e Flausino

Tabela V
RESULTADOS DO TREINAMENTO DO PERCEPTRON

Treinamento	Vetor de pesos iniciais				Vetor de pesos finais				Iterações
	w_0	w_1	w_2	w_3	w_0	w_1	w_2	w_3	
1	0,2719	0,3010	0,4383	0,5786	-3,0681	1,5550	2,4642	-0,7312	398
2	0,6772	0,7391	0,5919	0,9289	-3,0228	1,5417	2,4526	-0,7234	404
3	0,9483	0,2068	0,4362	0,7898	-3,1317	1,5660	2,5024	-0,7442	458
4	0,1317	0,4727	0,3077	0,6746	-3,1083	1,5818	2,5054	-0,7411	416
5	0,6484	0,9774	0,1399	0,5380	-3,0916	1,5719	2,4925	-0,7384	421

Tabela VI
RESULTADOS DO TREINAMENTO DO PERCEPTRON

Treinamento	Vetor de pesos iniciais				Vetor de pesos finais				Iterações
	w_0	w_1	w_2	w_3	w_0	w_1	w_2	w_3	
1	0,7587	0,8219	0,5760	0,7094	-2,3271	1,4296	2,3509	-0,6223	243
2	0,3931	0,8239	0,8742	0,1404	-2,3255	1,4289	2,3506	-0,6220	229
3	0,1702	0,8821	0,0080	0,7784	-2,3280	1,4297	2,3500	-0,6224	230
4	0,8601	0,9591	0,6032	0,6451	-2,3280	1,4300	2,3510	-0,6225	244
5	0,4102	0,4841	0,3789	0,4717	-2,3250	1,4287	2,3503	-0,6219	237

"Redes Neurais Artificiais para engenharia e ciências aplicadas", o qual trata do processo de destilação fracionada do petróleo. Preencha as Tabelas 3.2 e 3.3 e justifique as respostas [3].

B. Desenvolvimento

O problema abordado na referência da descrição da questão é constituído de três grandezas de entrada que classificam um determinado óleo em duas classes de pureza diferentes [1]. Tanto Perceptron como Adaline podem ser empregados para solucionar este problema.

Um ponto importante é que, para a entrada do *bias*, deve-se considerar x_0 como igual a -1. Também fica definido que η deve ser de 0,01 e o critério de parada quando o erro for 0 para o Perceptron. Como o problema não aborda a rede Adaline, e como esta geralmente não converge para valores de erro 0, foi arbitrado para este estudo que o critério de parada será quando o erro for menor ou igual a 0,4.

Outra configuração diz respeito ao vetor de pesos, que deve iniciar com valores aleatórios entre 0 e 1. São disponibilizados valores para treinamento, com as três entradas e valores desejados de saída.

O primeiro passo do projeto é executar cinco treinamentos para cada rede e registrar o vetor de pesos iniciais e finais, bem como o número de iterações executadas. Depois, com os pesos salvos, deve-se fazer a classificação para 10 amostras diferentes. Por fim, responder a dois questionamentos:

- Por que o número de iterações varia a cada execução?
- É possível dizer que as classes são linearmente separáveis?

C. Resultados

A Tabela V mostra os resultados dos treinamentos do Perceptron, e a Tabela VI do Adaline.

Para todos os treinamentos, tanto para a rede Perceptron como Adaline, percebe-se que mesmo com os pesos iniciando-se sempre em valores diferentes, os valores dos pesos finais são

similares em cada coluna, sendo que o desvio padrão médio dos pesos finais do Perceptron é igual a 0,0223, e 0,0006 para Adaline.

A média de iterações para que o Perceptron alcance o critério de parada é de 419,4, significativamente maior do que Adaline, que leva em média 236,6 iterações. Isso pode ser explicado pelo fato de que o erro considerado para parada do Adaline é maior. No entanto, por meio de testes, verificou-se necessário pois o algoritmo não atinge valores menores. Aqui, já é possível responder ao primeiro questionamento, sobre a variação do número de iterações a cada execução, que se traduz no fato de que, para cada treinamento, o vetor inicial de pesos é sempre diferente, o que leva mais ou menos iterações até que as redes encontrem os valores ideais.

O resultado da classificação de 10 amostras usando os cinco conjuntos de pesos calculados para cada rede é apresentado na Tabela VII.

Tabela VII
RESULTADOS DA VALIDAÇÃO DAS REDES PERCEPTRON E ADALINE.

Amostra	Perceptron					Adaline				
	y_1	y_2	y_3	y_4	y_5	y_1	y_2	y_3	y_4	y_5
1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
2	1	1	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1	1	1	1
5	1	1	1	1	1	1	1	1	1	1
6	1	1	1	1	1	1	1	1	1	1
7	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
8	1	1	1	1	1	1	1	1	1	1
9	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
10	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1

Para todas as amostras, independente do peso e rede, as respostas são sempre as mesmas, o que leva ao entendimento de que ambas as redes foram suficientemente treinadas, o critério de parada escolhido foi adequado, bem como o valor

de η escolhido. Além disso, pode-se concluir seguramente que as classes são linearmente separáveis, como resposta ao último questionamento do projeto.

IV. CONCLUSÃO

Foram apresentados dois problemas resolvidos computacionalmente fazendo uso de redes Perceptron e Adaline, o primeiro relativo ao comportamento de portas lógicas e o segundo a uma aplicação de mundo real, um processo de destilação fracionada de petróleo. Para cada problema, as redes tiveram de ser adaptadas de acordo com a quantidade de entradas e configurações específicas.

Foi possível observar que as redes tem potencial de solução desde problemas simples de classificação a outros mais complexos, para casos onde existe uma divisão linear entre classes, reduzindo consideravelmente o tempo que levaria um ser humano frente a mesma situação.

O poder destas simples redes pode ser grandemente expandido fazendo uso de algoritmos mais robustos, como acontece com MLPs. Ademais, os experimentos contribuem para entendimento didático e prático sobre o funcionamento de estruturas tão importantes no ambiente computacional.

REFERÊNCIAS

- [1] SILVA, I. N. da; SPATTI, D.; FLAUZINO, R. **Redes neurais artificiais para engenharia e ciências aplicadas**: curso prático. 2 ed. São Paulo: Artliber Editora Ltda, 2010.
- [2] SIQUEIRA, H. V. **Aula 8**: perceptron e adaline. Ponta Grossa: Universidade Tecnológica Federal do Paraná, 2021. 47 slides, colorido.
- [3] SIQUEIRA, H. V. **Lista 1**. Disponível em: <http://paginapessoal.utfpr.edu.br/hugosiqueira/disciplinas-mestrado/redes-neurais-artificiais/Lista%201%20-%20Redes%20Neurais.pdf>. Acesso em: 12 mar. 2021.