

**Relatório de Atividades**  
**Computação Inspirada pela Natureza**

## **TRABALHO 2: REDES NEURAIIS**

Davi Augusto Neves Leite

Professor: Fabricio Aparecido Breve

Departamento de Computação, Faculdade de Ciências, Universidade  
Estadual Paulista Júlio de Mesquita Filho

Maio/2023

# 1 Introdução

O Perceptron é um modelo de aprendizado de máquina que tem como objetivo classificar dados em duas ou mais classes, a partir de um conjunto de dados de entrada linearmente separáveis. Inspirado no funcionamento do neurônio biológico, este modelo basicamente consiste em traçar equações lineares para separar, quando possível, o conjunto de dados de entrada, para os quais já se sabe a quantidade de classes e, portanto, trata-se de um modelo supervisionado. Para que isso seja possível, o processo de treinamento é baseado no ajuste dos chamados pesos e *bias* (parâmetro que move a reta do eixo), associados às entradas, para a minimização do erro de classificação. Devido à sua simplicidade e eficiência, o Perceptron é amplamente utilizado em tarefas de classificação binária, como reconhecimento de padrões e detecção de spam, e nas áreas de processamento de imagens e análise de dados.

Diante disso, o presente relatório tem por objetivo investigar o desempenho do Perceptron em diferentes configurações e conjunto de dados, além de compará-lo com sua principal variação denominada Perceptron de Múltiplas Camadas (*Multi Layer Perceptron* - MLP). Para tanto, foram realizados três exercícios diferentes: o primeiro baseado no famoso *Iris Data Set*, o segundo baseado no *Wine Data Set*, e o terceiro baseado no *Blood Transfusion Service Center Data Set*. Para cada exercício, foram testadas diferentes configurações do Perceptron, como valores dos pesos, valores do *bias* e taxa de aprendizado (*learning rate* - LR). Além disso, no terceiro exercício, os resultados do Perceptron foram comparados, nas mesmas condições experimentais, diretamente com o MLP.

## 1.1 Organização do Trabalho

O presente relatório está estruturado da seguinte forma: na Seção 2 tem-se a descrição dos experimentos gerais propostos e materiais utilizados. Na Seção 3, são descritos os experimentos específicos de cada exercício, apresentados e discutidos os resultados obtidos. Por fim, na Seção 4 são apresentadas as considerações finais gerais considerando o contexto de aplicação do Perceptron em cada exercício.

## 2 Materiais e Métodos

A fim de medir-se a acurácia do Perceptron para os problemas propostos no enunciado do trabalho, foi realizada uma avaliação quantitativa baseada em duas partes. A primeira, considerando o melhor Perceptron executado de todas as configurações definidas, com os valores iniciais e finais dos pesos e *bias*, a LR, as matrizes de confusão dos subconjuntos de treinamento, validação e teste, e o gráfico de convergência do erro médio quadrático (MSE). Considera-se, neste contexto, o melhor Perceptron executado como aquele que obteve a melhor acurácia para o subconjunto de teste. Já a segunda com média, desvio padrão, mínimo, mediana e máximo para a acurácia, o número de épocas e o tempo de execução (em segundos), novamente considerando os os valores iniciais e finais dos pesos e *bias* e da LR. Por fim, os experimentos propostos estão descritos em na Seção 3, tendo os resultados descritos em capítulos separados para cada exercício.

### 2.1 Materiais

Os principais recursos necessários para a execução do trabalho podem ser vistos a seguir.

#### 1. Software

- Sistemas Operacionais: Windows 11 para *desktop*;
- Ambiente de Desenvolvimento Integrado: Microsoft Visual Studio Code;
- Linguagem de Programação: Python 3.11.3 64-bit.

Pacotes Extras Não-Nativos: NumPy, Matplotlib e SciKit-Learn.

#### 2. Hardware

- Notebook pessoal Lenovo Ideapad 330-15IKB com: processador Intel Core i7-8550U, HDD WD Blue WD10SPZX de 1TB, SSD Crucial BX500 de 240GB, 12 GB DDR4 de Memória RAM e placa de vídeo NVIDIA GeForce MX150 (2 GB GDDR5 de memória).

### 3 Experimentos e Resultados

O objetivo deste capítulo é apresentar os experimentos e resultados obtidos para cada um dos exercícios. Especificamente, o exercício 01 será detalhado na Subseção 3.1, o exercício 02 na Subseção 3.2 e, por fim, o exercício 03 na Subseção 3.3.

De modo comum, a metodologia adotada para a realização e avaliação dos experimentos consistiu em uma série de restrições e variações de parâmetros do Perceptron para um conjunto de dados específico de cada exercício. Foram realizados **quatro ciclos de execução**, em sequência, onde os valores de pesos ( $W$ ) e *bias* iniciais, para uma LR pré-estabelecida, foram gerados aleatoriamente e fixados para um ciclo de execução. Em outras palavras, cada LR possui os valores de  $W$  e *bias* aleatórios, mas estes valores são repetidos para as taxas de aprendizado consecutivas. Cada ciclo teve o algoritmo sendo executado **25 vezes para cada LR pré-estabelecida**, de tal forma a garantir uma amostra estatisticamente significativa para as tabelas. Também, optou-se pela aplicação do *zscore* (normalização prévia do *data set*) e pela divisão, em cada exercício, do conjunto de dados original nos seguintes três subconjuntos: **treinamento, com 70%; validação, com 15%; e teste, com 15%.**

Desta forma, os parâmetros adotados para a configuração do Perceptron (e MLP) foram os seguintes:  $W$  e *bias* iniciais variáveis por ciclo, com valores dentro do intervalo  $[0; 1]$ ; número máximo de épocas (ou iterações) permitido de 1000; número máximo de "paciência" para a estagnação do algoritmo de 10. Ademais, foram pré-estabelecidos quatro diferentes taxas de aprendizado: 1, 0.1, 0.01 e 0.001. Cada taxa, conforme explicado, foi utilizada em um ciclo de execução. Considerou-se como sendo o melhor Perceptron, conforme já dito, aquele cuja acurácia foi maior dentre todos os ciclos de execução.

Por fim, foram gerados o gráfico de convergência do MSE de treinamento e validação e as matrizes de confusão a partir do melhor Perceptron. É importante lembrar que as matrizes de confusão apresentam um resumo da performance do modelo de classificação com relação às classes reais, sendo uma ferramenta útil para avaliar a qualidade das previsões e identificar possíveis erros ou limitações. Além disso, as tabelas de acurácia, número de épocas e tempo de execução (em segundos) foram obtidas com relação aos dados totais dos experimentos realizados.

### 3.1 Exercício 1

O primeiro exercício tinha por objetivo reconhecer as classes advindas do *Iris Data Set*<sup>1</sup> por meio do Perceptron. De modo geral, este conjunto de dados é composto por **150 amostras** de **três espécies de flores** do gênero *Iris*: *Iris Setosa*, *Iris Versicolour* e *Iris Virginica*. Cada espécie (ou classe) possui **50 amostras**, com **quatro características numéricas** associadas a cada uma delas: comprimento e largura da sépala (em cm) e comprimento e largura da pétala (em cm). É importante ressaltar que as classes são previamente conhecidas e linearmente separáveis uma em relação às outras, o que permite a aplicação do Perceptron de **forma individual para cada classe**. Por fim, na Subsubseção 3.1.1 é possível visualizar as informações e resultados do melhor Perceptron obtido durante os experimentos e na Subsubseção 3.1.2 os dados gerais relativos a todos os experimentos executados.

#### 3.1.1 Melhor Perceptron

Esta subsubseção apresenta os resultados obtidos somente do melhor Perceptron executado dentre todos os experimentos. Na ocasião, o **experimento 10** do **ciclo 1** ( $LR = 1.0$ ) obteve os melhores valores. Na Figura 1, é possível visualizar o gráfico de convergência do tipo MSE por Época, para os subconjuntos de treinamento e validação. No Bloco de Anotação 1 é possível visualizar os dados relativos aos parâmetros iniciais e finais de execução do Perceptron. Por fim, na Tabela 1, na Tabela 2 e na Tabela 3 são apresentadas as matrizes de confusão dos subconjuntos de treinamento, validação e teste, respectivamente.

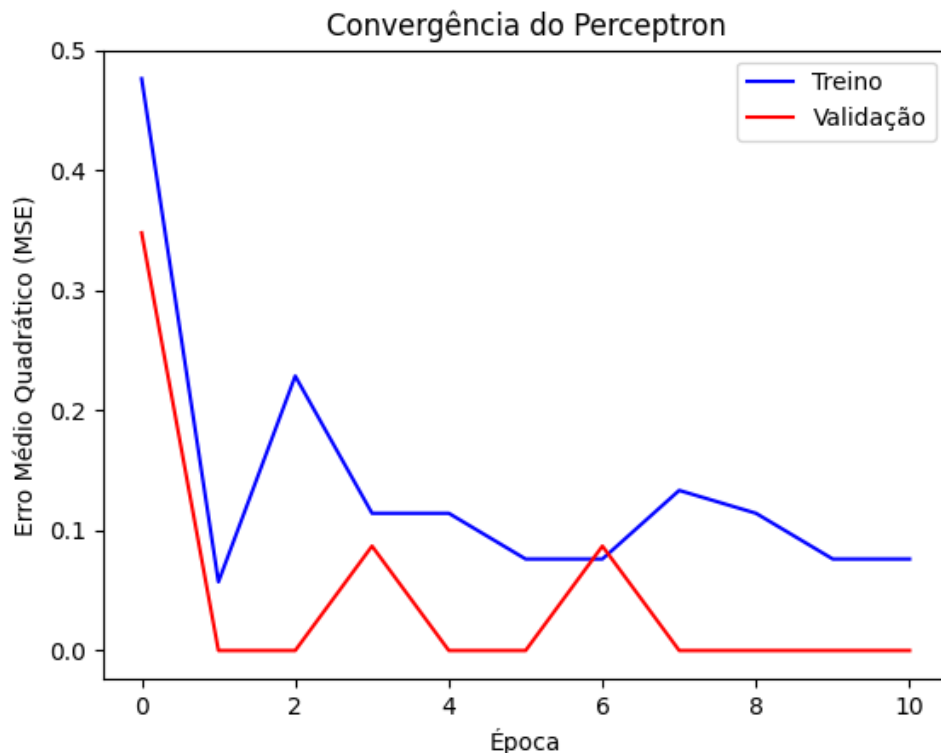


Figura 1: Gráfico de Convergência com o MSE do Treino e Validação.

<sup>1</sup>Disponível em: <https://archive.ics.uci.edu/ml/datasets/Iris>. Acesso em 05 de maio de 2023.

```

1 W_inicial:
2   [
3     X1  [0.38816993, 0.64328822, 0.45825289],
4     X2  [0.54561679, 0.94146481, 0.38610264],
5     X3  [0.96119056, 0.90535064, 0.19579113],
6     X4  [0.0693613 , 0.100778 , 0.01822183]
7   ]
8
9 bias_inicial:
10  [
11    X0  [0.09444296, 0.68300677, 0.07118865]
12  ]
13
14 W_final:
15  [
16    X1  [-1.87514334, 2.85304987, 1.07934184],
17    X2  [ 2.11181856, -0.47011656, -0.73103035],
18    X3  [-3.546546 , 0.42625446, 4.64566631],
19    X4  [-3.11013831, -1.02984034, 5.83854558]
20  ]
21
22 bias_final:
23  [
24    X0  [-0.16755736, 3.21233911, -1.81817503]
25  ]
26
27 Taxa de Aprendizado: 1.0

```

Bloco de Anotação 1: Dados dos Parâmetros do melhor Perceptron.

Treinamento	Setosa (0)	Versicolour (1)	Virginica (2)
Setosa (0)	31	0	0
Versicolour (1)	0	32	5
Virginica (2)	0	0	37

Tabela 1: Matriz de Confusão para o Subconjunto de Treinamento

Validação	Setosa (0)	Versicolour (1)	Virginica (2)
Setosa (0)	6	0	0
Versicolour (1)	0	10	0
Virginica (2)	0	0	7

Tabela 2: Matriz de Confusão para o Subconjunto de Validação

Teste	Setosa (0)	Versicolour (1)	Virginica (2)
Setosa (0)	13	0	0
Versicolour (1)	0	3	0
Virginica (2)	0	0	6

Tabela 3: Matriz de Confusão para o Subconjunto de Teste

Com base nos dados coletados, é possível inferir que houve **o total de 5 erros de classificação para o subconjunto de treinamento**, indicado pelo falso negativo da classe 1 na classe 2 ( $MC_{Treinamento_{23}}$ ). No entanto, é importante ressaltar que **nenhum erro** foi registrado para os subconjuntos de validação e teste, indicando que o modelo desenvolvido apresentou boa generalização para novos dados. Diante dessas informações, pode-se concluir que o Perceptron foi capaz de aprender com eficiência as características do conjunto de dados analisado e produzir resultados satisfatórios na tarefa de classificação.

### 3.1.2 Dados Gerais do Conjunto de Experimentos

Esta subsubseção apresenta os resultados obtidos sobre todos os experimentos executados. Na Tabela 4, na Tabela 5 e na Tabela 6 são apresentadas tabelas de acurácia, número de épocas e tempo de execução, respectivamente.

Taxa de Aprendizado	Média	Desvio Padrão	Mínimo	Mediana	Máximo
1.0000	<b>0.9982</b>	<b>0.0089</b>	<b>0.9545</b>	<b>1.0000</b>	<b>1.0000</b>
0.1000	<b>0.9982</b>	<b>0.0089</b>	<b>0.9545</b>	<b>1.0000</b>	<b>1.0000</b>
0.0100	0.9855	0.0279	0.9091	<b>1.0000</b>	<b>1.0000</b>
0.0010	0.9055	0.0760	0.7273	0.9091	<b>1.0000</b>

Tabela 4: Acurácia dos Experimentos Executados

Taxa de Aprendizado	Média	Desvio Padrão	Mínimo	Mediana	Máximo
1.0000	13.4800	<b>2.4839</b>	11.0000	<b>13.0000</b>	20.0000
0.1000	<b>13.3200</b>	2.6940	<b>10.0000</b>	<b>13.0000</b>	<b>18.0000</b>
0.0100	15.2000	3.3823	12.0000	14.0000	26.0000
0.0010	24.2400	8.4204	<b>10.0000</b>	23.0000	42.0000

Tabela 5: Número de Épocas dos Experimentos Executados

Taxa de Aprendizado	Média	Desvio Padrão	Mínimo	Mediana	Máximo
1.0000	0.1742	0.0603	0.0727	0.1799	0.2837
0.1000	0.1558	<b>0.0468</b>	<b>0.0717</b>	0.1614	<b>0.2323</b>
0.0100	<b>0.1456</b>	0.0526	0.0738	<b>0.1582</b>	0.2532
0.0010	0.2344	0.1182	0.1007	0.2013	0.4926

Tabela 6: Tempo (em segundos) de Execução dos Experimentos

Com base nos dados coletados, é possível inferir que as taxas  $LR = 1.0$  e  $LR = 0.1$  tiveram melhor desempenho em termos de acurácia do que as outras taxas testadas, como indicado pelos seus valores de média, desvio padrão, mínimo, mediana e máximo. Vale ressaltar que o melhor valor obtido, nesta ocasião, foi de **100%**. No entanto, a  $LR = 0.1$  obteve um melhor desempenho em termos do número de épocas executadas, exigindo do menor número possível de iterações durante os experimentos. Além disso, a taxa  $LR = 0.1$  também teve os melhores valores gerais em relação ao tempo de execução, como indicado pelos seus valores de desvio padrão, mínimo e máximo.

## 3.2 Exercício 2

O segundo exercício tinha por objetivo reconhecer as classes advindas do *Wine Data Set*<sup>2</sup>, que foi resultado de uma análise química de vinhos cultivados na mesma região da Itália, mas derivados de três cultivares diferentes. O conjunto de dados é composto por **178 amostras** de três tipos de vinho distintos, distribuídos em: **classe 1 com 59 amostras**, **classe 2 com 71 amostras** e **classe 3 com 48 amostras**. Cada amostra possui **13 características numéricas**, que representam diferentes atributos químicos dos vinhos, tais como Álcool (em %), Ácido Málico (em g/L), Cinzas (em g/L), Alcalinidade das Cinzas (em mg/L), Magnésio (em mg/L), Fenóis Totais (em mg/L), Flavonoides (em mg/L), Fenóis Não Flavonoides (em mg/L), Proantocianidinas (em mg/L), Intensidade da Cor (em unidades de absorbância a 520 nm), Matiz (em graus) e OD280 (em unidades de absorbância). É importante destacar que, semelhante ao *Iris Data Set* da Subseção 3.1, **todas as classes são conhecidas** e **todos os atributos são do tipo numérico**. Por fim, na Subsubseção 3.2.1 é possível visualizar as informações e resultados do melhor Perceptron obtido durante os experimentos e na Subsubseção 3.2.2 os dados gerais relativos a todos os experimentos executados.

### 3.2.1 Melhor Perceptron

Esta subsubseção apresenta os resultados obtidos somente do melhor Perceptron executado dentre todos os experimentos. Na ocasião, o **experimento 05 do ciclo 1** ( $LR = 1.0$ ) obteve os melhores valores. Na Figura 2, é possível visualizar o gráfico de convergência do tipo MSE por Época, para os subconjuntos de treinamento e validação. No Bloco de Anotação 2 é possível visualizar os dados relativos aos parâmetros iniciais e finais de execução do Perceptron. Por fim, na Tabela 7, na Tabela 8 e na Tabela 9 são apresentadas as matrizes de confusão dos subconjuntos de treinamento, validação e teste, respectivamente.

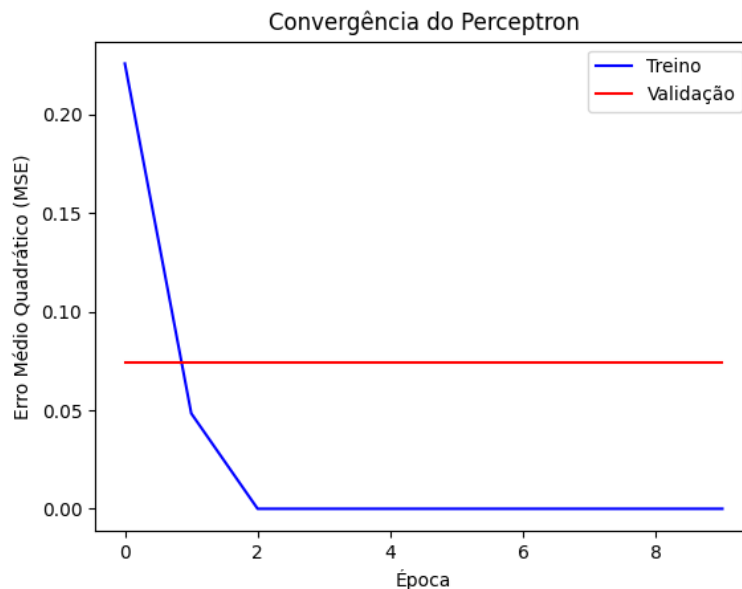


Figura 2: Gráfico de Convergência com o MSE do Treino e Validação.

<sup>2</sup>Disponível em: <https://archive.ics.uci.edu/ml/datasets/Wine>. Acesso em 05 de maio de 2023.



```

1 W_inicial:
2   [
3       X1      [0.24966801, 0.48942496, 0.22120944],
4       X2      [0.98766801, 0.94405934, 0.03942681],
5       X3      [0.70557517, 0.92524832, 0.18057535],
6       X4      [0.56794523, 0.9154883 , 0.03394598],
7       X5      [0.69742027, 0.29734901, 0.9243962 ],
8       X6      [0.97105825, 0.94426649, 0.47421422],
9       X7      [0.86204265, 0.8445494 , 0.31910047],
10      X8      [0.82891547, 0.03700763, 0.59626988],
11      X9      [0.23000884, 0.12056689, 0.0769532 ],
12      X10     [0.69628878, 0.33987496, 0.72476677],
13      X11     [0.06535634, 0.31529034, 0.53949129],
14      X12     [0.79072316, 0.3187525 , 0.62589138],
15      X13     [0.88597775, 0.61586319, 0.23295947]
16   ]
17
18 bias_inicial:
19   [
20       X0      [0.02440078, 0.87009887, 0.02126941]
21   ]
22
23 W_final:
24   [
25       X1      [ 3.24249302, -3.18571448, 1.70149676],
26       X2      [ 2.16575398, -1.91046754, 2.24277022],
27       X3      [ 3.89048993, -4.43062829, 2.50452508],
28       X4      [-4.21027078, 5.33175283, -0.18383252],
29       X5      [-2.80128198, 2.93509857, 2.04047624],
30       X6      [ 2.62619917, 0.42570587, -1.92646657],
31       X7      [ 4.39270769, 0.02963911, -1.77198477],
32       X8      [-0.1609927 , 1.7697079 , -0.45179547],
33       X9      [ 0.14555461, 2.94812758, -1.35968331],
34       X10     [ 2.56046905, -3.98574311, 3.97706027],
35       X11     [-1.87460599, 5.41349204, -0.97548552],
36       X12     [ 1.87403786, 0.61765675, -1.66464296],
37       X13     [ 4.30698302, -3.5906482 , 1.07253365]
38   ]
39
40 bias_final:
41   [
42       X0      [-0.32829932, 1.61812824, 0.35816272]
43   ]
44
45 Taxa de Aprendizado: 1.0

```

Bloco de Anotação 2: Dados dos Parâmetros do melhor Perceptron.

Treinamento	Classe 0	Classe 1	Classe 2
Classe 0	39	1	0
Classe 1	0	50	0
Classe 2	0	0	34

Tabela 7: Matriz de Confusão para o Subconjunto de Treinamento

Validação	Classe 0	Classe 1	Classe 2
Classe 0	9	0	0
Classe 1	0	9	1
Classe 2	0	0	8

Tabela 8: Matriz de Confusão para o Subconjunto de Validação

Teste	Classe 0	Classe 1	Classe 2
Classe 0	10	0	0
Classe 1	0	11	0
Classe 2	0	0	6

Tabela 9: Matriz de Confusão para o Subconjunto de Teste

Com base nos dados coletados, é possível inferir que houve **o total de 1 erro de classificação para o subconjunto de treinamento**, indicado pelo falso negativo da classe 0 na classe 1 ( $MC_{Treinamento_{12}}$ ), e o **o total de 1 erro de classificação para o subconjunto de validação**, indicado pelo falso negativo da classe 1 na classe 2 ( $MC_{Validacao_{23}}$ ). No entanto, é importante ressaltar que **nenhum erro** foi registrado para o subconjunto de teste, indicando que o modelo desenvolvido apresentou boa generalização para novos dados. Diante dessas informações, pode-se concluir que o Perceptron foi capaz de aprender com eficiência as características do conjunto de dados analisado e produzir resultados satisfatórios na tarefa de classificação.

### 3.2.2 Dados Gerais do Conjunto de Experimentos

Esta subsubseção apresenta os resultados obtidos sobre todos os experimentos executados. Na Tabela 10, na Tabela 11 e na Tabela 12 são apresentadas tabelas de acurácia, número de épocas e tempo de execução, respectivamente.

Taxa de Aprendizado	Média	Desvio Padrão	Mínimo	Mediana	Máximo
1.0000	0.9467	<b>0.0279</b>	<b>0.8889</b>	<b>0.9630</b>	<b>1.0000</b>
0.1000	<b>0.9526</b>	0.0306	<b>0.8889</b>	<b>0.9630</b>	<b>1.0000</b>
0.0100	0.9304	0.0337	0.8519	0.9259	<b>1.0000</b>
0.0010	0.8726	0.0791	0.6296	0.8889	0.9630

Tabela 10: Acurácia dos Experimentos Executados

Taxa de Aprendizado	Média	Desvio Padrão	Mínimo	Mediana	Máximo
1.0000	11.4800	1.3600	<b>10.0000</b>	<b>11.0000</b>	<b>14.0000</b>
0.1000	<b>11.0800</b>	<b>1.1973</b>	<b>10.0000</b>	<b>11.0000</b>	<b>14.0000</b>
0.0100	13.5600	2.6993	<b>10.0000</b>	12.0000	21.0000
0.0010	22.8000	6.4374	15.0000	22.0000	37.0000

Tabela 11: Número de Épocas dos Experimentos Executados

Taxa de Aprendizado	Média	Desvio Padrão	Mínimo	Mediana	Máximo
1.0000	0.1241	0.0502	0.0717	0.1081	0.2114
0.1000	<b>0.1207</b>	<b>0.0436</b>	<b>0.0705</b>	<b>0.0972</b>	<b>0.1963</b>
0.0100	0.1703	0.0544	0.0830	0.1724	0.2879
0.0010	0.2237	0.1073	0.1008	0.1906	0.5241

Tabela 12: Tempo (em segundos) de Execução dos Experimentos

Com base nos dados coletados, é possível inferir que as taxas  $LR = 1.0$  e  $LR = 0.1$  tiveram melhor desempenho em termos de acurácia do que as outras taxas testadas, como indicado pelos seus valores de média ( $LR = 0.1$  melhor), desvio padrão ( $LR = 1.0$  melhor), mínimo, mediana e máximo. Vale ressaltar que o melhor valor obtido, nesta ocasião, foi de **100%**. No entanto, a  $LR = 0.1$  obteve um melhor desempenho em termos do número de épocas executadas, exigindo do menor número possível de iterações gerais durante os experimentos. Além disso, a taxa  $LR = 0.1$  também teve os melhores valores gerais em relação ao tempo de execução, como indicado pelos seus valores de média, desvio padrão, mínimo, mediana e máximo.

### 3.3 Exercício 3

O terceiro exercício consistia na aplicação de uma ou mais redes neurais para quaisquer outro tipos de problemas (como classificação ou regressão). Optou-se, desta forma, por avaliar a aplicabilidade do Perceptron com relação sua principal variância denominada Perceptron de Múltiplas Camadas (MLP) para um problema de classificação. O conjunto de dados escolhido foi o *Blood Transfusion Service Center Data Set*<sup>3</sup>, composto por informações sobre doações de sangue coletadas de um centro de serviços de transfusão de sangue na cidade de Hsin-Chu, Taiwan. O objetivo é prever se um doador fará ou não uma doação na próxima campanha, que ocorre a cada três meses. O conjunto contém um total de **748 amostras**, com cada amostra representando um doador de sangue que realizou uma doação em março de 2007. Cada amostra possui **4 características numéricas** no formato AFMT: quantidade de meses desde a última doação (Atualidade), número total de doações (Frequência), total em cm<sup>3</sup> de sangue doado (Monetário) e quantidade de meses desde a primeira doação (Tempo). É importante destacar que, das **748 amostras, 178 doadores realizaram uma doação (classe 1)** na campanha seguinte, enquanto que **570 doadores não realizaram uma doação (classe 0)**, o que caracteriza um conjunto desequilibrado e, portanto, um desafio para o modelo de classificação do Perceptron ou MLP. Por fim, na Subsubseção 3.3.1 é possível visualizar as informações e resultados dos experimentos com o Perceptron, na Subsubseção 3.3.2 as informações e resultados dos experimentos com o MLP e na Subsubseção 3.3.3 a comparação entre as duas redes neurais baseada nos resultados obtidos.

#### 3.3.1 Perceptron

Esta subsubseção apresenta os resultados obtidos com a execução do Perceptron, sendo possível visualizar os dados do melhor Perceptron obtido no parágrafo 3.3.1.1 e os dados gerais de todos os experimentos executados no parágrafo 3.3.1.2.

##### 3.3.1.1 Melhor Perceptron

Este parágrafo apresenta os resultados obtidos somente do melhor Perceptron executado dentre todos os experimentos. Na ocasião, o **experimento 12** do **ciclo 3** ( $LR = 0.01$ ) obteve os melhores valores. Na Figura 3, é possível visualizar o gráfico de convergência do tipo MSE por Época, para os subconjuntos de treinamento e validação. No Bloco de Anotação 3 é possível visualizar os dados relativos aos parâmetros iniciais e finais de execução do Perceptron. Por fim, na Tabela 13, na Tabela 14 e na Tabela 15 são apresentadas as matrizes de confusão dos subconjuntos de treinamento, validação e teste, respectivamente.

---

<sup>3</sup>Disponível em: <https://encurtador.com.br/imuvC>. Acesso em 05 de maio de 2023.

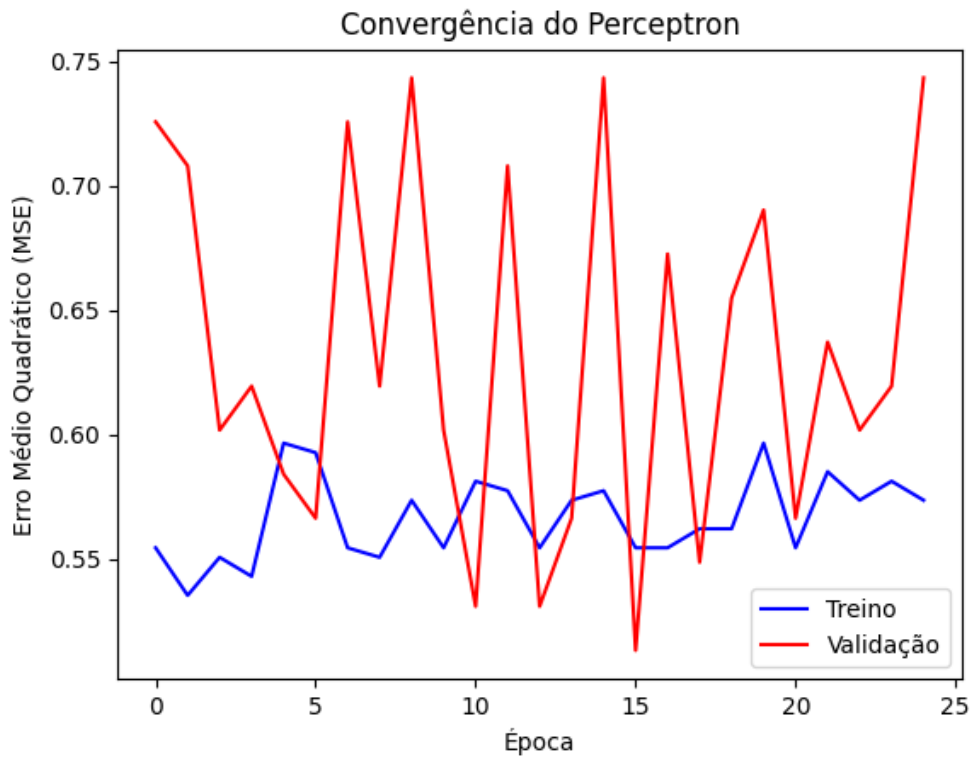


Figura 3: Gráfico de Convergência com o MSE do Treino e Validação.

```

1 W_inicial:
2   [
3     W1      W2
4     X1 [0.94045858, 0.95392858],
5     X2 [0.91486439, 0.3701587 ],
6     X3 [0.01545662, 0.92831856],
7     X4 [0.42818415, 0.96665482]
8   ]
9 bias_inicial:
10  [
11    W0      W0
12    X0 [0.96361998, 0.85300946]
13  ]
14 W_final:
15  [
16    W1      W2
17    X1 [0.23980558, 0.21473429],
18    X2 [0.9969705 , 0.80506247],
19    X3 [0.33021851, 0.58770716],
20    X4 [0.71106022, 0.66136279]
21  ]
22 bias_final:
23  [
24    W0      W0
25    X0 [0.40965058, 0.36185229]
26  ]
27 Taxa de Aprendizado: 0.01

```

Bloco de Anotação 3: Dados dos Parâmetros do melhor Perceptron.

Treinamento	Não Doou Sangue (1)	Doou Sangue (2)
Não Doou Sangue (1)	371	34
Doou Sangue (2)	77	41

Tabela 13: Matriz de Confusão para o Subconjunto de Treinamento

Validação	Não Doou Sangue (1)	Doou Sangue (2)
Não Doou Sangue (1)	72	5
Doou Sangue (2)	24	12

Tabela 14: Matriz de Confusão para o Subconjunto de Validação

Teste	Não Doou Sangue (1)	Doou Sangue (2)
Não Doou Sangue (1)	81	7
Doou Sangue (2)	14	10

Tabela 15: Matriz de Confusão para o Subconjunto de Teste

Com base nas matrizes de confusão coletadas, é possível inferir que o subconjunto de treinamento apresentou **um total de 111 erros de classificação**, dos quais 34 foram falsos positivos da classe 1 na classe 2 ( $MC_{Treinamento_{12}}$ ) e 77 foram falsos negativos da classe 2 na classe 1 ( $MC_{Treinamento_{21}}$ ). Já o subconjunto de validação apresentou **um total de 29 erros de classificação**, sendo 5 falsos positivos da classe 1 na classe 2 ( $MC_{Validacao_{12}}$ ) e 24 falsos negativos da classe 2 na classe 1 ( $MC_{Validacao_{21}}$ ). Por fim, o subconjunto de teste apresentou **um total de 21 erros de classificação**, dos quais 7 foram falsos positivos da classe 1 na classe 2 ( $MC_{Teste_{12}}$ ) e 14 foram falsos negativos da classe 2 na classe 1 ( $MC_{Teste_{21}}$ ). Apesar dos erros de classificação encontrados, o modelo desenvolvido apresentou uma boa generalização para novos dados, com uma acurácia média de 75% em relação aos três subconjuntos de dados. Diante desses resultados, conclui-se que o Perceptron foi capaz de aprender eficientemente as características do conjunto de dados analisado e produzir resultados satisfatórios na tarefa de classificação. No entanto, é importante salientar que o modelo teve certa dificuldade em distinguir as duas classes.

### 3.3.1.2 Dados Gerais do Conjunto de Experimentos

Este parágrafo apresenta os resultados obtidos sobre todos os experimentos executados. Na Tabela 16, na Tabela 17 e na Tabela 18 são apresentadas tabelas de acurácia, número de épocas e tempo de execução, respectivamente.

Taxa de Aprendizado	Média	Desvio Padrão	Mínimo	Mediana	Máximo
1.0000	0.7821	<b>0.0138</b>	<b>0.7589</b>	0.7857	0.8036
0.1000	0.7839	0.0168	0.7500	0.7857	0.8036
0.0100	<b>0.7889</b>	0.0155	0.7500	<b>0.7946</b>	<b>0.8125</b>
0.0010	0.7796	0.0144	0.7500	0.7857	0.8036

Tabela 16: Acurácia dos Experimentos Executados do Perceptron

Taxa de Aprendizado	Média	Desvio Padrão	Mínimo	Mediana	Máximo
1.0000	<b>13.7200</b>	<b>3.0003</b>	<b>10.0000</b>	<b>13.0000</b>	<b>21.0000</b>
0.1000	15.1600	5.2966	<b>10.0000</b>	14.0000	31.0000
0.0100	14.9600	4.7873	<b>10.0000</b>	<b>13.0000</b>	29.0000
0.0010	15.6000	4.4000	<b>10.0000</b>	14.0000	27.0000

Tabela 17: Número de Épocas dos Experimentos Executados do Perceptron

Taxa de Aprendizado	Média	Desvio Padrão	Mínimo	Mediana	Máximo
1.0000	0.5580	0.2627	0.2412	0.5241	<b>1.0509</b>
0.1000	0.5899	0.2965	<b>0.2363</b>	0.5435	1.3157
0.0100	0.5855	0.2980	0.2416	0.5501	1.5235
0.0010	<b>0.5206</b>	<b>0.2542</b>	0.2575	<b>0.3806</b>	1.1424

Tabela 18: Tempo (em segundos) de Execução dos Experimentos do Perceptron

Com base nos dados coletados, é possível inferir que a  $LR = 0.01$  teve um melhor desempenho em termos de acurácia do que as outras taxas testadas, como indicado pelos seus valores de média, mediana e máximo. Vale ressaltar que o melhor valor obtido, nesta ocasião, foi de **81,25%**. No entanto, a  $LR = 1.0$  obteve um melhor desempenho em termos do número de épocas executadas, exigindo do menor número possível de iterações gerais durante os experimentos. Por outro lado, a  $LR = 0.001$  teve os melhores valores gerais em relação ao tempo de execução, como indicado pelos seus valores de média, desvio padrão e mediana.

### 3.3.2 Multi Layer Perceptron (MLP)

Esta subsubseção apresenta os resultados obtidos com a execução do MLP, sendo possível visualizar os dados do melhor MLP obtido no parágrafo 3.3.2.1 e os dados gerais de todos os experimentos executados no parágrafo 3.3.2.2. Vale lembrar que o MLP de Classificação utilizado foi o da biblioteca *SciKit-Learn* e, portanto, não foi possível obter tanto os dados iniciais de  $W$  e  $bias$  quanto aos dados de MSE de validação durante a fase de treinamento.

#### 3.3.2.1 Melhor MLP

Este parágrafo apresenta os resultados obtidos somente do melhor MLP executado dentre todos os experimentos. Na ocasião, o **experimento 18 do ciclo 2** ( $LR = 0.1$ ) obteve os melhores valores. Na Figura 4, é possível visualizar o gráfico de convergência do tipo MSE por Época para o subconjunto de treinamento. No Bloco de Anotação 4 é possível visualizar os dados relativos aos parâmetros finais de execução do MLP. Por fim, na Tabela 19, na Tabela 20 e na Tabela 21 são apresentadas as matrizes de confusão dos subconjuntos de treinamento, validação e teste, respectivamente

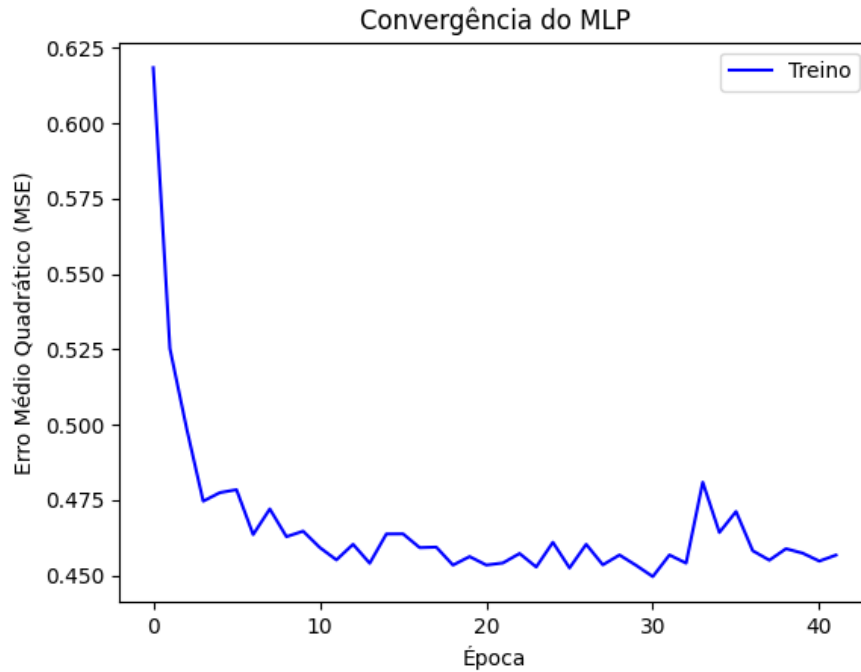


Figura 4: Gráfico de Convergência com o MSE do Treino e Validação.

```

1 W_final:
2   Layer_0:
3     [
4       X1 [ 0.26999322, -1.13178202, -1.91803837, 0.98824385,
5           0.30952927, 0.67662349, 1.0942234, -0.76561493],
6       X2 [-0.04420138, -0.29309681, 0.40112174, 0.08610402,
7           -0.14811847, 0.01243883, 0.51748758, -1.85662184],
8       X3 [-0.11817355, 0.57466788, 0.93007403, -0.77562422,
9           -0.15473909, 0.37245387, -0.0304468, -0.8533475 ],
10      X4 [ 0.42456575, -1.27479803, -1.20763379, -0.72109627,
11          0.13977206, 1.05033049, -0.61145965, 1.24399461]
12    ]
13   Layer_1:
14     [
15       a1 [-0.04640066], a2 [ 0.28481175], a3 [ 0.43619994],
16       a4 [-0.54595283], a5 [ 0.00819404], a6 [-0.27617814],
17       a7 [ 0.04027045], a8 [-0.59960764]
18     ]
19
20 bias_final:
21   Layer_0:
22     [
23       X0 [-0.80080489, -0.10056555, -0.45219772, 0.8667518,
24          -0.14883133, 0.13285289, -0.45781377, 0.28181391]
25     ]
26   Layer_1:
27     [
28       a0 [-0.59110128]
29     ]
30
31 Taxa de Aprendizado: 0.1

```

Bloco de Anotação 4: Dados dos Parâmetros do melhor MLP.



Treinamento	Não Doou Sangue (1)	Doou Sangue (2)
Não Doou Sangue (1)	452	31
Doou Sangue (2)	91	61

Tabela 19: Matriz de Confusão para o Subconjunto de Treinamento

Validação	Não Doou Sangue (1)	Doou Sangue (2)
Não Doou Sangue (1)	81	4
Doou Sangue (2)	19	9

Tabela 20: Matriz de Confusão para o Subconjunto de Validação

Teste	Não Doou Sangue (1)	Doou Sangue (2)
Não Doou Sangue (1)	83	4
Doou Sangue (2)	16	10

Tabela 21: Matriz de Confusão para o Subconjunto de Teste

Com base nas matrizes de confusão coletadas, é possível inferir que o subconjunto de treinamento apresentou **um total de 122 erros de classificação**, dos quais 31 foram falsos positivos da classe 1 na classe 2 ( $MC_{Treinamento_{12}}$ ) e 91 foram falsos negativos da classe 2 na classe 1 ( $MC_{Treinamento_{21}}$ ). Já o subconjunto de validação apresentou **um total de 23 erros de classificação**, sendo 4 falsos positivos da classe 1 na classe 2 ( $MC_{Validacao_{12}}$ ) e 19 falsos negativos da classe 2 na classe 1 ( $MC_{Validacao_{21}}$ ). Por fim, o subconjunto de teste apresentou **um total de 20 erros de classificação**, dos quais 4 foram falsos positivos da classe 1 na classe 2 ( $MC_{Teste_{12}}$ ) e 16 foram falsos negativos da classe 2 na classe 1 ( $MC_{Teste_{21}}$ ). Apesar dos erros de classificação encontrados, o modelo desenvolvido apresentou uma boa generalização para novos dados, com uma acurácia média de 80,95% em relação aos três subconjuntos de dados. Diante desses resultados, conclui-se que o MLP foi capaz de aprender eficientemente as características do conjunto de dados analisado e produzir resultados satisfatórios na tarefa de classificação. No entanto, é importante salientar que o modelo teve certa dificuldade em distinguir as duas classes.

### 3.3.2.2 Dados Gerais do Conjunto de Experimentos

Este parágrafo apresenta os resultados obtidos sobre todos os experimentos executados. Na Tabela 22, na Tabela 23 e na Tabela 24 são apresentadas tabelas de acurácia, número de épocas e tempo de execução, respectivamente.

Taxa de Aprendizado	Média	Desvio Padrão	Mínimo	Mediana	Máximo
1.0000	0.7688	<b>0.0058</b>	0.7434	0.7699	0.7788
0.1000	<b>0.7972</b>	0.0120	0.7699	<b>0.7965</b>	<b>0.8230</b>
0.0100	0.7929	0.0090	<b>0.7788</b>	<b>0.7965</b>	0.8142
0.0010	0.7798	0.0133	0.7522	0.7788	0.8053

Tabela 22: Acurácia dos Experimentos Executados do MLP

Taxa de Aprendizado	Média	Desvio Padrão	Mínimo	Mediana	Máximo
1.0000	<b>36.7200</b>	<b>10.3789</b>	<b>15.0000</b>	<b>34.0000</b>	<b>62.0000</b>
0.1000	44.1600	12.2888	23.0000	42.0000	69.0000
0.0100	118.9600	31.3949	70.0000	120.0000	182.0000
0.0010	219.2400	74.1562	123.0000	206.0000	440.0000

Tabela 23: Número de Épocas dos Experimentos Executados do MLP

Taxa de Aprendizado	Média	Desvio Padrão	Mínimo	Mediana	Máximo
1.0000	0.0866	0.0408	0.0320	0.0841	0.2116
0.1000	<b>0.0520</b>	<b>0.0132</b>	<b>0.0290</b>	<b>0.0510</b>	<b>0.0792</b>
0.0100	0.1985	0.1036	0.0851	0.1381	0.4408
0.0010	0.2880	0.1649	0.1452	0.2152	0.9264

Tabela 24: Tempo (em segundos) de Execução dos Experimentos do MLP

Com base nos dados coletados, é possível inferir que a  $LR = 0.1$  teve um melhor desempenho geral em termos de acurácia do que as outras taxas testadas, como indicado pelos seus valores de média, mediana e máximo. Vale ressaltar que o melhor valor obtido, nesta ocasião, foi de **82,30%**. No entanto, a  $LR = 1.0$  obteve um melhor desempenho em termos do número de épocas executadas, exigindo do menor número possível de iterações gerais durante os experimentos. Além disso, a  $LR = 0.1$  teve os melhores valores gerais em relação ao tempo de execução, como indicado pelos seus valores de média, desvio padrão, mínimo, mediana e máximo.

### 3.3.3 Comparação: Perceptron x MLP

Com base nas conclusões obtidas na Subsubseção 3.3.1 e na Subsubseção 3.3.2, é possível comparar o desempenho do Perceptron e do MLP em termos de acurácia, média, número de erros de classificação e capacidade de generalização, para o conjunto de dados *Blood Transfusion Service Center Data Set*.

O Perceptron apresentou um **total de 111 erros de classificação no subconjunto de treinamento, 29 erros no subconjunto de validação e 21 erros no subconjunto de teste**. Enquanto isso, o MLP apresentou **122 erros de classificação no subconjunto de treinamento, 23 erros no subconjunto de validação e 20 erros no subconjunto de teste**. Ambos os modelos tiveram certa dificuldade em distinguir as duas classes, mas o MLP apresentou um **desempenho um pouco melhor** em relação ao número total de erros de classificação.

Já em termos de acurácia média, **o MLP apresentou um desempenho ligeiramente melhor do que o Perceptron**, com uma **acurácia média de 80,95%** em relação aos três subconjuntos de dados, enquanto o Perceptron apresentou **uma acurácia média de 75%**. Quanto à capacidade de generalização, **ambos os modelos apresentaram uma boa generalização para novos dados**, conforme indicado pelas matrizes de confusão e pelas acurácias médias em todos os subconjuntos de dados.

Por fim, no que diz respeito à LR, o MLP obteve melhores resultados gerais com uma LR de  $LR = 0,1$ , com uma **acurácia máxima de 82,30%**. Já para o Perceptron, a  $LR = 0,01$  teve um melhor desempenho em termos de **acurácia média, com um valor máximo de 81,25%**.

## 4 Considerações Finais

Em síntese, este relatório teve como objetivo avaliar o desempenho do Perceptron em diferentes configurações e conjuntos de dados, comparando-o com sua variação MLP em uma das tarefas. Os resultados obtidos, de modo geral, demonstram que o Perceptron é uma ferramenta eficaz para tarefas de classificação binária nos mais diferentes cenários contextuais.

No primeiro exercício, o Perceptron se mostrou capaz de reconhecer as três espécies de flores do gênero *Iris* do conjunto de dados *Iris Data Set*, ainda que diante de todas as condições impostas nos experimentos, como as diferentes taxas de aprendizado e diferentes pesos e *bias* de inicialização. De modo geral, o modelo obteve apenas 5 erros de classificação para o subconjunto de treinamento, enquanto que não houveram erros para os subconjuntos de validação e teste. Vale ressaltar que a melhor taxa de variação foi de  $LR = 1.0$  e obteve uma acurácia máxima de 100% no subconjunto de testes.

No segundo exercício, o Perceptron se mostrou capaz de reconhecer as três classes diferentes de vinho do conjunto de dados *Wine Data Set*, ainda que diante de todas as condições impostas nos experimentos, como as diferentes taxas de aprendizado e diferentes pesos e *bias* de inicialização. De modo geral, o modelo obteve apenas 2 erros de classificação geral, sendo 1 para o subconjunto de treinamento e 1 para o subconjunto de validação, enquanto que não houveram erros para o subconjunto de teste. Vale ressaltar que a melhor taxa de variação foi de  $LR = 1.0$  e obteve uma acurácia máxima, novamente, de 100% no subconjunto de testes.

No terceiro exercício, o Perceptron e o MLP se mostraram capazes de reconhecer as duas classes diferentes de doadores de sangue do conjunto de dados *Blood Transfusion Service Center Data Set*, ainda que diante de todas as condições impostas nos experimentos, como as diferentes taxas de aprendizado e diferentes pesos e *bias* de inicialização. De modo geral, o melhor Perceptron apresentou uma acurácia média de 75%, com alguns erros de classificação, ao passo que o melhor MLP obteve uma acurácia média de 80,95%. Desta forma, o MLP mostrou-se ligeiramente superior ao Perceptron comum, ainda que ambos os modelos tiveram certa dificuldade em distinguir as duas classes do conjunto de dados.

Portanto, pode-se afirmar que o Perceptron é uma ferramenta eficiente e versátil para a solução de problemas de classificação binária em diversas áreas. Seus resultados precisos e sua grande adaptabilidade o tornam uma excelente escolha para a resolução de tarefas de classificação binária em diferentes contextos e cenários.