

**UNIVERSIDADE FEDERAL DE SÃO PAULO  
INSTITUTO DE CIÊNCIA E TECNOLOGIA  
BACHARELADO EM CIÊNCIA E TECNOLOGIA**

**Projeto 01:  
Implementação Multilayer Perceptron**

**Nome:** Willian Dihanster Gomes de Oliveira **RA:** 112269

**SÃO JOSÉ DOS CAMPOS  
2018**

## **Implementação:**

A implementação do Multilayer Perceptron foi desenvolvida em Python 3.

## **Base de Dados:**

- **Banknote Dataset**

O Banknote Dataset é um problema de classificação binária, onde a ideia é prever se uma nota (cédula) é autêntica ou não, dado medidas de uma fotografia.

É composta de 1.372 exemplos com 4 atributos, mais sua classe.

- **Pima Indians Diabetes Dataset**

Esse conjunto de dados é baseado na previsão do início de diabetes em 5 anos ao grupo de nativos-americanos, Pima.

É composto por 768 exemplos, sendo um problema de classificação binária, rotulados como '0' ou '1'. Além disso, cada exemplo possui 8 atributos, mais sua classe.

O Conjunto de Dados de Diabetes dos Índios Pima envolve a previsão do início do diabetes dentro de 5 anos em índios Pima que receberam detalhes médicos.

Os atributos vão de número de vezes grávida, pressão arterial, IMC a concentração plasmática de glicose.

- **Sonar Dataset**

Já o Sonar Dataset é baseado na predição se um objeto é uma mina ou uma rocha, dado a força do retorno solar em diferentes ângulos.

Trata-se também de um problema de classificação binária. Sendo composto de 208 exemplos, com 60 atributos, mais sua classe que é "R" para um rocha e "M" para minas.

## **Parâmetros iniciais:**

- Épocas = 100
- Learning Rate (eta) = 0.5
- Momentum (alfa) = 0.1
- Número de Neurônios = 10
- Camada Oculta = 1

## **Resultados**

Para os experimentos, foram separados 80% dos dados para treino e 20% para teste.

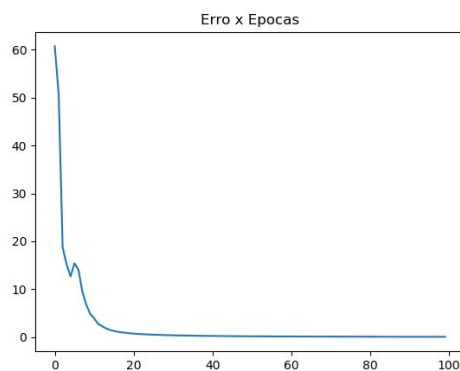
- **Banknote Dataset**

Com os parâmetros iniciais, foi possível obter uma acurácia de 100%, como pode ser visto na figura abaixo e pode-se observar pelo gráfico do erro quadrático médio, que o erro caiu conforme o esperado de acordo com o número de épocas, ficando bem próximo de 0.

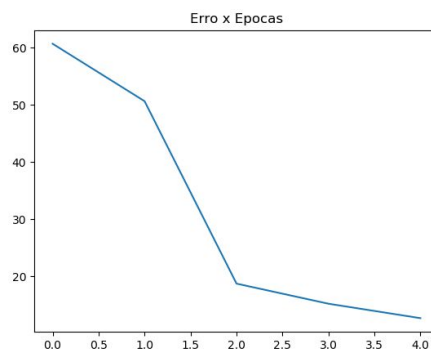
```

esperado=1, previsto=1
esperado=1, previsto=1
esperado=1, previsto=1
esperado=1, previsto=1
esperado=1, previsto=1
esperado=1, previsto=1
esperado=1, previsto=1
esperado=1, previsto=1
esperado=1, previsto=1
esperado=0, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
Acuracia = 100.0 %

```



Como a acurácia para esse modelo se mostrou maior do que o esperado, foram feitos teste com menos épocas. Na figura abaixo é possível ver o gráfico do erro quadrático médio, com apenas 5 épocas, já se obteve um erro bem baixo e uma boa acurácia (92.91 %).



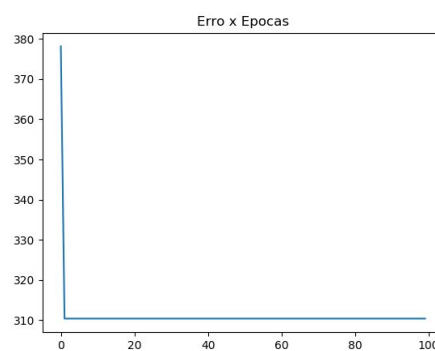
## ● Pima Indians Diabetes Dataset

Com os parâmetros iniciais, os resultados obtidos podem ser conferidos na Figura a seguir. Pode-se observa-se uma acurácia regular, mas o erro da fase de treinamento alto.

```

esperado=0, previsto=0
esperado=1, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
esperado=1, previsto=0
esperado=0, previsto=0
Acuracia = 64.28571428571429 %

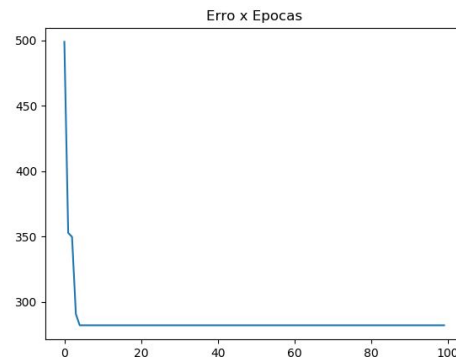
```



Como o erro se mostrou muito alto. Foram feitos novos experimentos variando o learning rate ( $\eta$ ).

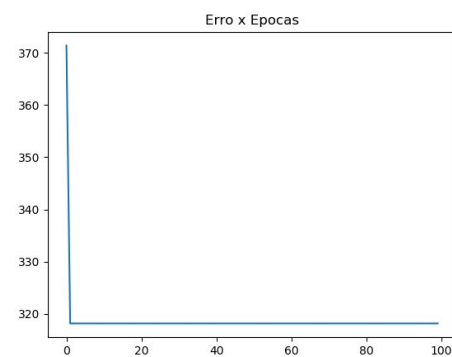
Com  $\eta = 0.1$ , temos o seguinte resultado.

```
esperado=0, previsto=0
esperado=1, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
esperado=1, previsto=0
esperado=0, previsto=0
Acuracia = 64.28571428571429 %
```



Com  $\eta = 0.6$ , temos o seguinte resultados:

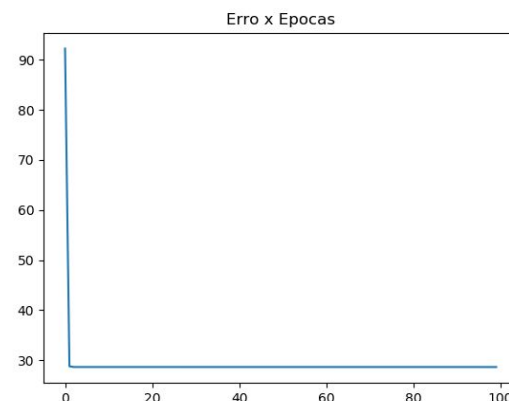
```
esperado=0, previsto=0
esperado=1, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
esperado=0, previsto=0
esperado=1, previsto=0
esperado=0, previsto=0
Acuracia = 64.28571428571429 %
```



## ● Sonar Dataset

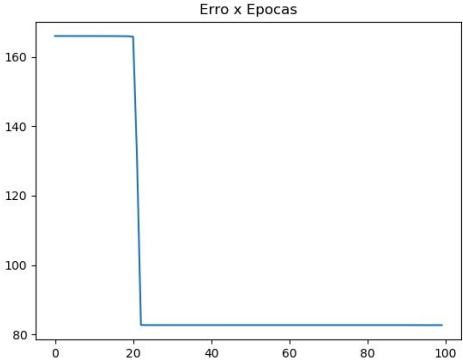
Com os parâmetros definidos inicialmente, os resultados não foram bons. O erro quadrático médio se manteve relativamente alto e acurácia regular com 59%

```
esperado=1, previsto=0
esperado=1, previsto=0
esperado=1, previsto=0
esperado=1, previsto=0
esperado=1, previsto=0
esperado=1, previsto=0
esperado=1, previsto=0
esperado=1, previsto=0
esperado=1, previsto=0
esperado=1, previsto=0
Acuracia = 59.523809523809526 %
```

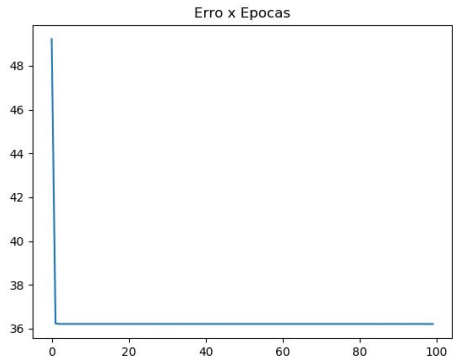


Com o erro se manteve mesmo conforme o número de épocas aumentasse, foram variados outros parâmetros como o número de neurônios e o learning rate ( $\eta$ ).

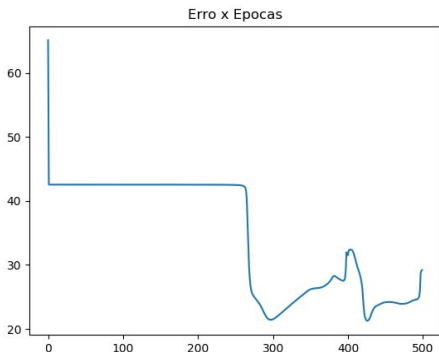
Com mais número de neurônios (20), temos o seguinte resultado:

[illegible]

Pode-se perceber um pior acurácia e uma demora maior no decaimento do erro quadrático médio. Sendo assim, foram feitos novos experimentos, agora diminuindo o número de neurônios (5). Os resultados podem ser conferidos na Figura a seguir.

[illegible]

Com 5 neurônios,  $\eta = 0.3$  e mais épocas (500), foi possível aumentar um pouco a acurácia, chegando a quase 65%. Enquanto que o erro quadrático médio se mostrou um pouco instável.

[illegible]

## Discussões

Os resultados obtidos para a base *Banknote Dataset* podem ser considerados bons em comparação às outras duas bases utilizadas. As duas outras bases tiveram um desempenho mediano/ruim. Para a base *Pima Indians Diabetes Dataset* a situação é diferente, pois mesmo com alterações o erro da fase de treinamento foi elevado, gerando uma situação de *underfitting*, onde a rede não conseguiu aprender nada.

Pesquisando mais a fundo sobre as bases, foi verificado que os resultados estão na média para os resultados encontrados pela internet.

Sendo esses resultados medianos explicados por falta de pré-processamento (como a normalização dos dados) ou mesmo, a qualidade dos dados.

### **Conclusões**

Sendo assim, pode-se verificar a eficácia de uma rede MLP - *Multilayer Perceptron* na classificação de dados. Além, de ser aplicável em várias áreas.