

---

---

# Lista 1

# Aprendizado Profundo

Edgard Henrique, André Hugo, Gustavo Costa

---

---

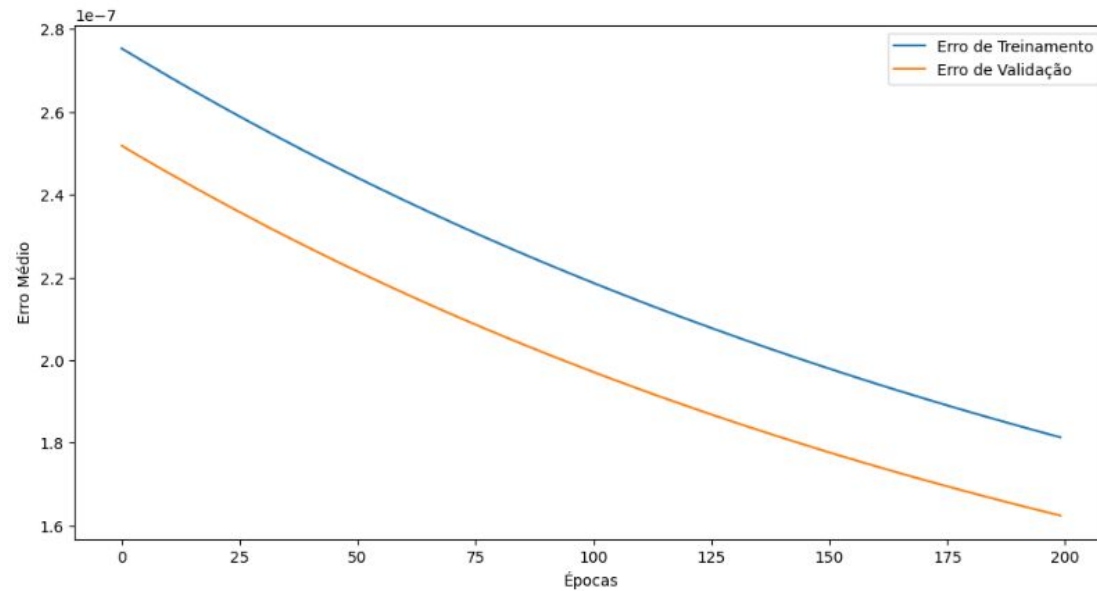
# Questão 1

- Letras A, B e C
    - **Objetivo:** Aproximar as funções  $f(x)=\exp(-x)$ ,  $f(x)=\log_{10}(x)$  no intervalo  $1 \leq x \leq 10$  e  $f(x) = 10x^5 + 5x^4 + 2x^3 - 0.5x^2 + 3x + 2$ , onde  $0 \leq x \leq 5$  utilizando a rede MLP.
    - **Arquitetura:** 2 camadas ocultas com 25 neurônios ativação Tanh para A e B e Relu para C
    - **Learning Rate:** 0.01
    - **Função de Perda:** Erro quadrático médio (MSE)
    - **Otimização:** Algoritmo de descida do gradiente usando Adam.
    - **Treinamento:** 80% para treino e 20% para validação, implementado com Early Stopping para evitar uso de recursos computacionais desnecessários
-

---

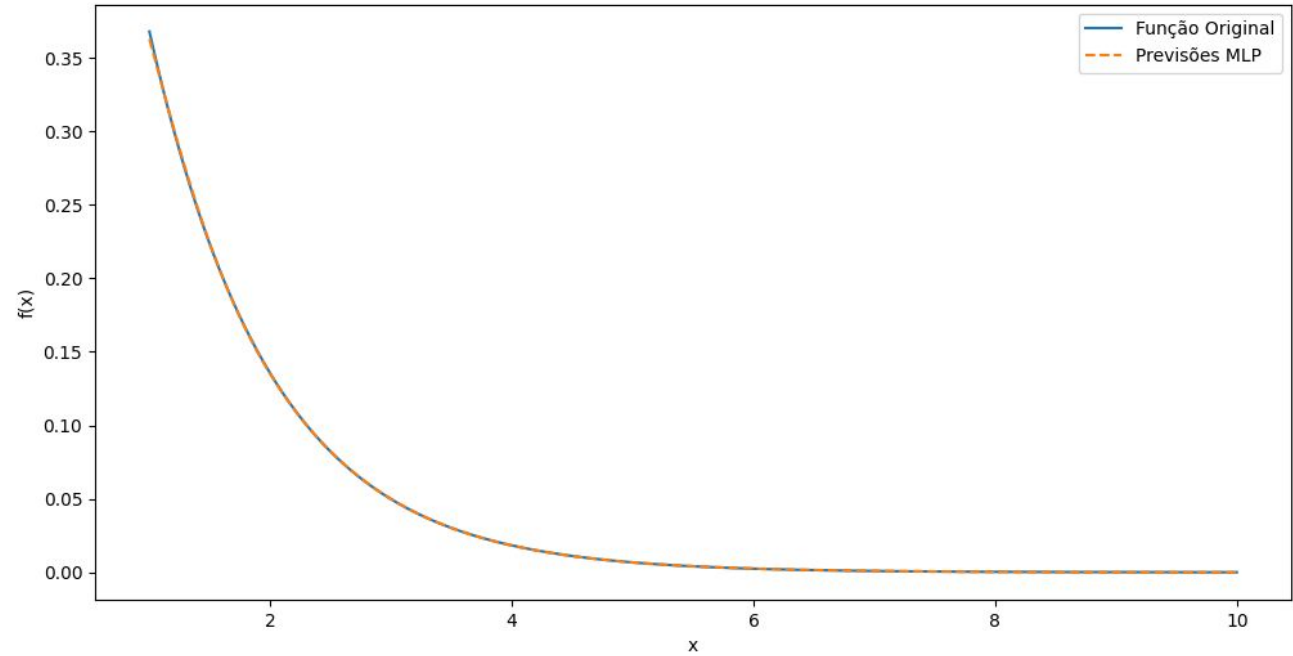
# Treinamento letra A

200 épocas, Loss: 0.0000, Val Loss: 0.0000



---

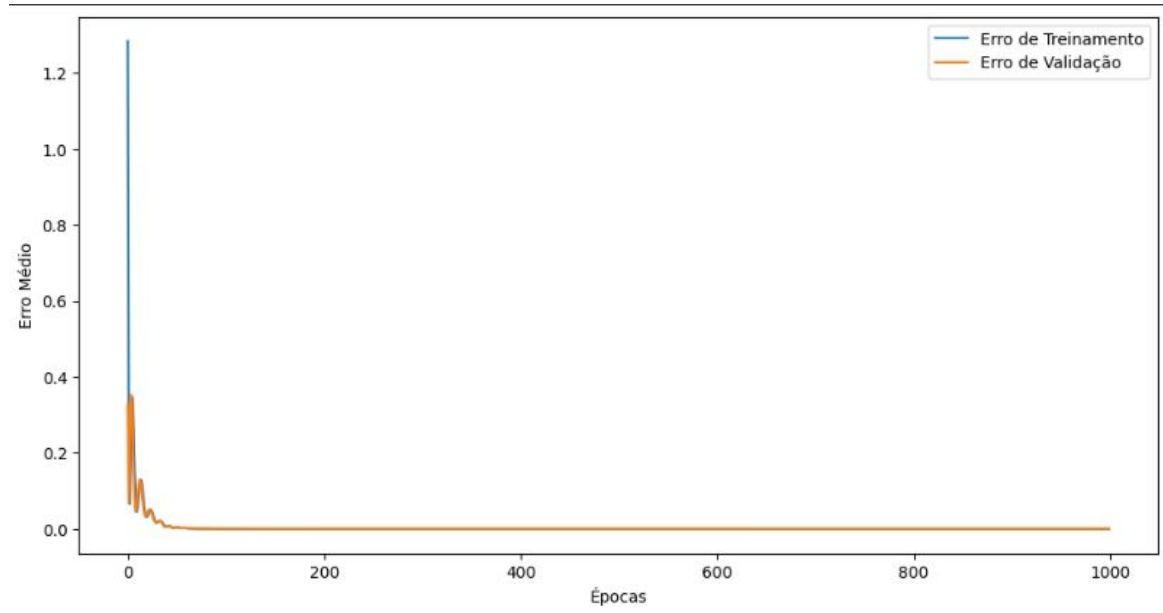
# Resultado letra A



---

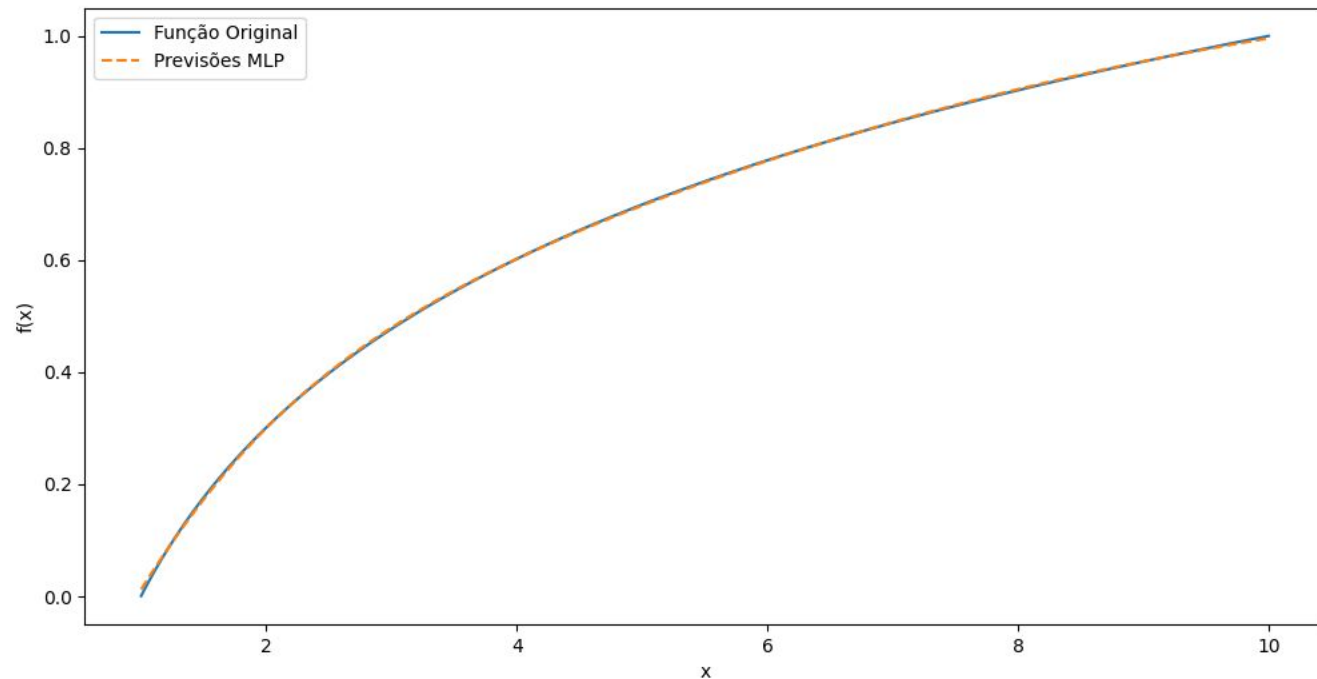
# Treinamento letra B

1000 épocas, Loss: 0.0000, Val Loss: 0.0000



---

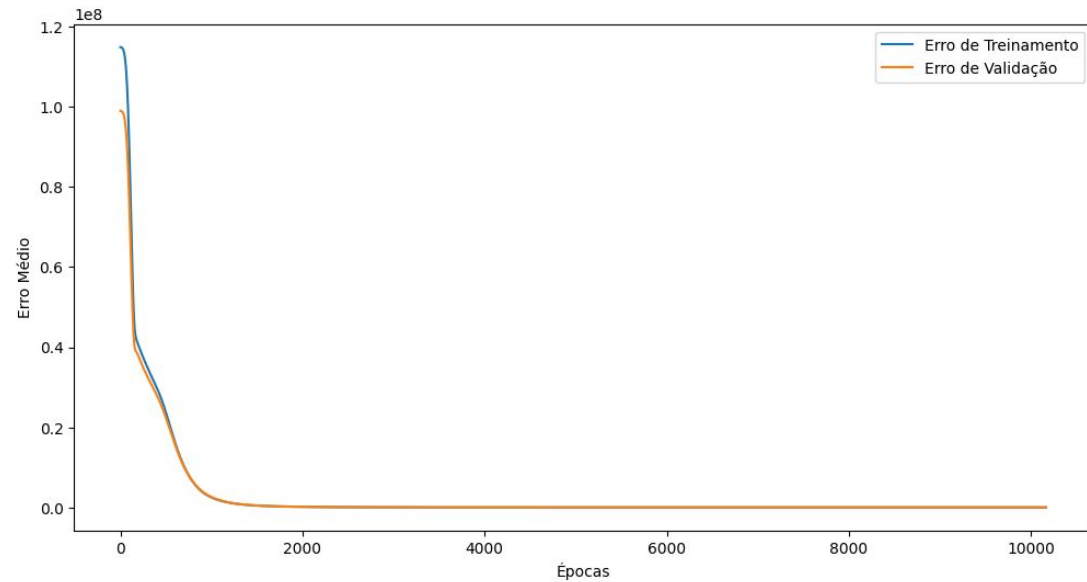
# Resultado letra B



---

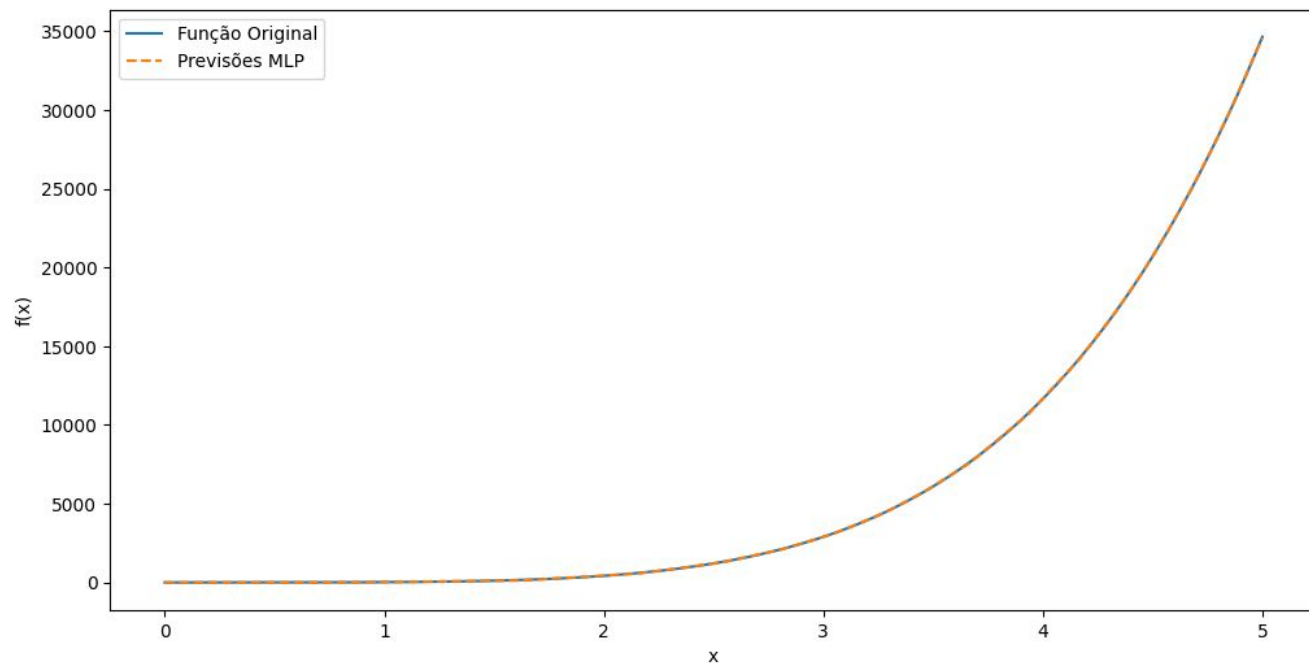
# Treinamento letra C

10168 épocas, Loss: 262.96, Val Loss: 269.00



---

# Resultado letra C

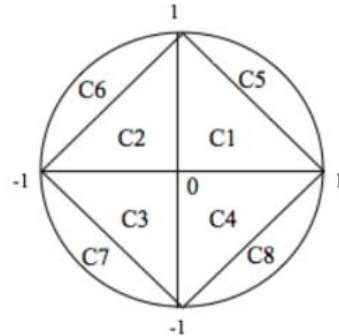




---

## Questão 2

- Duas abordagens de solução implementadas;
- Objetivo: Classificação de um ponto aleatório em um círculo de raio unitário;



---

# Arquitetura - Abordagem 1

- Camada de entrada para 2 valores (Par ordenado:  $x, y$ )
  - 2 camadas ocultas contendo 16 neurônios;
  - Saída com 8 neurônios que gera um valor de 1 a 8, onde cada valor inteiro representa uma classe.
-

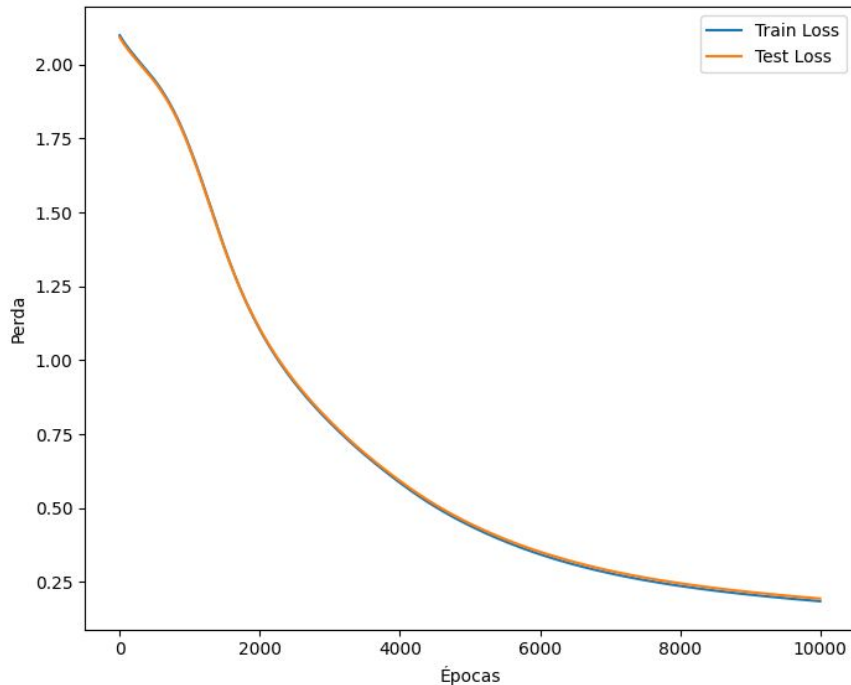
---

# Treinamento

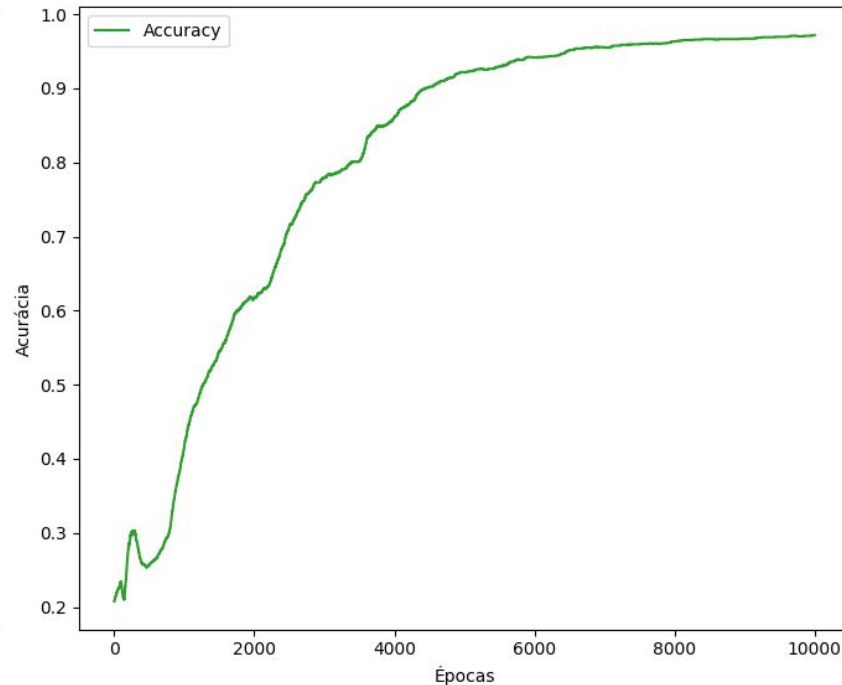
- Métrica de erro: CrossEntropyLoss;
  - Otimizador: SGD;
  - Conjunto de dados: Par ordenado aleatório com valores de -1 a 1 (10000 valores - 80% - Treino);
  - Função de ativação: ReLu nas camadas internas;
  - Épocas: 10000 com early stopping (0.0001 em 50 épocas);
  - Learning Rate: 0.01
-

# Treinamento sem momentum

Curva de Perda durante o Treinamento



Curva de Acurácia durante o Treinamento

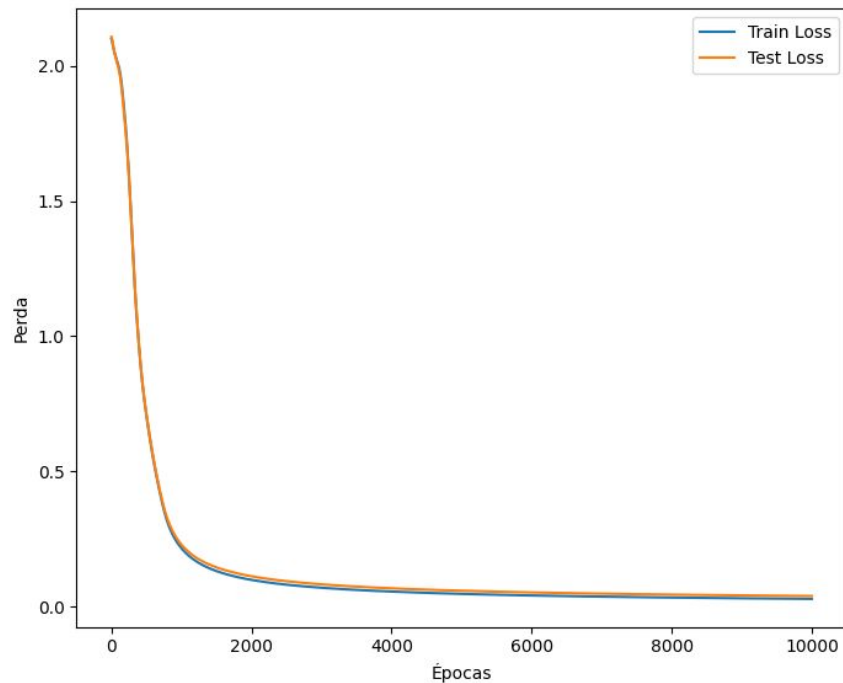




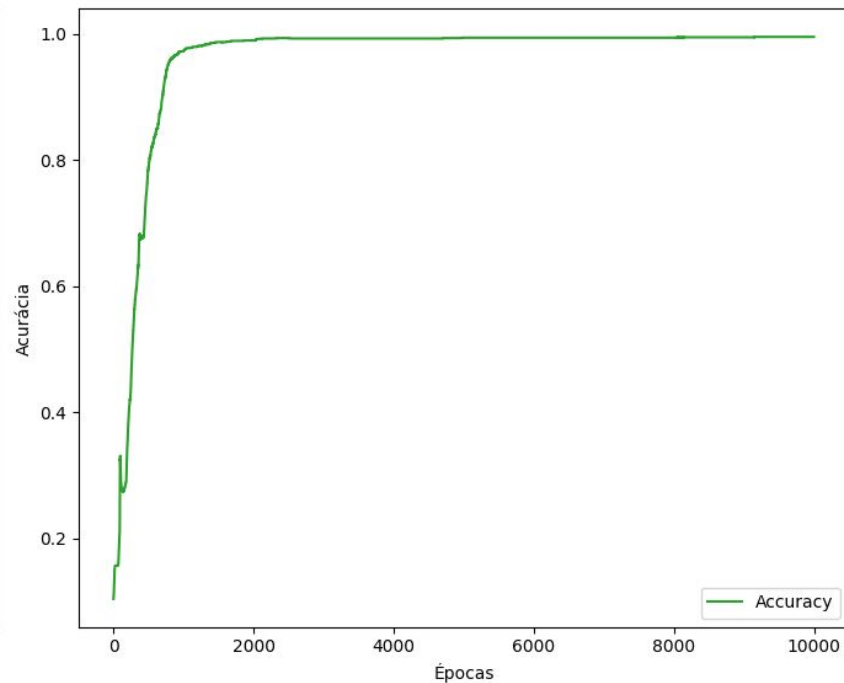
---

# Treinamento com momentum

Curva de Perda durante o Treinamento



Curva de Acurácia durante o Treinamento





---

## Arquitetura - Abordagem 2

- Camada de entrada para 2 valores (Par ordenado: x , y)
  - 2 camadas ocultas contendo 16 neurônios;
  - Saída com 8 neurônios que gera um vetor com 8 bits representando a ativação de cada classe (Ex: [01000000]).
-



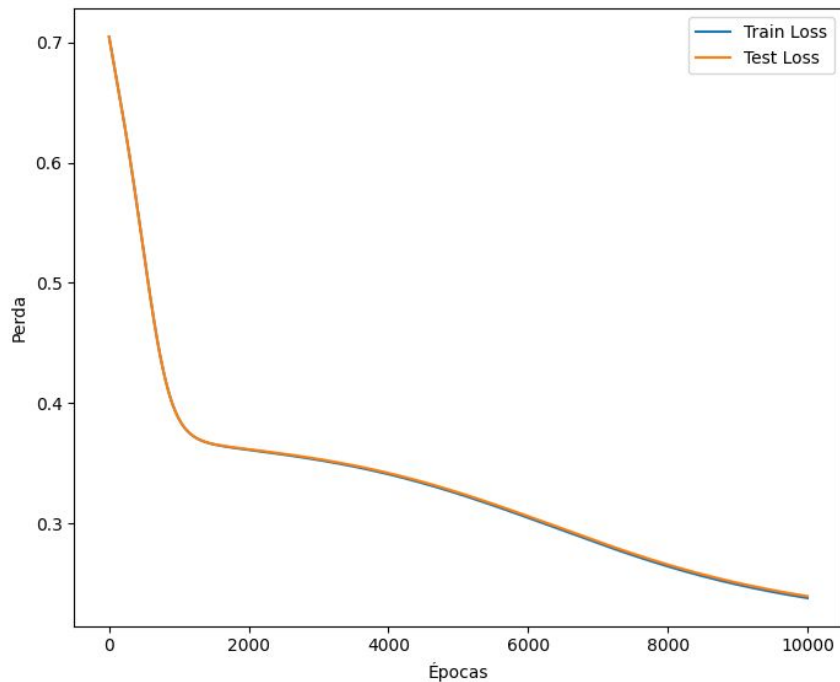
---

# Treinamento

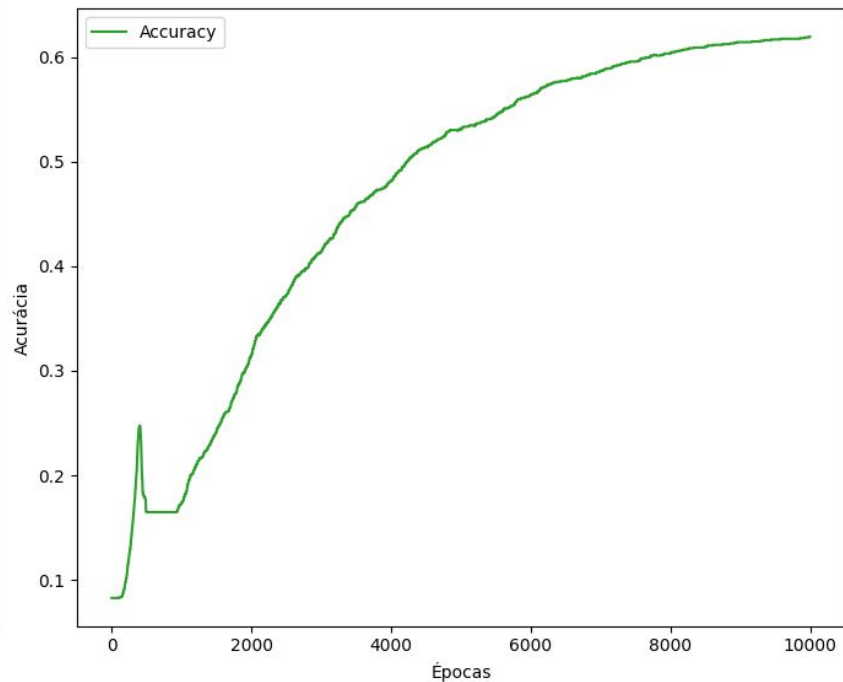
- Métrica de erro: BinaryCrossEntropyLoss;
  - Otimizador: SGD;
  - Encoding: OneHot
  - Conjunto de dados: Par ordenado aleatório com valores de -1 a 1 (10000 valores - 80% - Treino);
  - Função de ativação: ReLu nas camadas internas;
  - Épocas: 1000 com early stopping (0.0001 em 50 épocas);
  - Learning Rate: 0.01
-

# Treinamento sem momentum

Curva de Perda durante o Treinamento

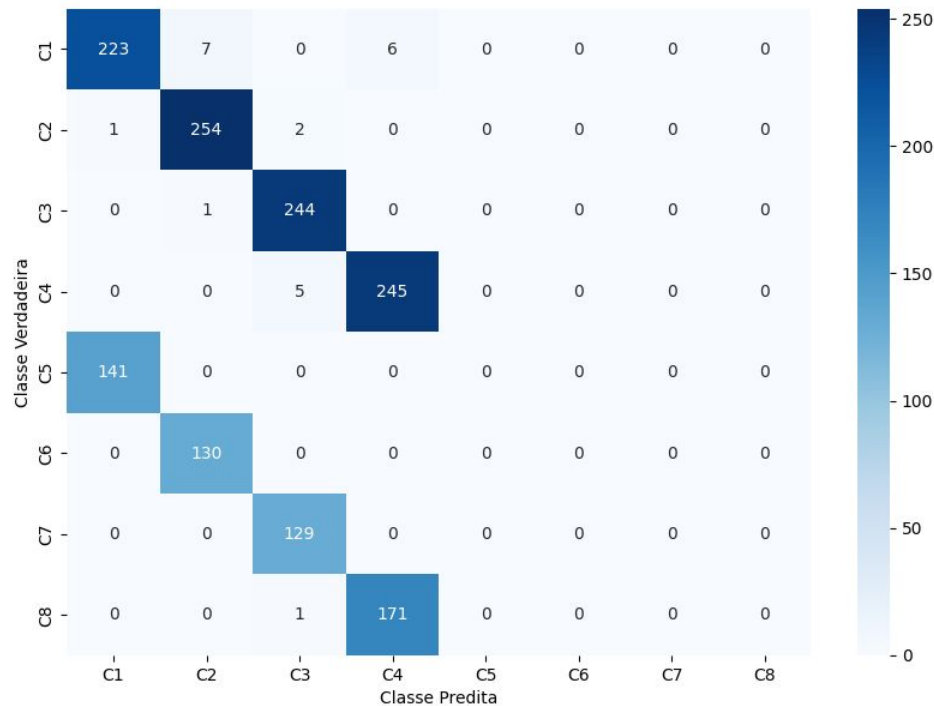


Curva de Acurácia durante o Treinamento



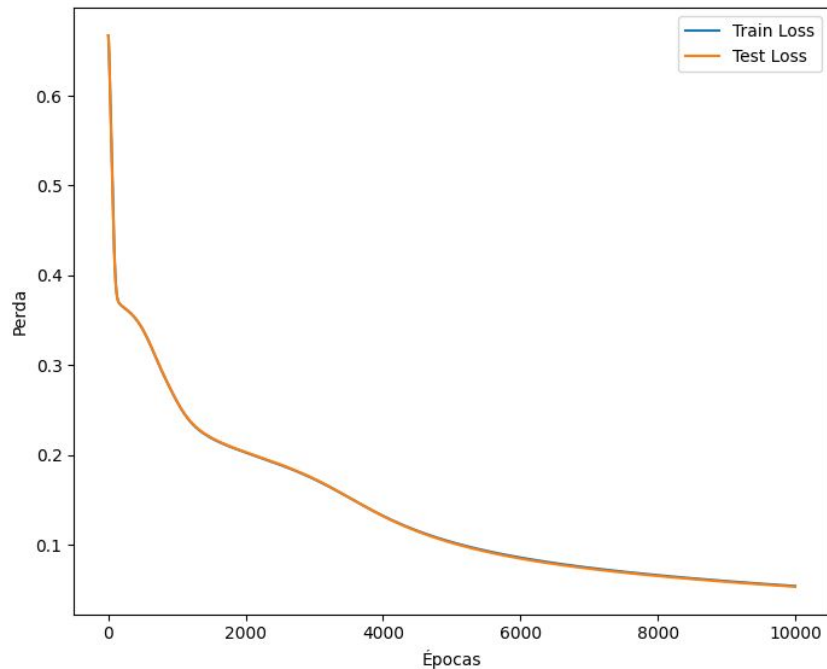
# Resultados Sem momentum

- Acurácia: 0.62
- Matriz de confusão:

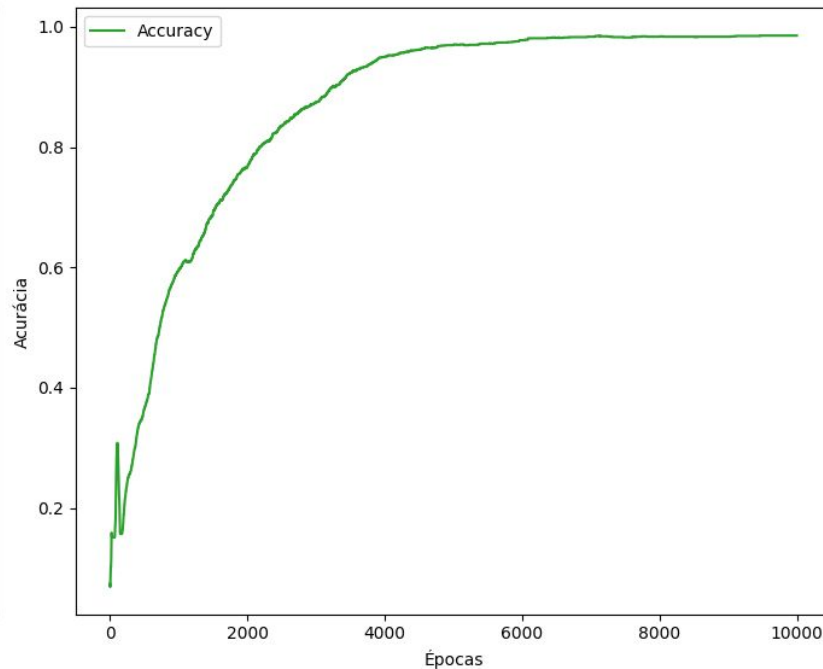


# Treinamento com momentum

Curva de Perda durante o Treinamento



Curva de Acurácia durante o Treinamento





---

## Questão 3

- Objetivo: Regressão e previsão de valores de uma função (Prever com K valores passados os 3 próximos valores);
  - Função:  $y(n) = 2 \cdot \sin(0.5 \cdot n) + \cos(1.5 \cdot n + (\pi/4))$
-

---

# Arquitetura

- Camada de entrada: 10 valores ( $K = 10$ );
  - 1 Camada oculta com 64 neurônios (Relu);
  - 1 Camada oculta com 32 neurônios (Relu);
  - Camada de saída com 3 neurônios;
-

---

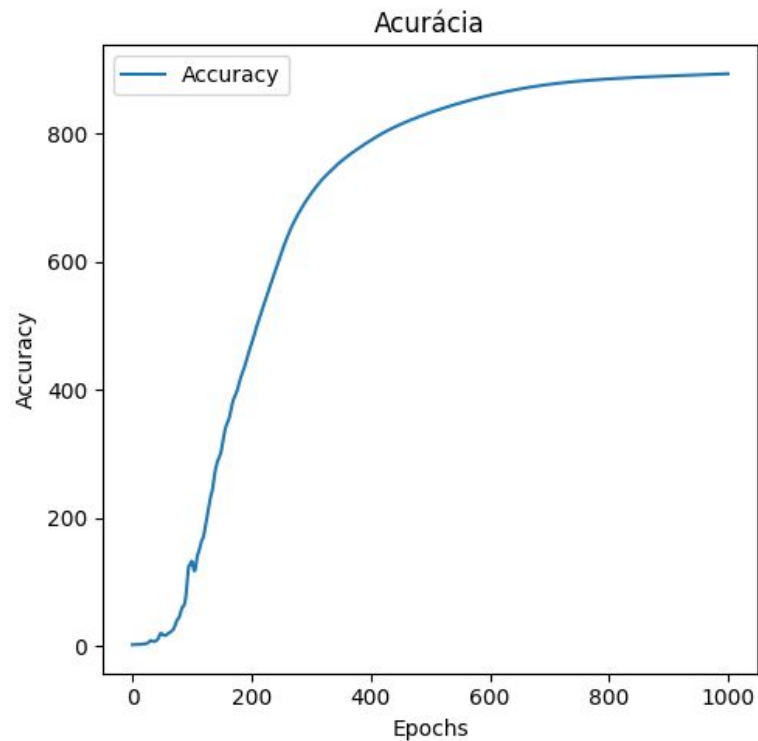
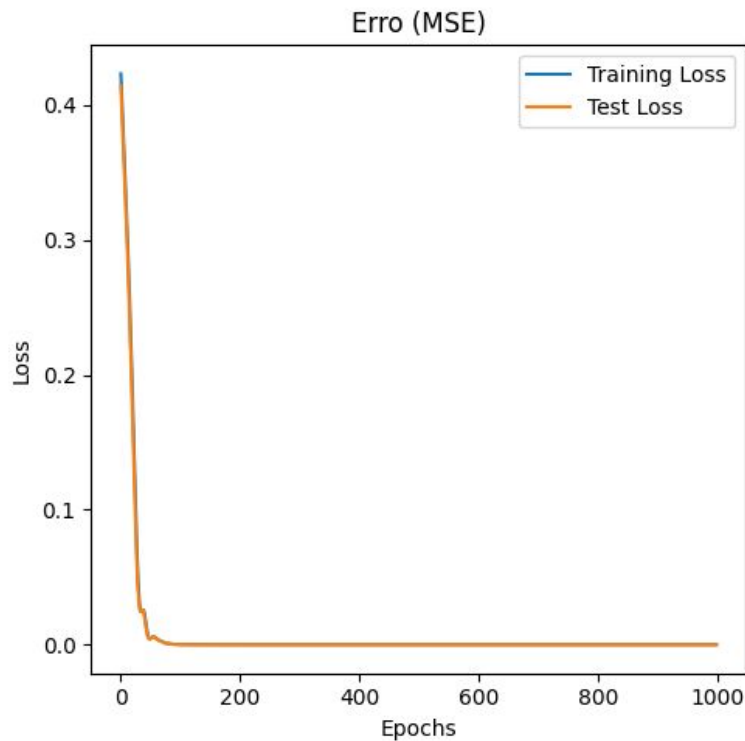
# Treinamento

- Métrica de erro: MSE;
  - Otimizador: Adam;
  - Conjunto de dados: valores resultantes da função (0, 200, 0.001);
  - Função de ativação: Relu;
  - Épocas: 1000 com early stopping (0.0001 em 50 épocas);
  - Learning Rate: 0.001
-



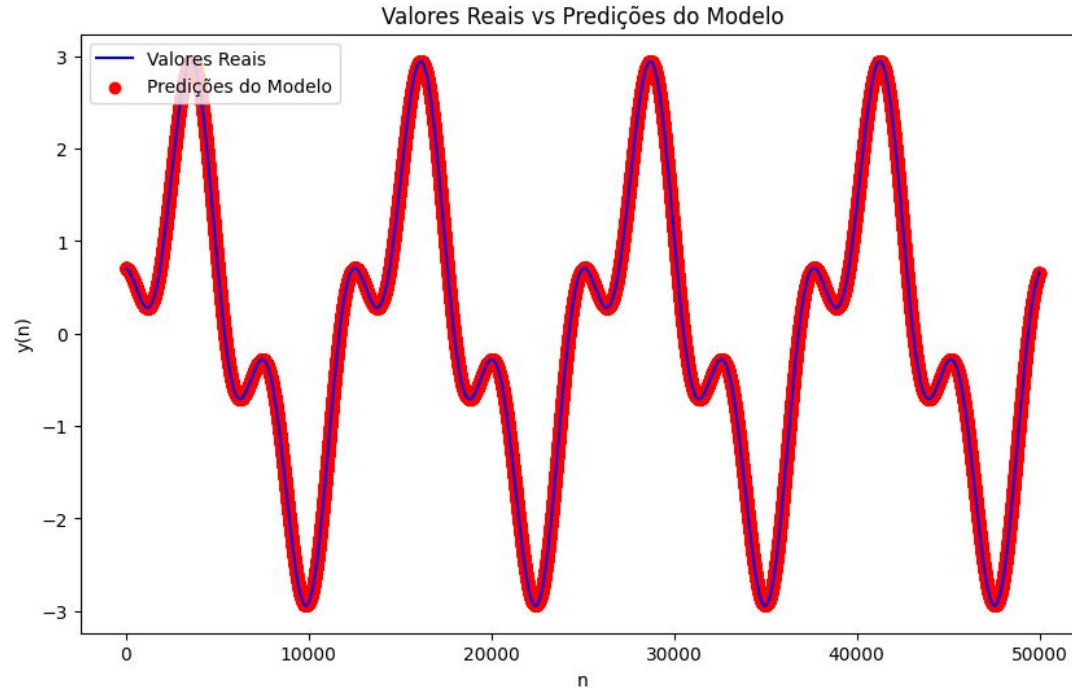
---

# Resultados do treinamento

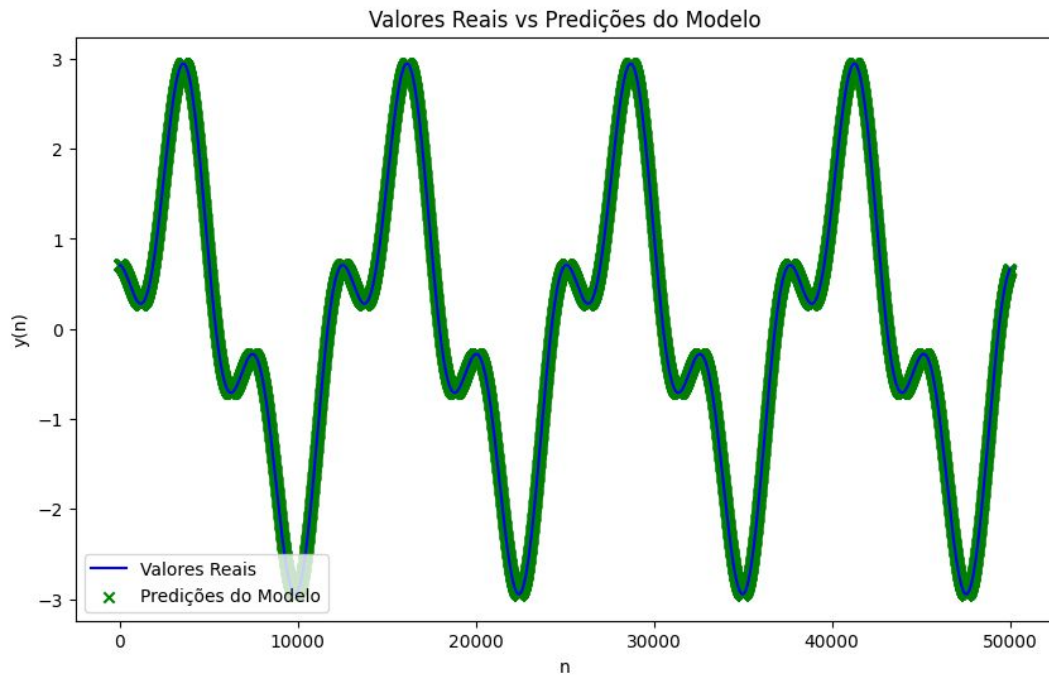


---

# Comparação Modelo x Valor Real (K+1)

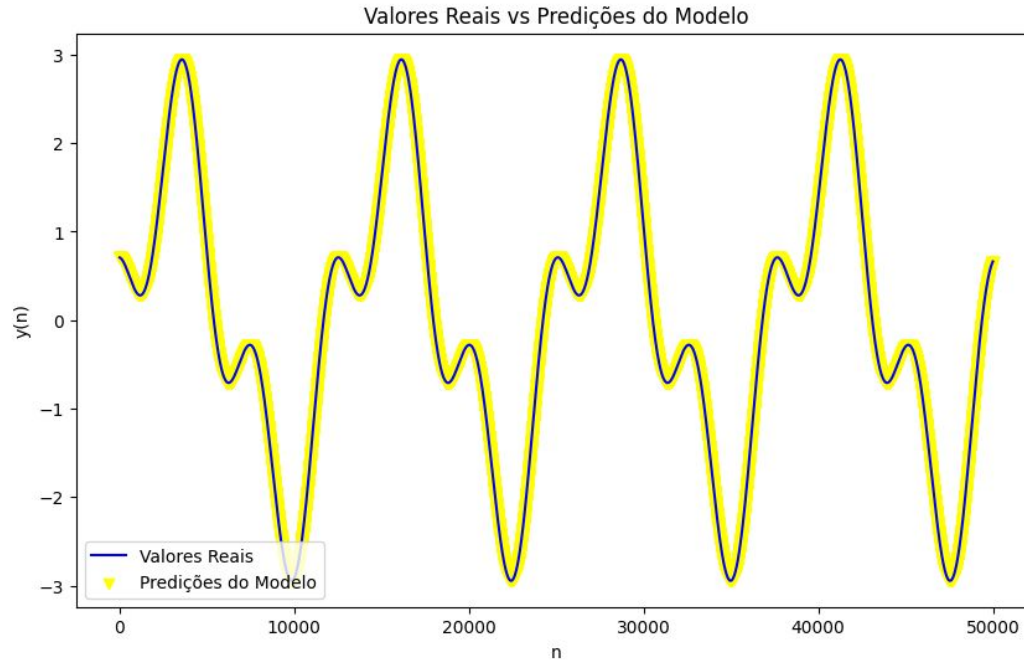


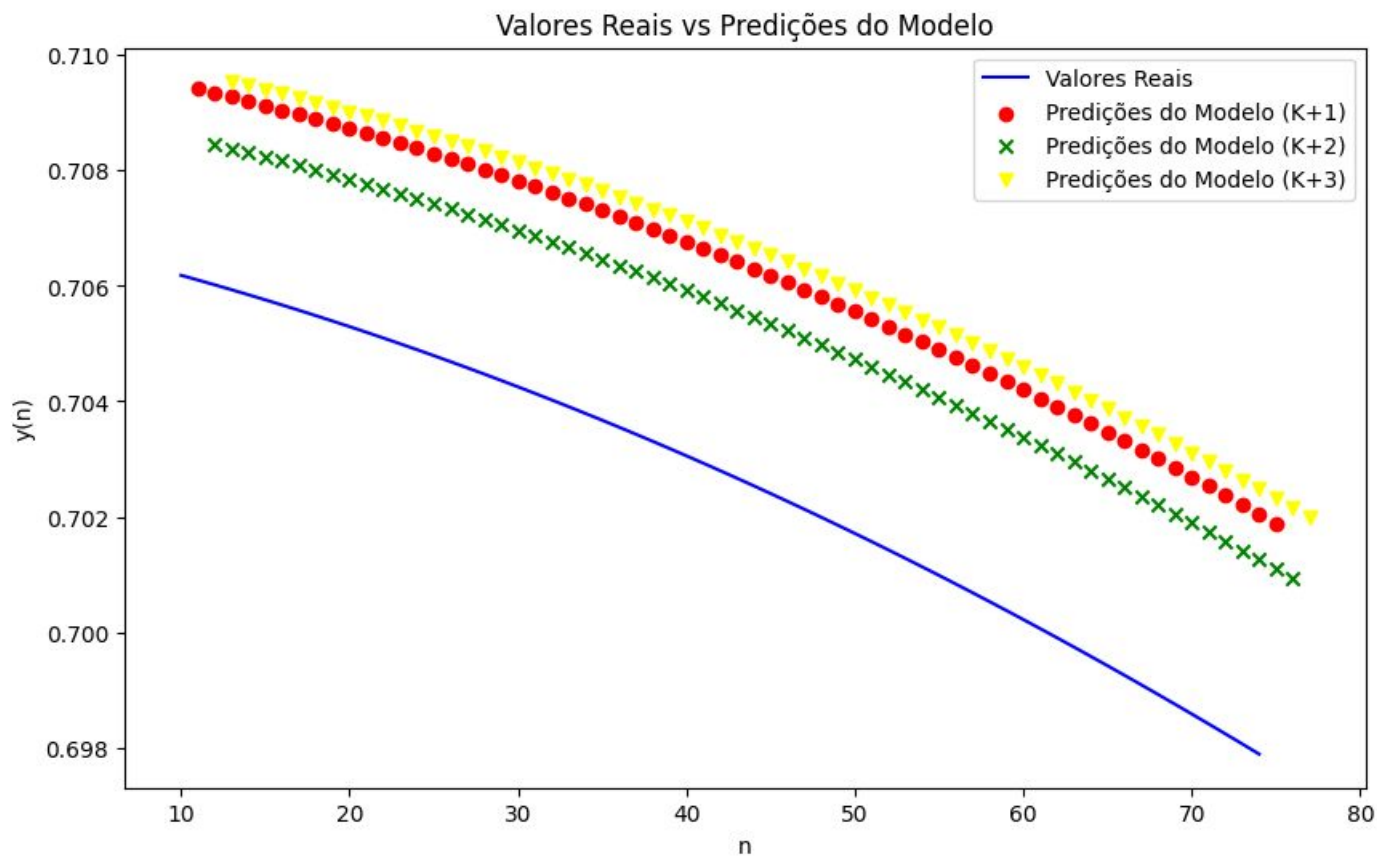
# Comparação Modelo x Valor Real (K+2)

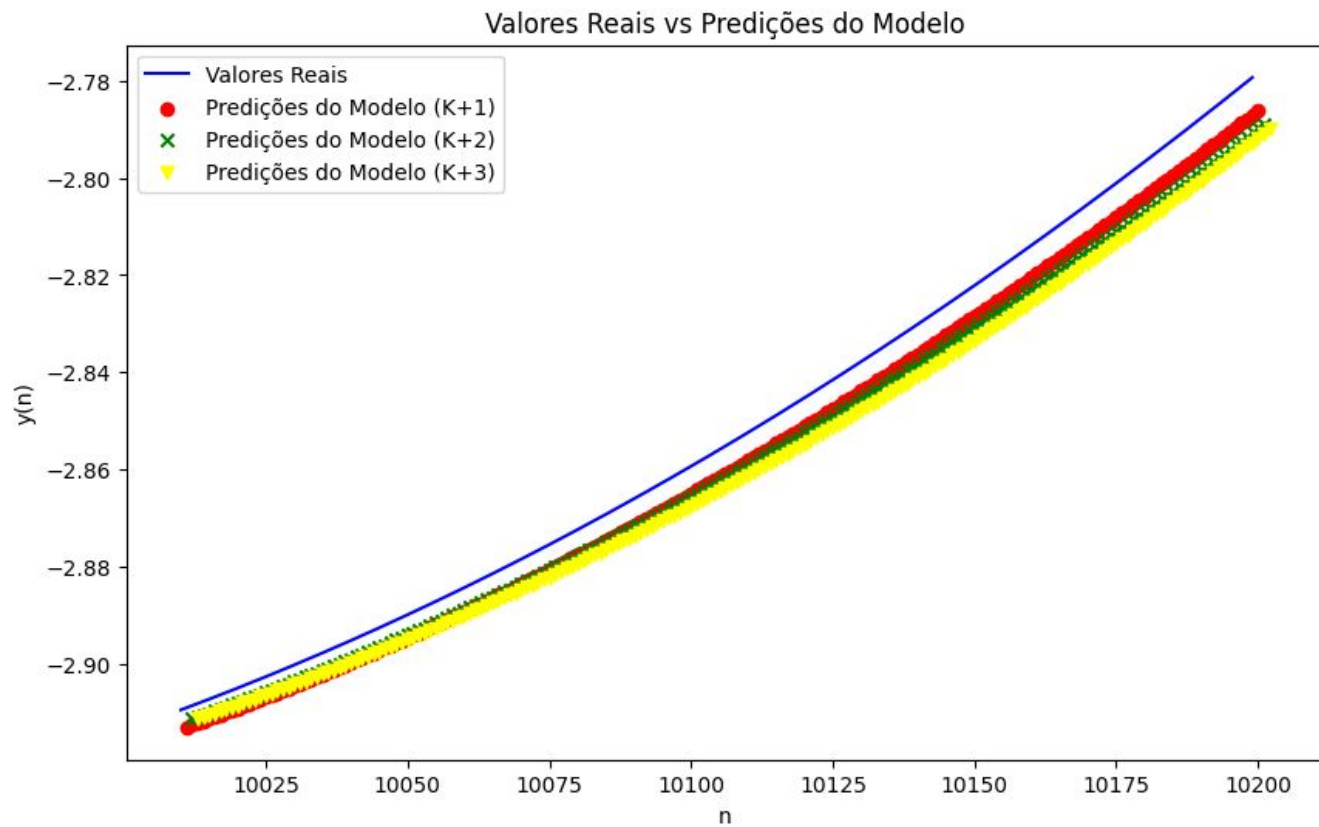


---

# Comparação Modelo x Valor Real (K+3)







---

## Questão 4 - Objetivo Principal

- Criar um modelo preditivo para determinar a probabilidade de sobrevivência de um passageiro no acidente do Titanic.
  - Implementar uma Rede Perceptron de Múltiplas Camadas (MLP) para realizar as previsões.
-

---

## Questão 4 - preparação do dados

- Contou-se o número de valores faltantes em cada coluna.
  - Identificou-se as colunas que contêm valores zerados e quantos valores zerados existem.
  - Criação de feature Family (SibSp + Parch).
  - Preencheu-se os valores faltantes na coluna Age com a média das idades.
  - Substituiu-se os valores zerados na coluna Fare pela mediana da coluna.
  - Converteu-se a coluna Age para o inteiro (int).
  - Mapeou-se a coluna Sex para valores numéricos: male para 0 e female para 1.
  - Preencheu-se os valores faltantes em Embarked com a moda.
  - Mapeou-se a coluna Embarked para valores numéricos: C para 1, Q para 2, e S para 3.
  - Removeu-se as colunas PassengerId, Name, Ticket, Cabin, SibSp, e Parch, pois não eram consideradas relevantes para a modelagem.
  - Padronização das features com StandardScaler.
-



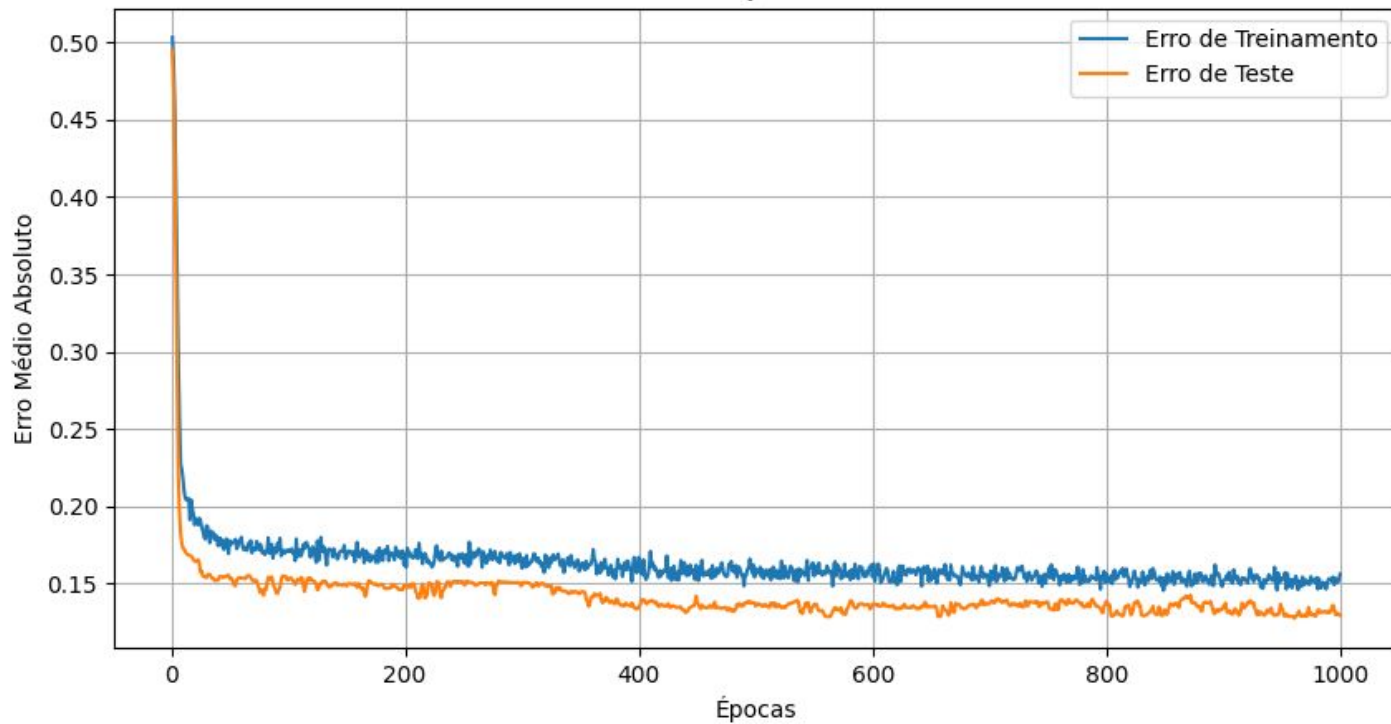
---

## Questão 4 - Criação da rede neural

- **Arquitetura com 3 camadas ocultas**
  - **Primeira Camada Oculta:** Camada com 64 neurônios, seguida por uma função de ativação ReLU e dropout com uma taxa de 30%
  - **Segunda Camada Oculta:** Camada com 32 neurônios, seguida por ReLU e dropout
  - **Terceira Camada:** com 16 neurônios, seguida por ReLU e dropout
  - Camada final com 1 neurônio, seguida por uma função de ativação Sigmoid
  - **dropout\_rate = 0.3:** Taxa de dropout usada para reduzir o overfitting.
  - **learning\_rate = 0.001:** Taxa de aprendizado do otimizador.
  - **num\_epochs = 1000:** Número de épocas para o treinamento.
  - **Otimização com Adam e regularização L2.**
-

## Questão 4

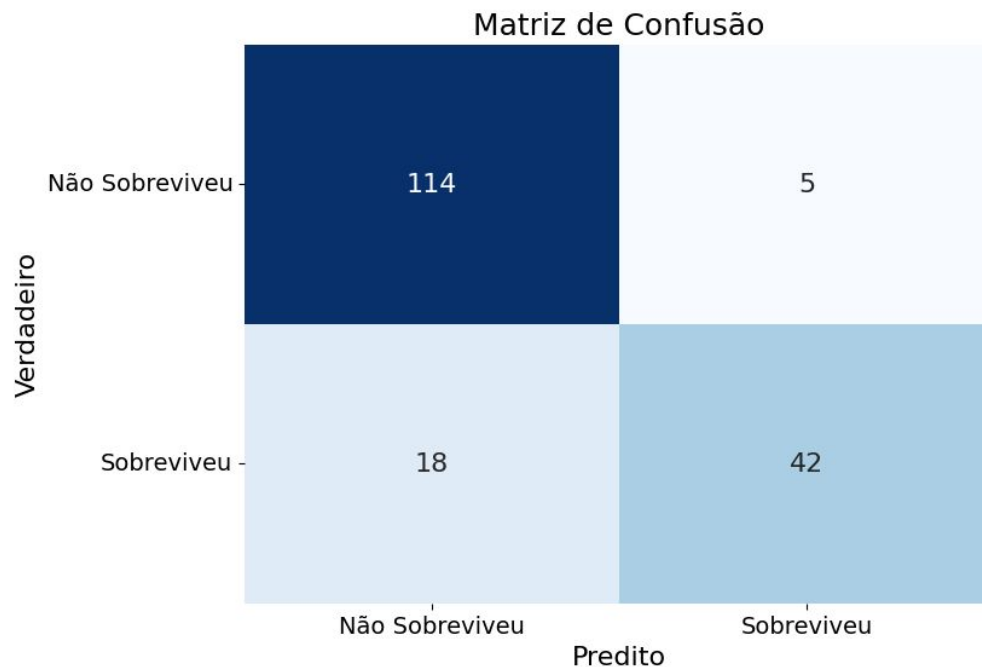
Curva do Erro Médio no Conjunto de Treinamento e Teste



## Questão 4

- Acurácia: 0.8715
- Matriz de confusão:

Relatório de Classificação:				
	precision	recall	f1-score	support
0.0	0.86	0.96	0.91	119
1.0	0.89	0.70	0.79	60
accuracy			0.87	179
macro avg	0.88	0.83	0.85	179
weighted avg	0.87	0.87	0.87	179



## Questão 4

- Acurácia: 0.8497
- Matriz de confusão:

Relatório de Classificação - Treinamento:				
	precision	recall	f1-score	support
0.0	0.84	0.93	0.88	430
1.0	0.87	0.73	0.79	282
accuracy			0.85	712
macro avg	0.85	0.83	0.84	712
weighted avg	0.85	0.85	0.85	712

