

# Relatório IA

## Problem 4: 0 or 1?

PL1 - Grupo 02

Andreia Qiu 79856

Eduarda Pereira 79749

Guilherme Carmo 79860



## Descrição do Problema e Algoritmo Utilizado

O problema consiste em desenvolver uma rede neuronal para classificar imagens em escala de cinzentos (20x20 pixels) de dígitos manuscritos como 0 ou 1. O conjunto de dados consiste em 800 amostras divididas em características (400 valores que representam intensidades de pixel normalizadas para o intervalo  $[0,1]$ ) e rótulos (0 ou 1). A tarefa utiliza uma abordagem de aprendizagem supervisionada, exigindo que a rede aprenda um mapeamento entre as imagens de entrada e as suas etiquetas binárias.

O algoritmo utilizado é uma rede neural feedforward com as seguintes características:

Camada de entrada: 400 neurónios correspondentes aos 400 valores de pixel de cada imagem.

Camada(s) oculta(s): o número e o tamanho foram ajustados para otimizar o desempenho.

Camada de saída: um único neurónio com uma função de ativação sigmoide, que produz um valor entre 0 e 1, o qual é limiarizado para classificar como 0 ou 1.

Função de ativação: Sigmoide (padrão).

Treino: o modelo utiliza a descida gradiente para minimizar a perda do erro quadrático médio (MSE) entre as etiquetas previstas e as verdadeiras. O conjunto de dados é dividido em subconjuntos de treino (80%) e de teste (20%) para evitar o sobreajuste. A paragem antecipada é implementada para interromper o treino quando o desempenho da validação atinge um patamar.

## Opções de design

Foram realizadas experiências iniciais para determinar o número ideal de camadas ocultas e de neurónios por camada. Foram testadas várias configurações (por exemplo, uma camada oculta vs. várias camadas) para obter um equilíbrio entre a complexidade do modelo e o desempenho.

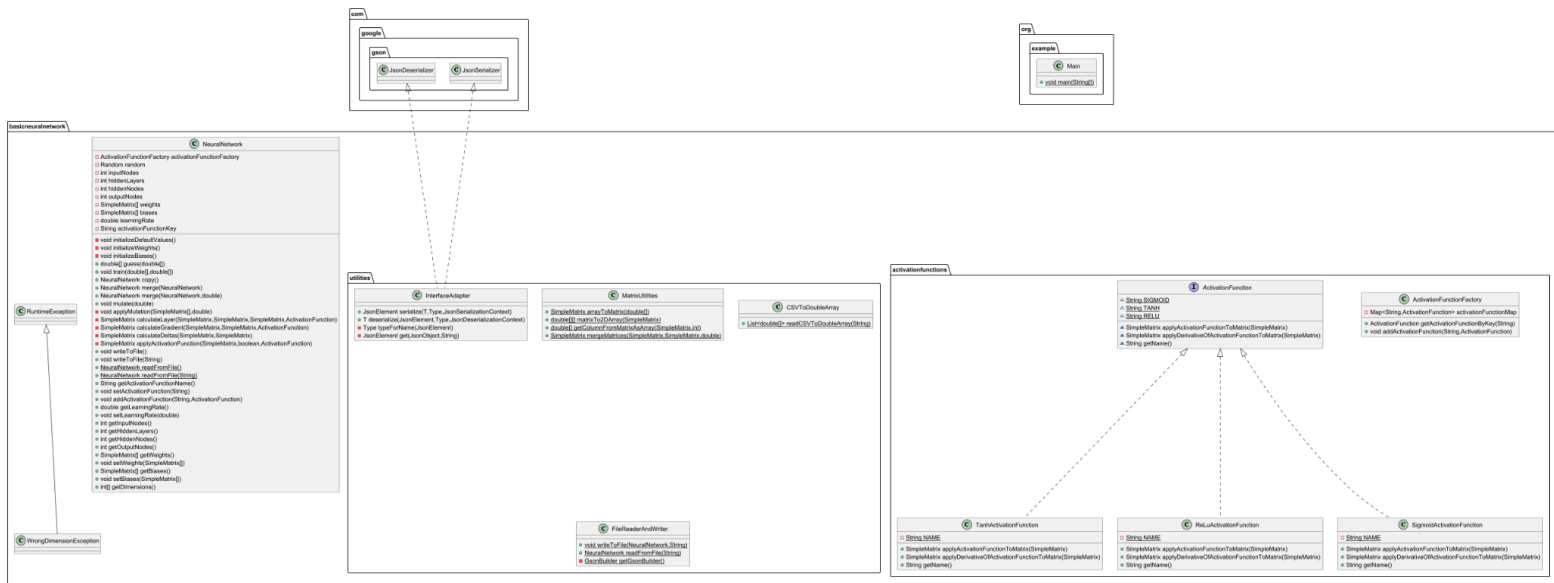
Função de ativação: a ativação sigmoide foi escolhida para todos os neurónios devido à sua compatibilidade com tarefas de classificação binária.

Pré-processamento de entrada: os valores dos pixels foram normalizados para o intervalo  $[0,1]$  para melhor convergência durante o treinamento.

Mitigação de sobreajuste: foi implementada uma paragem precoce para interromper o treino caso a perda de validação deixe de melhorar. O conjunto de dados foi dividido em conjuntos de treino e teste (80% / 20%) para validar o desempenho do modelo em dados não vistos.

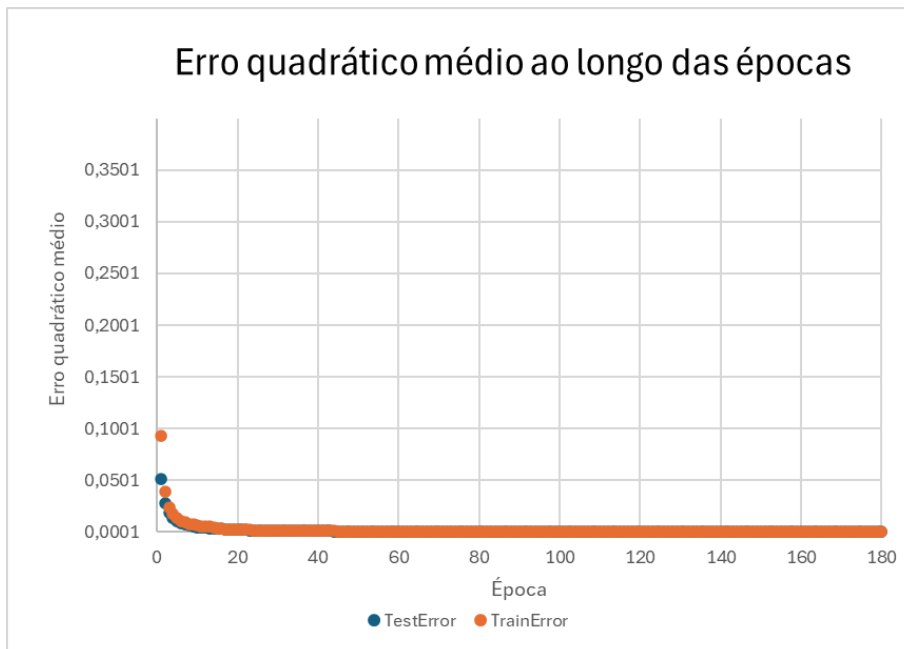
Hiperparâmetros: a taxa de aprendizagem, o tamanho do lote e o número de épocas foram ajustados para minimizar o MSE no conjunto de validação.

## Diagrama UML



## Resultados, análise e discussão

### Resultados:



### Análise:

Os resultados demonstram que a arquitetura escolhida foi eficaz para esta tarefa de classificação binária, uma vez que o MSE de treino diminuiu gradualmente ao longo das épocas, indicando que o modelo está a aprender com os dados de treino.

Da mesma forma, o MSE de teste diminuiu, mas com uma tendência mais estável, o que sugere que o modelo está a generalizar bem para dados não vistos.

No geral, o modelo apresentou bom desempenho, com o MSE de Treino e Teste diminuindo consistentemente ao longo das épocas. A diferença entre o MSE de Treino e de Teste é pequena, o que indica que o modelo não sofre de overfitting (excesso de ajuste aos dados de treino). O MSE de Teste segue uma tendência semelhante ao MSE de Treino, mas com valores ligeiramente maiores.

O modelo praticamente converge após aproximadamente 180 épocas, com valores de MSE muito baixos (próximos de 0.00002 para ambos os conjuntos). A convergência rápida e estável indica que o modelo foi bem ajustado e que os hiperparâmetros escolhidos (taxa de aprendizagem, número de camadas, etc.) foram apropriados. A normalização dos dados para o intervalo  $[0, 1]$  foi crucial para o bom desempenho do modelo, garantindo que os valores de entrada fossem tratados de forma uniforme pela rede neural.

#### Discussão:

O modelo não apresentou sinais de overfitting, pois o MSE de teste não aumentou mesmo após várias épocas. Também não houve underfitting, já que o MSE de treino diminuiu significativamente ao longo do tempo. A escolha de uma rede neural com uma única camada oculta e função de ativação sigmóide mostrou-se eficaz para o problema em questão. Aumentar o número de camadas ou neurónios poderia melhorar o desempenho, no entanto também aumentaria a complexidade do modelo e o tempo de treino.

## Conclusões

O modelo de rede neural teve um desempenho satisfatório na tarefa de distinguir entre os dígitos 0 e 1, obtendo um erro quadrático médio (MSE) bastante baixo para os dados de treino e de teste. A normalização dos dados e a escolha de uma arquitetura simples foram decisões cruciais que contribuíram para o sucesso do modelo. O processo sublinhou a importância da normalização dos dados e de uma divisão adequada entre os conjuntos de treino e de teste para obter um desempenho fiável. Além disso, a tarefa realçou a importância da afinação de hiperparâmetros e da experimentação sistemática no desenvolvimento de modelos de aprendizagem automática. Embora o modelo atual tenha provado ser eficaz, existe potencial para uma maior otimização através da exploração de técnicas adicionais, como a regularização, o abandono ou arquiteturas alternativas.

## Referências Webgráficas

[https://github.com/kim-marcel/basic\\_neural\\_network](https://github.com/kim-marcel/basic_neural_network)