

# Inteligência Artificial para Robótica Móvel CT-213

Instituto Tecnológico de Aeronáutica

## Relatório do Laboratório 7 - *Imitation Learning* com Keras

Leonardo Peres Dias

12 de maio de 2025



# Sumário

<b>1</b>	<b>Breve Explicação em Alto Nível da Implementação</b>	<b>3</b>
<b>2</b>	<b>Figuras Comprovando Funcionamento do Código</b>	<b>4</b>
2.1	Função de Classificação <code>sum_gt_zeros</code> . . . . .	4
2.1.1	Sem regularização . . . . .	4
2.1.2	Com regularização . . . . .	5
2.2	Função de Classificação XOR . . . . .	6
2.2.1	Sem regularização . . . . .	6
2.2.2	Com regularização . . . . .	7
2.3	<i>Imitation Learning</i> . . . . .	8
<b>3</b>	<b>Discussões</b>	<b>8</b>
3.1	Regularização . . . . .	8
3.2	<i>Imitation Learning</i> . . . . .	8

# 1 Breve Explicação em Alto Nível da Implementação

A rede neural foi implementada utilizando a API Keras. Nesta implementação, foi utilizado um modelo feedforward composto por três camadas densas. A primeira camada possui 75 neurônios com ativação linear e recebe um vetor de entrada unidimensional. Em seguida, aplica-se a função de ativação Leaky ReLU com um parâmetro de inclinação de 0.01.

Posteriormente, a segunda camada densa, composta por 50 neurônios, também utiliza uma ativação linear seguida de uma camada Leaky ReLU. Por fim, a terceira camada densa, com 20 neurônios sem a aplicação de uma função de ativação adicional (ativação linear).

O modelo utiliza o otimizador Adam com o objetivo de minimizar a função de custo definida pelo erro quadrático médio (MSE). Durante o treinamento, o conjunto de dados é fornecido ao modelo em um único batch correspondentes ao tamanho total do dataset.

## 2 Figuras Comprovando Funcionamento do Código

### 2.1 Função de Classificação `sum_gt_zeros`

#### 2.1.1 Sem regularização

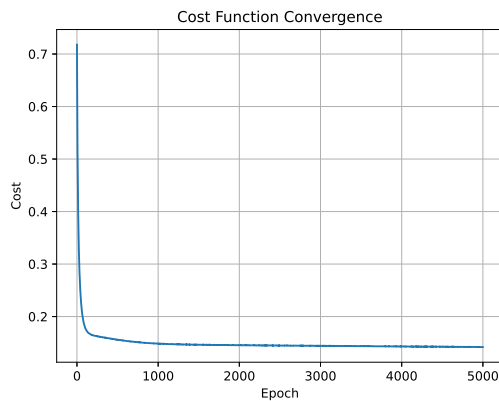


Figura 1: Convergência do custo

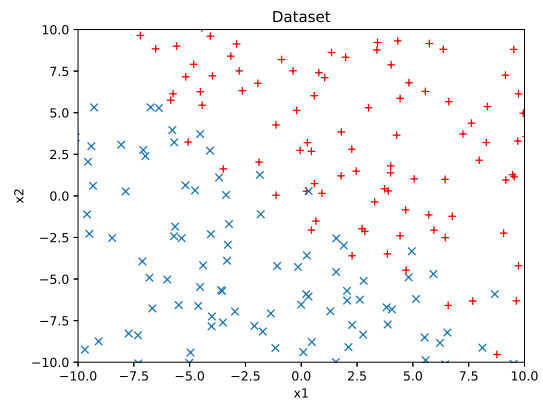


Figura 2: Dataset

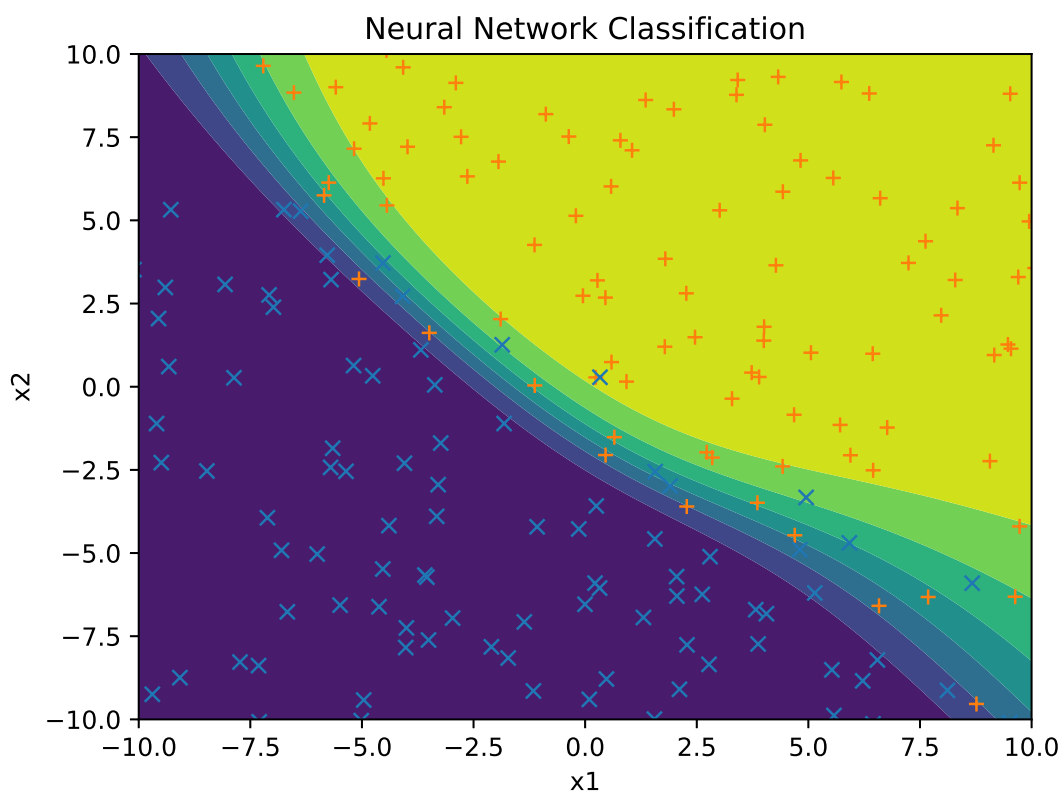


Figura 3: Tarefa de classificação

## 2.1.2 Com regularização

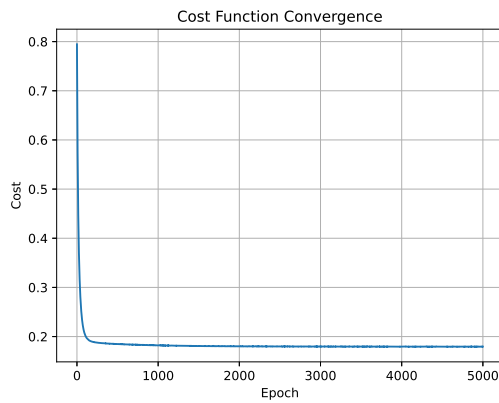


Figura 4: Convergência do custo

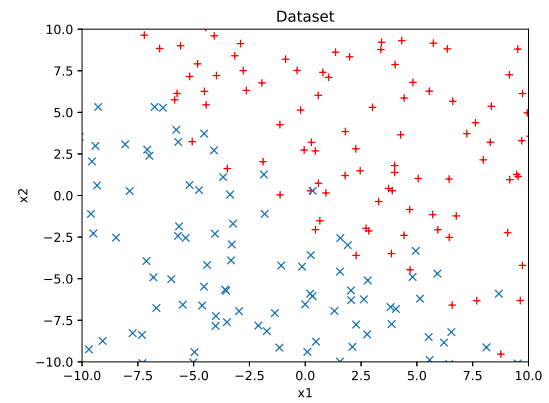


Figura 5: Dataset

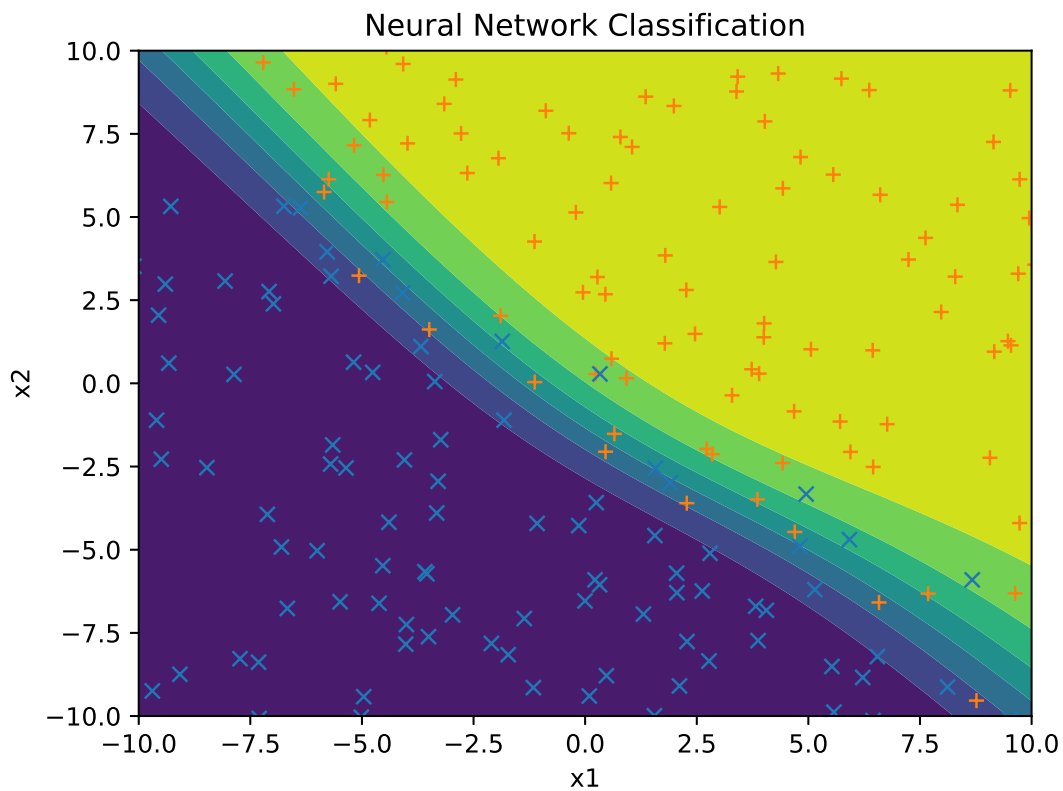


Figura 6: Tarefa de classificação

## 2.2 Função de Classificação XOR

### 2.2.1 Sem regularização

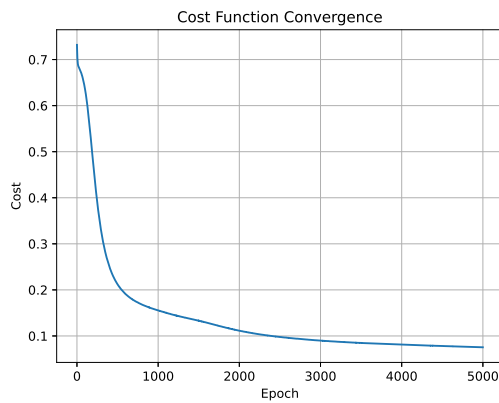


Figura 7: Convergência do custo

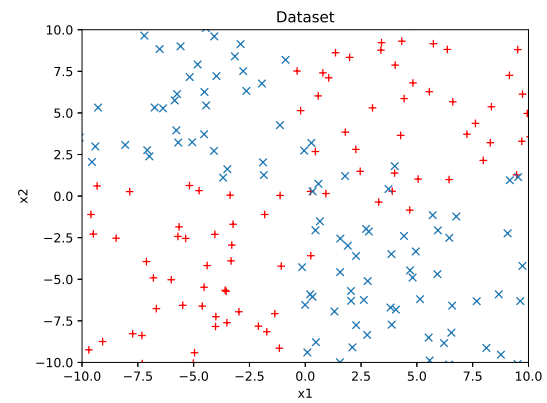


Figura 8: Dataset

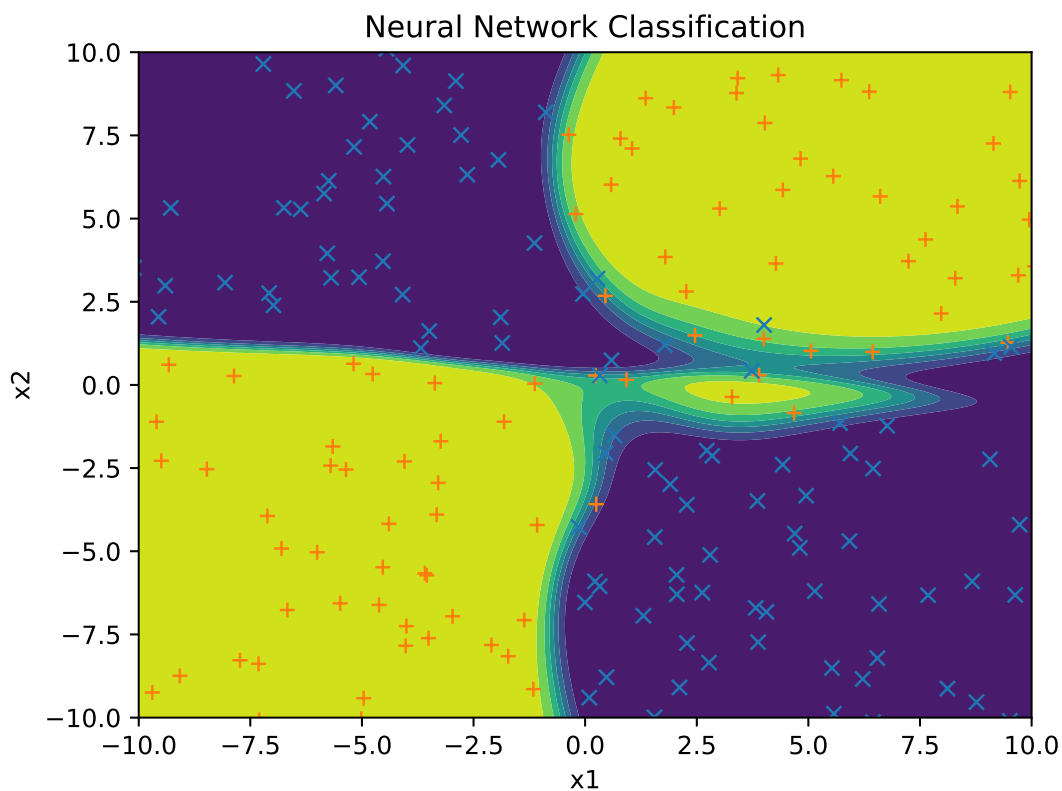


Figura 9: Tarefa de classificação

## 2.2.2 Com regularização

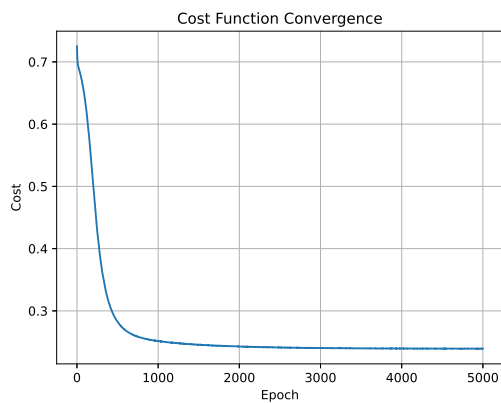


Figura 10: Convergência do custo

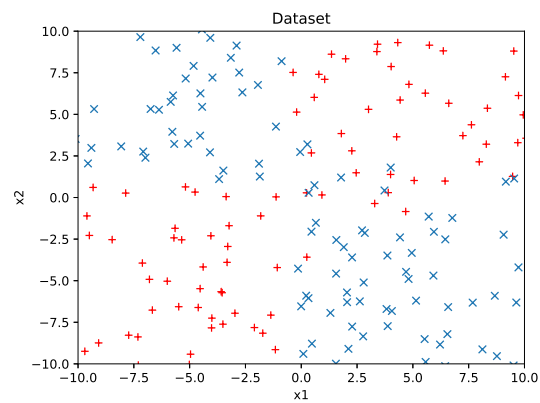


Figura 11: Dataset

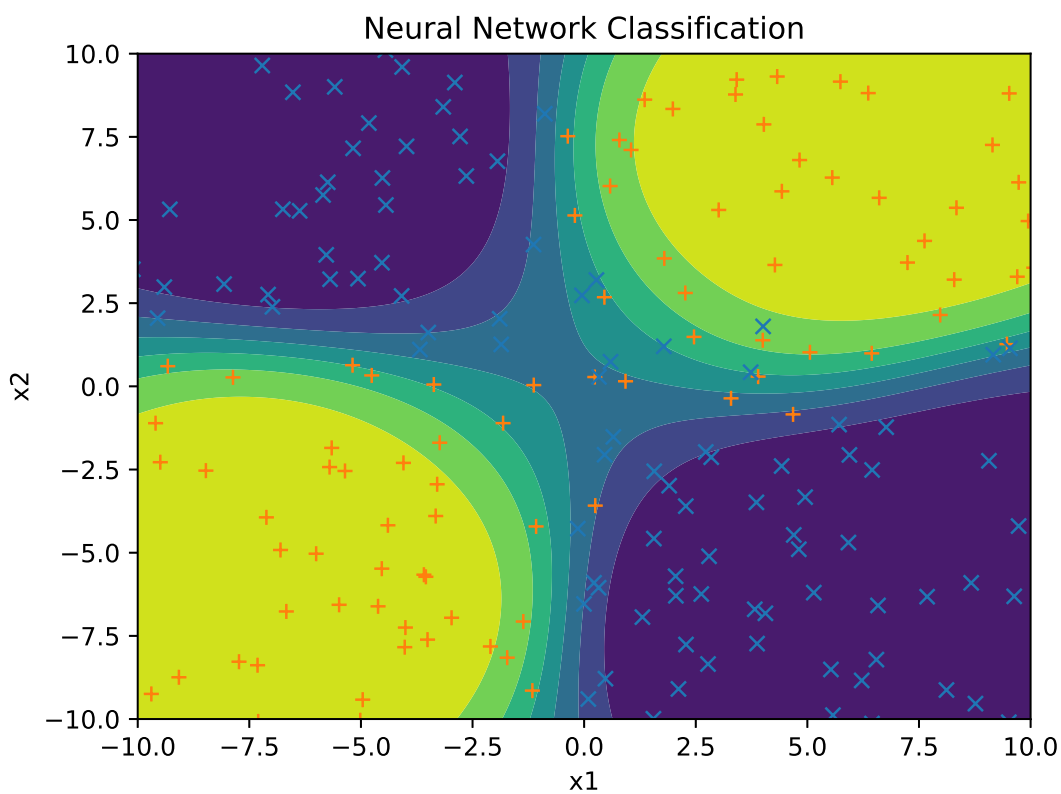


Figura 12: Tarefa de classificação

## 2.3 Imitation Learning

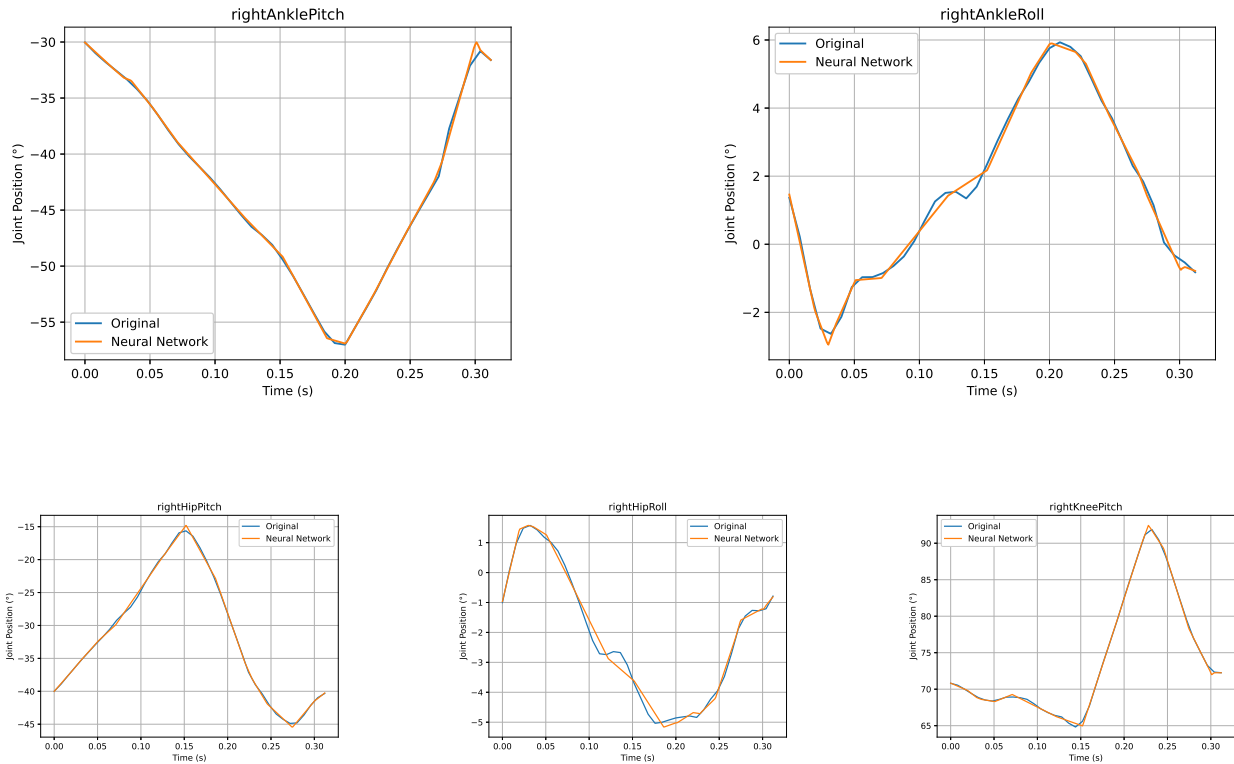


Figura 13: Movimento das juntas do robô obtidas pela rede neural.

## 3 Discussões

### 3.1 Regularização

Nota-se que o emprego de regularização L2 na tarefa de classificação `sum_gt_zeros` não trouxe resultados relevantes, uma vez que o dataset é linearmente separável. No entanto, na tarefa de classificação XOR, a regularização L2 demonstrou um desempenho superior em comparação à ausência de regularização, visto que previniu a rede de realizar *overfitting* e, assim, perder poder de generalização.

### 3.2 Imitation Learning

A rede neural foi capaz de aprender o movimento, reproduzindo o movimento das juntas do robô. A comparação entre os movimentos reais e os previstos pela rede neural mostra que a rede conseguiu capturar o padrão de movimento desejado, embora com algumas variações.