

# Lab 01 - Introdução à inteligência artificial e conceitos básicos

---

## Enunciados

Ex 1: Decisões com `if`, `else`

Ex 2: Fórmula Simples de "Machine Learning"

Ex 3: Neurónio Artificial Simples

## Solução

Sol 1: Decisões com `if`, `else`

Sol 2: Fórmula Simples de "Machine Learning"

Sol 3: Neurónio Artificial Simples

## Enunciados

### Ex 1: Decisões com `if`, `else`

#### Enunciado:

Escreva um programa em Python que peça ao utilizador a sua idade e o seu rendimento mensal. O programa deve então determinar se o utilizador é elegível para um empréstimo, considerando que os critérios de elegibilidade são:

- Ter pelo menos 18 anos de idade.
- Ter um rendimento mensal de pelo menos 1000 euros.

O programa deve imprimir uma mensagem informando se o utilizador é ou não elegível para o empréstimo.

### Ex 2: Fórmula Simples de Machine Learning

#### Enunciado:

Crie um programa em Python que utilize uma fórmula matemática simples para estimar a probabilidade de um cliente pagar um empréstimo, com base na sua idade e rendimento mensal. A fórmula deve ser a seguinte:

- $\text{probabilidade\_pagamento} = 0.01 * \text{idade} + 0.0005 * \text{rendimento} - 0.5$

O programa deve pedir ao utilizador a sua idade e o seu rendimento mensal, calcular a probabilidade de pagamento usando a fórmula e imprimir o resultado em percentagem.

### Ex 3: Neurónio Artificial Simples

#### Enunciado:

Implemente um programa em Python que simule um neurónio artificial simples para avaliar se um cliente é um bom candidato para um empréstimo, com base na sua idade e rendimento mensal. O neurónio deve utilizar os seguintes pesos e bias:

- $\text{peso\_idade} = 0.01$

- `peso_rendimento = 0.0005`
- `bias = -0.5`

O programa deve pedir ao utilizador a sua idade e o seu rendimento mensal, calcular a ativação do neurónio e, com base no resultado, imprimir se o cliente é ou não um bom candidato para o empréstimo. Lembre-se que a função de ativação deve retornar 1 se a ativação for maior que 0, e 0 caso contrário.

**Observação:** Em todas as partes, lembre-se que a idade mínima para elegibilidade ao empréstimo é de 18 anos e o rendimento mínimo é de 1000 euros.

## Solução

### Sol 1: Decisões com **if, else**

1. **Cenário:** Um sistema simples que decide se um cliente é elegível para um empréstimo com base na sua idade e rendimento.
2. **Código:**

```
idade = int(input("Qual a sua idade? "))
rendimento = float(input("Qual o seu rendimento mensal? "))

if idade >= 18 and rendimento >= 1000:
    print("Parabéns! Você é elegível para o empréstimo.")
else:
    print("Infelizmente, você não atende aos critérios para o empréstimo.")
```

3. **Explicação:** Este código demonstra como a IA pode tomar decisões básicas usando lógica condicional, mas é limitado a regras fixas.

### Sol 2: Fórmula Simples de Machine Learning

1. **Cenário:** Prever a probabilidade de um cliente pagar um empréstimo com base na sua idade e rendimento, usando uma fórmula linear simples.
2. **Código:**

```
idade = int(input("Qual a sua idade? "))
rendimento = float(input("Qual o seu rendimento mensal? "))

# Fórmula simplificada (na realidade, o ML usaria dados reais para aprender esta relação)
probabilidade_pagamento = 0.01 * idade + 0.0005 * rendimento - 0.5

print("A probabilidade de pagamento do empréstimo é de:", probabilidade_pagamento * 100, "%")
```

3. **Explicação:** Este exemplo ilustra como um modelo de ML pode usar uma fórmula (aprendida a partir de dados) para estimar a probabilidade de um evento, permitindo uma avaliação mais nuanced do risco.

### Sol 3: Neurónio Artificial Simples

1. **Cenário:** Decidir se um cliente é um bom candidato para um empréstimo com base na sua idade e rendimento, usando um neurónio artificial.
2. **Código:**

```
def neuronio(entradas, pesos, bias):
    ativacao = sum([entrada * peso for entrada, peso in zip(entradas, pesos)]) +
    bias
    return 1 if ativacao > 0 else 0

idade = int(input("Qual a sua idade? "))
rendimento = float(input("Qual o seu rendimento mensal? "))

entradas = [idade, rendimento]
pesos = [0.01, 0.0005] # Pesos ajustados manualmente (o aprendizado ajustaria
estes pesos)
bias = -0.5

resultado = neuronio(entradas, pesos, bias)
print("Você é um bom candidato para o empréstimo" if resultado == 1 else "Você não
é um bom candidato para o empréstimo")
```

3. **Explicação:** Este código demonstra como um neurónio artificial pode ser usado para tomar decisões binárias, mas a rede precisaria de mais neurónios e camadas para considerar fatores adicionais e fazer previsões mais precisas.