

# Desafio

Você faz parte de uma equipe que está desenvolvendo modelos de Machine Learning para identificar a probabilidade de inadimplência em empréstimos concedidos por uma instituição financeira. Após treinar os modelos, sua tarefa é avaliar seu desempenho usando algumas métricas de avaliação. Nesse contexto, o desafio é criar um algoritmo que receba  $n$  matrizes de confusão e retorne o índice, precisão e acurácia da matriz que apresenta o melhor desempenho com base no cálculo dessas métricas. Lembrando que:

- **Acurácia é calculada pela fórmula:**  $(VP + VN) / (VP + FP + FN + VN)$
- **Precisão é calculada pela fórmula:**  $VP / (VP + FP)$

Onde:

- **VP (Verdadeiro Positivo):** Casos em que o modelo previu corretamente a classe positiva.
- **FP (Falso Positivo ou Erro Tipo I):** Casos em que o modelo previu incorretamente a classe positiva.
- **FN (Falso Negativo ou Erro Tipo II):** Casos em que o modelo previu incorretamente a classe negativa.
- **VN (Verdadeiro Negativo):** Casos em que o modelo previu corretamente a classe negativa.

## Entrada:

A entrada consiste em uma string composta por:  $n$ , representando o número de matrizes de confusão, seguido dos valores que compõem as  $n$  matrizes.

Cada matriz consiste em quatro valores, onde os dois primeiros representam a primeira linha da matriz, composta por verdadeiros positivos (VP) e falsos positivos (FP); os dois últimos valores representam a segunda linha, que é composta por falsos negativos (FN) e verdadeiros negativos (VN). As duas linhas e os valores que as compõem estão separados por vírgulas.

## Saída:

O resultado esperado inclui o valor do índice, acurácia e precisão (arredondada para duas casas decimais) da matriz com melhor desempenho com base no cálculo dessas métricas.

## Exemplos

A tabela abaixo apresenta exemplos com alguns dados de entrada e suas respectivas saídas esperadas. Certifique-se de testar seu programa com esses exemplos e com outros casos possíveis.

Entrada	Saída
3 50,10,5,85 20,5,8,67 30,12,4,88	Índice: 1 Acurácia: 0.9 Precisão: 0.83
4 70,15,8,78 60,20,10,80 45,5,3,92 80,7,15,98	Índice: 3 Acurácia: 0.94 Precisão: 0.9
2 100,0,0,50 80,10,2,98	Índice: 1 Acurácia: 1.0 Precisão: 1.0

### Atenção:

Se você não está familiarizado com a linguagem de programação, não se preocupe! Você pode usar uma das seguintes inteligências artificiais para te ajudar a entender o código:

- **ChatGPT:** <https://chat.openai.com/>
- **Copilot:** <https://copilot.microsoft.com/>
- **Gemini:** <https://gemini.google.com/>
- **Amazon Q** (Para Empresas): <https://aws.amazon.com/pt/q/>

Abaixo adicionamos algumas sugestões e uso e Prompts para te auxiliar na resolução:

Sugestões de Uso	Sugestões de Prompts
Explicação de Conceitos	Pode me explicar o que são estruturas de dados e dar exemplos?
Entendimento do Problema	Quais são as restrições ou requisitos específicos que devo considerar neste desafio?
Sugestões de Abordagem	Quais são as etapas principais que devo seguir para resolver este desafio?
Ajuda na Depuração	Estou recebendo um erro de sintaxe neste trecho de código. O que pode estar errado?
Revisão de Algoritmos	Você pode revisar meu algoritmo de ordenação e me dar feedback sobre sua eficiência?