



EXERCÍCIO 5

REGRESSÃO LOGÍSTICA

Objetivo

Neste exercício, será implementado o método de Regressão Logística: uma técnica tradicional de classificação que contribuiu para o crescimento da área. A teoria por trás deste método servirá como base matemática para métodos mais poderosos, como as redes neurais artificiais e as máquinas de vetores de suporte.

A Regressão Logística é um método de aprendizado supervisionado baseado em otimização, que tem como objetivo calcular a probabilidade de uma amostra pertencer a uma determinada classe, modelando o resultado através de uma função sigmoideal (logística).

Ao longo do exercício, diversos conceitos importantes para a área de Aprendizado de Máquina serão abordados, como função `sigmoid` e regularização.

Após concluir o exercício, é esperado que você consiga executar o método de Regressão Logística através da implementação de diversas etapas do algoritmo, desde a aplicação da função logística até a predição final, aplicando ou não regularização no cálculo do custo durante o aprendizado.

O exercício

Ao longo do exercício, você deverá completar 4 funções:

- `sigmoid`: responsável por modelar uma entrada numérica através da função sigmoid, i.e., a função logística.
- `funcaoCusto`: responsável por calcular o custo de usar um determinado valor de θ para o conjunto de dados do treinamento;
- `predicao`: responsável por prever a classe (0 ou 1) de cada uma das amostras pertencentes a um conjunto de teste;
- `funcaoCustoReg`: responsável por calcular o custo de usar um determinado valor de θ para o conjunto de dados do treinamento, utilizando um parâmetro λ de regularização;

Preencha o código apenas nos espaços delimitados por comentários, normalmente iniciados por um comentário “COMPLETE O CÓDIGO AQUI” e instruções para a implementação.

As implementações devem genéricas e funcionar para qualquer conjunto de dados. Na avaliação, as funções serão testadas em bases com quantidade de amostras e atributos diferentes das fornecidas com o exercício, tendo em comum apenas o nome da coluna que contém a classe das amostras. Não adicione comandos do tipo `print` ou `display` dentro das funções que serão completadas, apenas o código da função..

Aproveite para interagir com o *notebook*: teste valores diferentes para as amostras que serão previstas e procure entender como o método funciona.

Os casos de teste

Este exercício possui **5 casos de teste** que buscam avaliar cada uma das funções implementadas. A distribuição de tarefas avaliadas por cada caso é feita da seguinte forma:

- **Casos de teste 1:** testa a função `sigmoid`;
- **Caso de teste 2:** testa a função `funcaoCusto`;
- **Caso de teste 3:** testa a função `predicao`;
- **Caso de teste 4 e 5:** testa a função `funcaoCustoReg`.

Repare que a função `sigmoid` é utilizada em todas as demais funções que precisam ser implementadas. Portanto, um erro na implementação da função `sigmoid` propagará problemas para os casos de teste seguintes.

Detalhe sobre a implementação

Para calcular o custo e o gradiente nas funções `funcaoCusto` e `funcaoCustoReg`, deve ser utilizada multiplicação matricial, através de funções como `np.dot(a, b)`, `np.matmul(a, b)` e `a@b`. Utilizar `np.sum(a * b)` resultará em erro no *online judge* devido a imprecisão numérica nas casas decimais.