

# Trabalho 2

## Regressão Linear Regularizada e Polinomial

Universidade Federal do Ceará  
Aprendizado de Máquina  
Prof. Victor Farias



### Instruções

- A atividade permite o uso das bibliotecas numpy.
- Não pode usar scikit-learn.
- Os métodos devem ser implementados em 2 funções: uma para treino e outra para predição.

### Implementação

1. Implemente uma função de treino para seguinte método:
  - a. Regressão Linear multivariada regularizada – método analítico. Não esquecer de adicionar a coluna de 1s aqui e colocar o primeiro termo de  $\lambda I$  como 0.
2. Implemente a função para calcular predições
  - a. Função `predict_multivariate(X, w)` que recebe as amostras **X** para calcular a predição e o vetor de pesos **w**. Essa função retorna o vetor de predições que contém a predição para cada uma dessas amostras (na mesma ordem que as amostras foram dadas). Aqui você deve adicionar a coluna de 1s para casar com a quantidade de parâmetros do modelo.
3. Implemente as funções
  - a. `MSE(y_true, y_predict)`

- b. MAPE(y\_true, y\_predict)
- c. R2(y\_true, y\_predict)

## Experimento

1. Carregue o conjunto de dados **artificial1d.csv** [link](#).
2. Embaralhe o conjunto de dados (dica: np.random.shuffle)
3. Crie o vetor X com a variável de entrada e o y com a variável de saída
4. Crie as variáveis polinomiais  $X^2$ ,  $X^3$  e  $X^4$  e adicione à matriz X (dica: np.c\_[])
5. Separe o conjunto de treino e teste (70%/30%)
6. Treine o modelo de regressão linear regularizada
7. Reporte o MSE, MAPE e R2 para o conjunto de teste
8. Repita os passos 6 e 7 usando vários valores de  $\lambda$ .  $\lambda = \{0, 0.001, 0.01, 0.1, 1, 10, 100\}$