

## Atividade em Grupo 1

### 1. Especificação

Nesta atividade, cada grupo (máx. de 3 integrantes) deverá estender o código do Regressor Logístico com *regularização L2* visto em aula para um problema de classificação multi-classe. A validação do modelo será feita para a base de dados **Wine dataset** disponível no [Scikit-learn](https://scikit-learn.org/stable/datasets/real_world.html#wine-dataset).

O grupo deverá implementar as seguintes novas funcionalidades para o regressor logístico:

- Diferentes versões do gradiente descendente (otimizador):
  - Hiperparâmetro **optimizer** passado no construtor da classe:
  - Algoritmos:
    - Batch gradient descent (*optimizer="batch", default*)
    - Mini-batch gradient descent (*optimizer="mini"*)
    - Stochastic gradient descent (*optimizer="stochastic"*)
  - Incluir uma validação para este hiperparâmetro.
- Ao menos uma estratégia de classificação multi-classe.
  - Sugerido: One vs Rest

Seguem os critérios exigidos a serem avaliados. Cada critério tem um conjunto de pontos que servirão como um guia para seu desenvolvimento. Outros pontos não mencionados aqui também podem ser considerados.

- Desenvolvimento do modelo [5 pontos]
  - Códigos
  - Sugestão para debugging:
    - ◆ Crie um dataset 2D fake com 3 ou mais classes usando a função *make\_blobs* do sklearn;
    - ◆ Visualize as decisões boundaries aprendidas;
- Treinamento, Validação e Teste [4 pontos]
  - Dataset: **Wine dataset** disponível no [Scikit-learn](https://scikit-learn.org/stable/datasets/real_world.html#wine-dataset).
  - Limpeza e pré-processamento;
  - Fine-tuning;
  - Uso de Pipelines (opcional)

- Comparar ao menos 5 modelos diferentes. Sugestão:
  - ◆ Seu modelo com valores default para os hiperparâmetros usando:
    - Batch gradient descent
    - Mini-batch gradient descent
    - Stochastic gradiente descent
  - ◆ Seu modelo após otimização de hiperparâmetros via Fine-tuning;
  - ◆ *LogisticRegression* do sklearn (versão default e após fine-tuning);
  - ◆ *SGDClassifier* do sklearn (versão default e após fine-tuning);
  - ◆ PS: para uma comparação justa, garanta que todos os modelos estão usando exatamente os mesmos conjuntos de dados: treinamento, validação (k-fold) e teste;
- Relatório (Notebook): **[1 ponto]**
  - Organização do relatório;
  - Clareza na apresentação dos textos e códigos;
  - Qualidade do código;

## 2. Entregáveis

Cada grupo deverá preparar um **único jupyter notebook** com os códigos feitos para a resolução dos problemas, bem como comentários e discussões sobre os mesmos.

## 3. Submissão (prazo: 12/05/22)

- A submissão da atividade será feita em tarefa específica no Moodle da disciplina.
- O grupo enviar um jupyter notebook (.ipynb) ou o link do repositório online com o código (ex., Google Colab, GitHub, Kaggle).
  - No caso dos links para repositórios ou plataformas online, serão considerados apenas aqueles com atualização até o prazo de entrega desta atividade.
- Apenas **um membro do grupo** deverá submeter a atividade.