Questão 1: Regressão Linear

Implemente um modelo de regressão linear. Para isso, utilize um conjunto de dados sintético gerado com a equação

y = 3x + 5 + epsilon

onde x segue distribuição uniforme entre -10 e 10 e epsilon é um ruído gaussiano com média zero e desvio padrao de 2.

Faça os seguintes passos:

* Gere um conjunto de dados com pelo menos 100 pontos.
* Divida os dados em treino (80\%) e teste (20\%).
* Implemente modelos de regressao linear empregando:
  + a solução de mínimos quadrados (pseudo-inversa);
  + uma rede neural com uma camada treinada via gradiente descendente utilizando MSELoss (Erro Quadratico Medio) e otimizador SGD.
* Apresente as soluções para cada um dos métodos acima.
* Avalie o desempenho dos modelos e visualize os resultados.

Questao 2:

Implemente um modelo de regressao logistica para resolver um problema de classificacao binaria utilizando um conjunto de dados sintetico.

Passos:

* Utilize a funcao make\_classification da biblioteca Scikit-Learn para gerar um conjunto de dados com 500 amostras, 2 variaveis preditoras e 2 classes.
* Divida os dados em treino (70\%) e teste (30\%).
* Implemente um modelo de regressao logistica.
* Treine o modelo utilizando gradiente descendente (versão não-estocástica) (conforme visto em sala).
* Avalie a acuracia no conjunto de teste e visualize a fronteira de decisao do classificador.

Questao 3

Implemente uma rede neural do tipo MLP para a tarefa de classificacao binaria. Nesta questao, voce deve usar um conjunto de validacao para selecionar o numero adequado de neuronios na camada oculta.

Passos:

* Utilize a funcao make\_moons da biblioteca Scikit-Learn para gerar um conjunto de dados com 500 amostras.
* Divida os dados em treino (70\%), validacao (15\%) e teste (15\%).
* Implemente uma MLP com:
  + Uma camada oculta com n neuronios e ativacao ReLU.
  + Uma camada de saida com 1 neuronio e ativacao sigmoid.
  + Utilize a funcao de perda BCELoss e o otimizador Adam ou SGD.
* Treine modelos com diferentes numeros de neuronios na camada oculta (exemplo: 5, 10, 20, 50).
* Plote a evolução da função custo (loss) ao longo do treinamento (épocas).
* Escolha o melhor numero de neuronios com base na menor perda no conjunto de validacao.
* Avalie o modelo escolhido no conjunto de teste e visualize a fronteira de decisao.

Questao 4:

Agora, implemente uma rede neural para a classificacao de imagens do conjunto MNIST.

Faça os seguintes passos:

* Carregue o conjunto de dados MNIST utilizando Torchvision ou Keras.
* Normalize as imagens e divida em treino (80\%) e teste (20\%).
* Implemente uma rede neural MLP com:
  + Uma camada oculta de 128 neuronios e ativacao ReLU.
  + Uma camada oculta de 64 neuronios e ativacao ReLU.
  + Uma camada de saida com 10 neuronios e ativacao softmax.
* Utilize a funcao de perda CrossEntropyLoss e o otimizador Adam our SGD.
* Treine a rede por 10 epocas e avalie a acuracia no conjunto de teste.
* Exiba algumas previsoes feitas pelo modelo, mostrando imagens e suas respectivas classes previstas.

**Sobre a entrega**

Um relatório deve ser enviado até a data especificada no início deste documento via google classroom. Preferencialmente, o relatório deve consistir de um único python notebook com as respostas, gráficos, comentários e códigos.

Cada relatório pode ser desenvolvido por até duas pessoas.

Para que a nota seja contabilizada, a equipe deve agendar a apresentação do trabalho com professor. Os membros da equipe devem ser capazes de responder perguntas relacionadas à detalhes de implementação e conceitos básicos dos conteúdos.

As datas disponíveis para apresentação são:

* dia 17.02 (horário: 08-10h)
* dia 19.02 (horário: 08-10h)
* dia 21.02 (horário: 10-12h)