

Treinamento CIS - 3º Período (Redes Neurais)

1. Conteúdos do Período

- a. Introdução a Redes Neurais:
 - i. Perceptron com 1 ou mais camadas;
 - ii. Backpropagation;
 - iii. Operações vetoriais;
 - iv. Gradiente descendente;
 - v. Ótimo local e global;
 - vi. Métricas de avaliação;
 - vii. Função de custo;
 - viii. Overfitting e underfitting;
 - ix. Função de ativação;
 - x. Learning rate.
- b. Implementação de Regressão Linear com perceptron;
- c. Problemas lineares e não lineares;
- d. Implementação usando TensorFlow.

2. Conteúdo de apoio

- a. [Playlist Neural Networks 3blue1brown](#) - Playlist que explica o que são Redes Neurais, Gradiente descendente e Backpropagation.
- b. [MIT Introduction to Deep Learning](#) - Aula do MIT sobre fundamentos do Deep Learning.
- c. [Neural Networks and Deep Learning](#) - Aulas do curso 1 da especialização em Deep Learning do deeplearning.ai.
- d. [Improving Deep Neural Networks: Hyperparameter Tuning, Regularization and Optimization](#) - Aulas do curso 2 da especialização em Deep Learning do deeplearning.ai.
- e. [DeepLearning Book](#) - Livro em português sobre Deep Learning. Para o quarto período, recomenda-se os capítulos 1 ao 21.
- f. [Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow](#) - Livro completo: Para o terceiro período recomenda-se o capítulo 10.

3. Tarefas

- a. Base de Dados do Período - [Credit Card Fraud Detection](#)
- b. Atividade obrigatória:
 - i. Classificação binária para prever fraudes nas transações com cartões de crédito usando um Perceptron com uma camada oculta feito somente com numpy:
 1. Separar a label das features e o dataset em subsets de treinamento e teste;
 2. Inicialização randômica dos pesos;
 3. Definir a função de ativação e calcular sua derivada;
 4. Treinar o modelo testando diferentes valores de épocas e learning rate, identificando quando acontece Overfitting e Underfitting.



5. Fazer as previsões nos dados de teste e avaliar o modelo.
- c. Atividades sugeridas:
 - i. Testar o efeito da Regularização ou outros métodos de otimização;
 - ii. Testar o efeito de mais uma camada oculta de neurônios;
 - iii. Comparar com a implementação usando TensorFlow;
 - iv. Aplicar as técnicas em outro conjunto de dados como o MNIST, que pode ser importado diretamente para o notebook através do Keras.
 - d. No meio do período, haverá uma reunião com o monitor para consolidação das informações;
 - e. A entrega é individual e deverá ser colocada no seu GitHub pessoal