

### 1ª Lista de Exercícios - Classificação

Inspirado no comportamento do cérebro humano, Redes Neurais são sistemas com capacidade de armazenar conhecimento e aprender através da apresentação iterativa dos dados. As aplicações de Redes Neurais são diversas, como classificação, regressão, previsão de séries temporais e agrupamento.

Nesta lista de exercícios, serão discutidos todos os passos para o problema de classificação. São disponibilizadas as seguintes bases de dados:

- Breast Cancer
- Diabetes
- German Credit
- Heart Disease
- Occupancy Detection
- Thyroid
- Titanic
- Bank Marketing
- MNIST
- Wine Quality
- Ionosphere
- Credit Analysis

Por mais que as bases de dados sejam diferentes, com atributos, particularidades e natureza do problema distintos, este trabalho será dividido em quatro etapas que são comuns a qualquer problema de classificação. Cada etapa terá uma série de perguntas que servirão como um guia para o relatório da Prática I. Em cada pergunta, **fundamente a sua escolha e faça análises que sejam pertinentes**, com base no que foi aprendido em sala de aula. Evite inserir apenas o resultado da simulação, pois a análise é a parte mais relevante para avaliar o relatório.

Além deste roteiro de atividades, também é disponibilizado um arquivo em Python (class\_pratica1.ipynb) e a base de dados escolhido previamente. O script possui alguns códigos já escritos para auxiliar o andamento desta prática, além de campos definidos para gerar os resultados das perguntas. Por conta disso, além do relatório escrito, seguindo as perguntas deste roteiro, envie também o arquivo pré-executado do script em Jupyter Notebook.

## Roteiro de atividades

### 1. Compreensão do problema e análise de variáveis.

- 1) Observe a base de dados do problema. Existem variáveis que podem ser eliminadas do dataset? Justifique.
- 2) Implemente técnicas de visualização de dados e seleção de variáveis para extrair características importantes sobre a base de dados. Explique a motivação destas técnicas e o que é possível inferir dos resultados obtidos.

### 2. Treinamento do modelo de Rede Neural

- 1) Com as configurações do modelo MLP previamente definidas no script, faça o treinamento da Rede Neural **sem** normalizar os atributos numéricos. Comente o resultado obtido, baseado nas métricas de avaliação disponíveis (acurácia, precision, recall, F1-Score, Matriz de confusão, etc.)
- 2) Agora normalize os dados de entrada e treine novamente o modelo MLP. Avalie os resultados obtidos e comente o efeito da normalização no treinamento da Rede Neural.

### 3. Mudança de configurações do modelo

- 1) Insira o conjunto de validação para o treinamento do modelo. Avalie o resultado obtido.
- 2) Modifique o tempo de treinamento (épocas) da Rede Neural. Escolha valores distintos (e.g. 1 e 1000 épocas) e avalie os resultados.
- 3) Modifique a taxa de aprendizado da Rede Neural. Escolha valores distintos (e.g. 0,001 e 0,1) e avalie os resultados.

- 4) Modifique a quantidade de neurônios na camada escondida da Rede Neural. Escolha valores distintos (e.g. 2 e 70 neurônios) e avalie os resultados.

## 4. Teste livre

- 1) Faça novos testes para avaliar o desempenho da Rede Neural no problema designado usando a técnica K-Fold (com  $K = 10$ ). Comente o resultado obtido.

Dica: não é necessário utilizar a ferramenta KFold do scikit-learn para o desenvolvimento deste item. Uma alternativa simples é fazer um for loop, selecionando os índices em bloco.

- 2) Faça análises e novas implementações que você julgue importante para o seu trabalho. Não esqueça de explicar a motivação da análise realizada.