Universidade Federal do Ceará - Campus Crateús
Professor: Renan Gomes Vieira
Disciplina: Ciência dos Dados

T3 (peso 6)

As notas da disciplina de Ciência dos dados 2025.2 serão atribuídas a partir de trabalhos desenvolvidos em grupo. Este terceiro e último trabalho tem como objetivo desenvolver as habilidades de **Aprendizado de Máquina** e **Processamento de Linguagem Natural**. Sigam as seguintes instruções abaixo:

- A entrega deve ser feita pelo mesmo grupo estabelecido nos trabalho 1 e 2.
- O trabalho tem dois principais objetivos:
  1. Aplicar na prática os fluxos de treinamento e avaliação para problemas de aprendizado de máquina (Regressão, Classificação e Clusterização).
  2. Aplicar na prática o fluxo de processamento de linguagem natural para problemas de classificação de texto.
- Na parte da **Aprendizado de máquina** (Peso 4), é esperado:
  - Avaliação de diferentes modelos de aprendizado de máquina para um problema de *classificação* em dados tabulares, aplicando o processo adequado de processamento dos dados, avaliação de modelos e comparação de métricas (Matriz de confusão, Acuracia, F1-score, Recall, precision). É necessário comparar pelo menos 5 modelos diferentes, comparando diferentes valores de hiperparâmetros em cada modelo. Usem validação cruzada. Ao final da comparação, indique qual modelo mais promissor e quais os hiperparâmetros encontrados. Atente-se para problemas de desbalanceamento dos dados e dados faltantes. Trate adequadamente quando necessário.
  - Avaliação de diferentes modelos de aprendizado de máquina para um problema de *regressão* **OU** de *clusterização*, aplicando o processo adequado de processamento dos dados, avaliação dos modelos e comparação de métricas (R^2 e RMSE para regressão; Silhouette/Davies–Bouldin index para clusterização). Em caso de regressão, use validação cruzada. Em casso de clusterização, descreva bem o processo de selecionar o número de grupos, caso o algoritmo necessite. É necessário comparar pelo menos 3 modelos diferentes, comparando diferentes valores de hiperparâmetros. Ao final da comparação, indique qual modelo mais promissor e quais os hiperparâmetros encontrados. Atente-se para problemas dados faltantes. Trate adequadamente quando necessário.
  - **BONUS**: ganhe pontos extras ao trazer explicabilidade dos modelos no problema de classificação e regressão. Ganhe pontos extras ao visualizar os dados em 2d em problemas de clusterização, utilizando técnicas de redução de dimensionalidade.
- Na parte de **Processamento de linguagem natural** (Peso 2) é esperado:

  - Aplicação de técnicas de processamento de linguagem natural para problemas de classificação de texto. A ideia é fazer algo muito similar a parte de aprendizado de máquina com problemas de classificação, mas agora utilizando dados textuais. Compare o desempenho dos modelos usando TF-IFF e Bag of Words como veteores para representar os textos. Sugiro explorar diferentes valores para os hiperparâmetros de cada técnica (conforme a API do sklearn) a seguir:
    * Bag of Words: vect__ngram_range: define se você usa só unigramas ou inclui bigramas/trigramas (captura mais contexto); vect__min_df: remove termos muito raros (aparecem em poucos documentos), reduz ruído e dimensionalidade; vect__max_df: remove termos muito comuns (aparecem em muitos documentos), ajudando a tirar "stopwords implícitas".
    * TF-IDF: tfidf__ngram_range: igual ao BoW, controla o nível de contexto (uni/bi/tri-gramas); tfidf__min_df: mesma ideia do BoW: corta termos raros pra reduzir ruído; tfidf__sublinear_tf: usa 1 + log(tf) em vez de tf puro; diminui o peso de repetições muito altas no mesmo documento.
    * É incentivado a exploração de outros hiperparâmetros além desses.
  - **BONUS**: ganhe pontos extras ao trazer visualizações usando os dados textuais, como por exemplo nuvens de palavras.

- Será necessário apresentar pelo menos um Notebook para cada entrega (Aprendizado de máquina e NLP), sendo eles claros e lineares com conclusões e comentários marcadas ao longo do arquivo.
- Será necessário também entregar um breve relatório de três páginas no máximo (não é necessário capa, basta cabeçalho simples e texto corrido), com uma breve contextualização dos dados escolhidos, os principais aprendizados e desafios durante o desenvolvimento do trabalho.
- Cada grupo deve apresentar os artefatos do trabalho em uma apresentação de até 20 minutos. A apresentação é livre (você pode fazer slides ou apresentar diretamente do código e/ou notebooks) mas é necessário mostrar os notebooks e códigos utilizados no desenvolvimento do trabalho. Ela deve ser dividida em dois momentos: primeiro o trabalho de aprendizado de máquina e depois de processamento natural de texto. Cada parte será avaliada de forma separada.
- **Tudo deve estar disponível no Github do projeto**.
- Cada parte do trabalho será avaliada seguindo os seguintes critérios
  1. Organização e reprodutibilidade (10 %): estrutura do projeto, código limpo, seeds/splits claros, instruções de execução.
  2. Relatório (15 %): contextualização do problema, descrição dos dados, justificativas curtas, resultados bem resumidos (tabelas/gráficos) e conclusões/limitações.
  3. Apresentação (15 %): clareza na fala/slides/notebooks, capacidade de explicar decisões, tempo bem usado, respostas a perguntas.
  4. Corretude técnica (30 %): pipelines corretos, pré-processamento adequado, métricas apropriadas por tarefa, splits/validação bem feitos, interpretação coerente.

5. Detalhamento e exploração (30 %): variedade de modelos (baseline + alternativas), GridSearch/ajuste bem definido, comparação justa, análises extras (erros, resíduos, importância de features, visualizações/2D na clusterização, etc.).

- **Outras fontes de dados**: Caso você consiga utilizar algum dos dados dos trabalhos anteriores, ótimo. Caso não, fiz outra curadoria com mais locais para encontrar dados para essa atividade.
  - **UCI Machine Learning Repository (geral; muitos tabulares p/ classificação e também úteis p/ clusterização):**
    https://archive.ics.uci.edu/datasets
  - **OpenML (geral; bom p/ baixar datasets padronizados e fazer benchmarking):**
    https://www.openml.org/search?type=data
  - **Kaggle Datasets (geral; muita variedade e casos do mundo real):**
    https://www.kaggle.com/datasets
  - **Google Dataset Search (buscador de datasets na web; bom p/ achar coisa bem específica):**
    https://datasetsearch.research.google.com/
  - **Hugging Face Datasets (NLP forte; muitos benchmarks prontos):**
    https://huggingface.co/datasets
    https://huggingface.co/docs/datasets/en/index
  - **TensorFlow Datasets (TFDS) (coleção pronta p/ texto/visão; útil p/ classificação e também p/ testes de clusterização):**
    https://www.tensorflow.org/datasets
    https://www.tensorflow.org/datasets/catalog/overview
  - **Registry of Open Data on AWS (datasets grandes e públicos, multidomínio):**
    https://registry.opendata.aws/
  - **fast.ai datasets (coleção bem prática; tem NLP/classificação bem conhecida):**
    https://course20.fast.ai/datasets.html
  - **fast.ai NLP no Registry (IMDb, AG-News, Yelp, DBpedia, etc.):**
    https://registry.opendata.aws/fast-ai-nlp/