Requisitos para o Desenvolvimento do MVP

Contexto:

1. Escolha bases de dados que não tenham sido utilizadas em aula. Sugere-se usar bases de dados disponibilizada em algum dos repositórios a seguir:

- UCI Machine Learning Repository: https://archive.ics.uci.edu/ml/datasets.php
- Kaggle: https://www.kaggle.com/datasets
- Google Datasets: https://datasetsearch.research.google.com/
- Hugging Face: https://huggingface.co/datasets

Aproveite os filtros que oferecem os repositórios para encontrar com mais facilidade os datasets da sua preferência. Caso você prefira usar um dataset que reflita um problema real da sua empresa (cuidado apenas com a confidencialidade dos dados), será muito bem-vindo.

2. Você deverá escolher e resolver um problema de aprendizado de máquina (**supervisionado ou não supervisionado**) em um dos tipos de problema: classificação, regressão, clusterização ou previsão de séries temporais (forecasting). O problema pode estar em diferentes áreas de aplicação, como visão computacional, processamento de linguagem natural (NLP), ou outros domínios (dados tabulares, sensores, dados temporais, etc.). A solução pode empregar modelos clássicos de aprendizado de máquina e/ou aprendizado profundo (Deep Learning), treinados do zero ou via fine-tuning de modelos pré-treinados.

O MVP deve contemplar:

- Carga e preparação dos dados (quando pertinente; o MVP da sprint de Análise de Dados & Boas Práticas pode servir de base).
- Divisão dos dados adequada ao problema (treino, validação, teste; validação cruzada; sempre evitando vazamento de dados)
- Tratamento de dados: limpeza, transformação, seleção/extração de atributos, engenharia de atributos, etc.
- Modelagem: treinar e comparar abordagens e modelos diferentes; quando possível, usar pipelines reproduzíveis.
- Otimização de hiperparâmetros: explorar ajustes relevantes para cada modelo.
- Avaliação: utilizar métricas adequadas ao tipo de problema, comparar modelos e discutir limitações e melhorias.

- Boas práticas: estabelecer baseline, fixar seeds para reprodutibilidade, relatar recursos computacionais usados e tempo de treino, e documentar as decisões de projeto.

O *notebook* pode seguir a estrutura proposta neste template, mas isso não é mandatório. A melhor organização do seu MVP dependerá, em grande parte, da forma como você deseja abordar o problema e comunicar seus insights. Caso opte por usar o template, lembre-se de fazer uma cópia para o seu Drive antes de editar, já que este link leva a uma versão apenas para leitura.

3. Produza um notebook no Google Colab, considerando o item 1 e o item 2, com as características a seguir:

a. O notebook servirá como relatório, descrevendo textualmente (utilizando as células de texto) o contexto do problema e as operações com os dados (veja o checklist sugerido abaixo)
b. Utilize a linguagem Python e bibliotecas que considere apropriadas para abordar o problema.
c. Crie o notebook seguindo as boas práticas de codificação vistas na disciplina *Programação Orientada a Objetos*.

Observações:

- Os datasets podem ser selecionados de acordo com a sua escolha, desde que não sejam os datasets vistos nas disciplinas da sprint. O dataset deve ser carregado através de uma URL dentro do próprio notebook, de forma a permitir a execução direta do código sem necessidade de realizar qualquer tipo de configuração ou ajuste. Veja um exemplo de importação através de URL no link https://colab.research.google.com/github/dipucriodigital/ciencia-de-dados-e-analytics/blob/main/analise-exploratoria-pre-processamento-de-dados/AEPD_importando_datasets.ipynb (forma 1 do notebook).
- Durante a sua análise dos resultados dos modelos, no momento de utilizar gráficos, podem ser utilizadas as bibliotecas Python vistas nas disciplinas, ou outras à sua escolha. Se forem utilizadas para a construção dos gráficos outras ferramentas que não as bibliotecas Python, você deverá adicionar no notebook uma figura com cada gráfico produzido.
- Apesar de recomendarmos a utilização do Colab, você pode construir o código na IDE que desejar. Porém, a entrega final do MVP deverá obrigatoriamente ser um notebook público no Colab.

Requisitos e composição da nota:

- ***(1,0 pt)*** **Execução sem erros:** o notebook deve poder ser executado pelo professor do início ao fim sem erros.

- ***(2,0 pts) Documentação consistente:*** *utilize blocos de texto que expliquem textualmente cada etapa e cada decisão do seu código, contando uma história completa e compreensível, do início ao fim.*
- ***(1,0 pt)* Código limpo:** seu código deve estar legível e organizado. Devem ser utilizadas as boas práticas de codificação em Python (sugestão: consulte a disciplina *Programação Orientada a Objetos*), mas não é necessário que você crie classes no seu código.
- ***(2,0 pts)* Análise de resultados do modelo:** você deverá escrever um bloco de texto resumindo os principais achados, analisando os resultados e levantando eventuais pontos de atenção. Sugerimos também incluir um bloco de texto de conclusão do problema como um todo, resumindo os principais pontos e fazendo um fechamento.
- ***(2,0 pt)* Checklist:** você deverá responder às perguntas (aplicáveis ao seu dataset) do checklist fornecido, utilizando-o como guia para o desenvolvimento do trabalho.
- ***(2,0 pts)*** Capricho e qualidade do trabalho como um todo.

Checklist sugerido:

Definição do Problema

Objetivo: entender e descrever claramente o problema que está sendo resolvido.

- Qual é a descrição do problema?
- Você tem premissas ou hipóteses sobre o problema? Quais?
- Que restrições ou condições foram impostas para selecionar os dados?
- Descreva o seu dataset (atributos, imagens, anotações, etc).

Preparação de Dados

Objetivo: realizar operações de preparação dos dados.

- Separe o dataset entre treino e teste (e validação, se aplicável).
- Faz sentido utilizar um método de validação cruzada? Justifique se não utilizar.
- Verifique quais operações de transformação de dados (como normalização e padronização, transformação de imagens em tensores) são mais apropriadas para o seu problema e salve visões diferentes do seu dataset para posterior avaliação dos modelos.
- Refine a quantidade de atributos disponíveis, realizando o processo de *feature selection* de forma adequada.

Modelagem e treinamento:

Objetivo: construir modelos para resolver o problema em questão.

- Selecione os algoritmos mais indicados para o problema e dataset escolhidos, justificando as suas escolhas.
- Há algum ajuste inicial para os hiperparâmetros?

- O modelo foi devidamente treinado? Foi observado problema de *underfitting*?
- É possível otimizar os hiperparâmetros de algum dos modelos? Se sim, faça-o, justificando todas as escolhas.
- Há algum método avançado ou mais complexo que possa ser avaliado?
- Posso criar um comitê de modelos diferentes para o problema (*ensembles*)?

Avaliação de Resultados:

Objetivo: analisar o desempenho dos modelos gerados em dados não vistos (com a base de teste)

- Selecione as métricas de avaliação condizentes com o problema, justificando.
- Treine o modelo escolhido com toda a base de treino, e teste-o com a base de teste.
- Os resultados fazem sentido?
- Foi observado algum problema de *overfitting*?
- Compare os resultados de diferentes modelos.
- Descreva a melhor solução encontrada, justificando.

Sobre a entrega:

Você deverá disponibilizar UM ÚNICO notebook com o código em Python e eventuais arquivos com os datasets necessários para a execução do seu código em um repositório público do GitHub. A utilização do dataset dentro do notebook deve ser feita através da URL do seu repositório do GitHub (veja o exemplo abaixo). O caminho do notebook deve ser informado na tarefa de entrega do MVP, no ambiente do curso. Garanta que os links informados funcionem quando fizer a entrega do seu MVP

Se tiver dúvidas sobre como criar um repositório público no GitHub, consulte: https://docs.github.com/pt/repositories/creating-and-managing-repositories/creating-a-new-repository

Exemplo de como ler o dataset a partir do seu repositório do GitHub: colocar o arquivo em um repositório seu do github e referenciá-lo na URL com a sua versão raw, por exemplo: https://raw.githubusercontent.com/tatianaesc/datascience/main/diabetes.csv