Você é um assistente experiente em **Engenharia de Dados**, **Machine Learning**, **Python**, **Google Colab**, e **Git/GitHub**. O objetivo é construir um **projeto de machine learning completo** com foco em **contratos** e **variáveis do CVCRM**, estruturado para ser versionado no GitHub e executado no Google Colab. Este projeto deve incluir:

**1. Contextualização**

O projeto foca em **contratos do CVCRM**, com uso extensivo de variáveis, como as que discutimos. Queremos explorar modelos que sejam capazes de prever categorias de contratos ou identificar padrões nas variáveis fornecidas. A solução precisa ser **modular**, **reutilizável** e **colaborativa**, garantindo que as etapas do pipeline sejam claras e bem documentadas.

Inclua insights aprendidos:

* Como estruturar um pipeline que receba variáveis em JSON.
* Como usar **modelos de machine learning** com validação e predição.
* Detalhes sobre como tratar erros comuns, como entradas inválidas ou inconsistentes.

**2. Estrutura de Pastas e Arquivos**

Crie uma estrutura que suporte organização e escalabilidade:

plaintext

Copy code

project/

├── data/ # Dados do projeto

│ ├── raw/ # Dados brutos

│ ├── processed/ # Dados pré-processados

├── notebooks/ # Notebooks para análise e experimentação

├── model\_training/ # Scripts para treinar os modelos

├── model\_validation/ # Validação e métricas do modelo

├── model\_prediction/ # Scripts para inferência

├── utils/ # Funções auxiliares

├── tests/ # Testes unitários

├── requirements.txt # Dependências do projeto

├── README.md # Documentação do projeto

├── .gitignore # Arquivos a serem ignorados pelo Git

├── .env # Configurações sensíveis

└── main.py # Script principal do projeto

Inclua:

* Código para tratamento de variáveis no formato JSON.
* Scripts prontos para executar no Google Colab ou localmente.

**3. Configuração do Repositório no GitHub**

Passos para configurar o Git/GitHub:

1. **Criação do Repositório**
   * Escolha um nome descritivo, como cvcrm-ml-project.
   * Adicione um README.md inicial.
2. **Inicializar localmente**:

bash

Copy code

git init

git remote add origin <URL\_DO\_REPOSITORIO>

git pull origin main

1. **Configurar para o Google Colab**:
   * Clone no Colab:

bash

Copy code

!git clone <URL\_DO\_REPOSITORIO>

* + Use o repositório no Colab para desenvolvimento colaborativo.

**4. Pipeline Completo de Machine Learning**

Crie um pipeline com as seguintes etapas:

**1. Coleta de Dados**

* Leia dados de variáveis de contratos em JSON.
* Exemplo de entrada:

json

Copy code

{

"variavel": "valor1",

"outra\_variavel": "valor2"

}

**2. Pré-Processamento**

* Código para conversão e validação das entradas:

python

Copy code

import pandas as pd

import numpy as np

def preprocess\_data(json\_data):

try:

df = pd.DataFrame([json\_data])

df = df.astype(float)

return df

except Exception as e:

print(f"Erro no pré-processamento: {e}")

**3. Treinamento de Modelos**

* Treine modelos, como RandomForestClassifier ou LogisticRegression.
* Exemplo:

python

Copy code

from sklearn.ensemble import RandomForestClassifier

model = RandomForestClassifier()

model.fit(X\_train, y\_train)

**4. Validação**

* Métricas para avaliar o desempenho (e.g., AUC, Accuracy).

**5. Predição**

* Receba entradas via JSON e gere predições.

**5. Execução no Google Colab**

* Monte o Drive para acesso aos dados:

python

Copy code

from google.colab import drive

drive.mount('/content/drive')

* Clone o repositório:

bash

Copy code

!git clone <URL\_DO\_REPOSITORIO>

* Instale as dependências:

bash

Copy code

!pip install -r requirements.txt

**6. Documentação**

Crie um README.md que inclua:

* **Descrição do projeto.**
* **Exemplo de uso**:
  + Comando cURL para teste:

bash

Copy code

curl -X POST http://127.0.0.1:5000/predict -H "Content-Type: application/json" -d "{\"variavel\": \"1.23\", \"outra\_variavel\": \"4.56\"}"

**7. Boas Práticas**

* Use mensagens de commit claras:

bash

Copy code

git commit -m "Adiciona pipeline de treinamento"

* Armazene credenciais em .env:

plaintext

Copy code

API\_KEY=chave\_secreta

**Objetivo Final**

Ter um projeto **completo**, **versionado no GitHub**, e **executável no Google Colab**, com pipeline funcional para tratar e processar variáveis de contratos do CVCRM, treinar modelos e gerar predições. O projeto deve ser colaborativo, modular e escalável.