PETROBRAS

Explicação do código da Análise CSFR

INTERNA

Explicação do Código de Análise CSFR

Realizado por:
David de Assis Vieira
Cientista de dados
Morning Star Consulting

Explicação do Código de Análise CSFR

Bloco 1: Importação de Bibliotecas e Carregamento dos Dados

O código começa instalando os pacotes necessários como pandas, scikit-learn, nltk e openpyxl. Estas são ferramentas que permitem manipular dados, aplicar Machine Learning e trabalhar com arquivos Excel.

Em seguida, importa as bibliotecas específicas que serão utilizadas. O pandas (pd) é usado para manipulação de dados tabulares, NumPy (np) para operações numéricas, e várias funções do scikit-learn para os modelos de Machine Learning.

Um sistema de logging é configurado para registrar informações sobre a execução do código, facilitando a identificação de problemas.

O código baixa uma lista de “stopwords” (palavras comuns como “e”, “o”, “a" que geralmente são ignoradas em análises de texto) e verifica a existência do arquivo Excel antes de tentar carregá-lo.

Por fim, carrega o arquivo Excel em uma estrutura de dados chamada DataFrame, que funciona como uma tabela manipulável.

Bloco 2: Análise Exploratória e Pré-processamento

Nesta etapa, o código analisa os dados e prepara-os para o modelo de Machine Learning:

- Verifica quantos valores nulos existem em cada coluna
- Define quais colunas serão usadas como entradas (features) e quais serão previstas (targets)
- Verifica se todas as colunas necessárias existem no arquivo
- Preenche valores vazios nas colunas de entrada
- Transforma o texto da coluna "Descrição Linha de Serviço" em valores numéricos usando TF-IDF
- Converte colunas categóricas como "Classificação", "Fornecedor" e "Moeda" em formato numérico
- Separa as linhas que já têm valores de destino (para treino) das que precisam ser preenchidas.

Bloco 3: Treinamento dos Modelos de Machine Learning

O código treina dois modelos diferentes:

1. **Um classificador para os índices**: Usa RandomForestClassifier para prever quais índices (como IGP, INS, etc.) devem ser atribuídos a cada linha.
2. **Um regressor para os pesos**: Usa RandomForestRegressor para prever os valores numéricos dos pesos associados a cada índice.

Antes de treinar, verifica se há dados suficientes e usa validação cruzada para avaliar a qualidade dos modelos.

Bloco 4: Previsões e Garantia de Consistência

Para as linhas com valores faltantes, o código:

- Usa os modelos treinados para prever índices e pesos
- Garante a consistência entre índices e pesos:
    - Se um peso tem valor positivo, mas não tem índice, atribui"IGP" como índice padrão
    - Se um índice está vazio, zera o peso correspondente
- Normaliza os pesos quando a soma excede 1.0

Bloco 5: Combinação de Previsões com Dados Originais

O código mantém os valores originais onde existem e somente preenche os valores faltantes com as previsões:

- Verifica cada linha para identificar quais campos precisam ser preenchidos
- Aplica previsões somente para pares válidos de índice e peso
- Registra quais células foram preenchidas pelo modelo para destacá-las depois.

Bloco 6: Exportação para Excel com Destaque Visual

Por fim, o código cria uma planilha Excel:

- Define cores diferentes para cada grupo de índice, peso e fator
- Escreve todos os dados na planilha
- Destaca com cores as células preenchidas pelo modelo
- Salva o arquivo com as previsões destacadas

Resumo do Funcionamento

1. **Preparação**: Carrega e organiza os dados
2. **Aprendizado**: Treina modelos para reconhecer padrões nos dados existentes
3. **Previsão**: Usa esses padrões para preencher dados faltantes
4. **Consistência**: Garante que os dados previstos sigam regras lógicas
5. **Visualização**: Cria uma planilha com os novos valores destacados em cores

Os modelos Machine Learning analisam características como descrição do serviço, classificação, fornecedor e moeda para determinar quais índices e pesos são mais apropriados para cada linha de serviço, baseando-se nos padrões encontrados nos dados existentes.