

Apresentação: Análise de Dados e Resultados do Código

Introdução

Este projeto foi desenvolvido para analisar um conjunto de dados sobre ingressantes e formandos, com foco na comparação entre a quantidade de formandos do sexo masculino e feminino. O código implementa duas abordagens principais: **Árvore de Decisão** e **Apriori**.

Estrutura do Código

1. Importação de Bibliotecas

O código começa com a importação de bibliotecas essenciais para análise de dados e aprendizado de máquina:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from mlxtend.frequent_patterns import apriori, association_rules
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.decomposition import PCA
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix
```

2. Funções de Processamento e Análise

a. Função `carregar_dados`

Esta função carrega dados de um arquivo Excel ou CSV e os retorna como um DataFrame do Pandas.

b. Função `verificar_e_transformar_dados`

Transforma os dados em uma matriz de cestas para a análise com o algoritmo Apriori. Essa função também exibe um exemplo dos dados.

c. Função `aplicar_apriori`

Aplica o algoritmo Apriori para identificar conjuntos frequentes de itens em uma matriz de cestas.

d. Função `gerar_regras`

Gera regras de associação a partir dos conjuntos frequentes encontrados, usando métricas como confiança e lift.

e. Função `gerar_grafico_comparacaosexo`

Cria um gráfico de barras comparando a quantidade de formandos por sexo.

f. Funções para a Árvore de Decisão

- ⑩ `explorar_dados`: Exibe um resumo estatístico dos dados e um mapa de calor indicando valores ausentes.
- ⑩ `limpar_dados`: Remove duplicatas e preenche valores ausentes com a mediana das colunas numéricas.
- ⑩ `transformar_dados`: Normaliza as variáveis numéricas e transforma variáveis categóricas com codificação de rótulos.
- ⑩ `reduzir_dimensao`: Aplica PCA para reduzir a dimensionalidade dos dados.
- ⑩ `treinar_modelo`: Treina um modelo de classificação RandomForest e exibe um relatório de classificação e uma matriz de confusão.

3. Função main

A função principal orquestra a execução do código, solicitando ao usuário a escolha entre a análise com a árvore de decisão ou o algoritmo Apriori.

Resultados

Análise dos Dados

Os resultados da análise mostraram que **o número de formandos do sexo masculino é maior do que o número de formandos do sexo feminino**. Esse resultado foi visualizado através de gráficos e análises estatísticas detalhadas.

Gráficos Gerados

- ⑩ **Gráfico de Comparação de Formandos por Sexo**: Mostra claramente a diferença entre a quantidade de formandos masculinos e femininos.
- ⑩ **Matriz de Confusão**: Apresenta a acurácia do modelo de árvore de decisão e destaca a distribuição de previsões entre os diferentes sexos.
- ⑩ **Gráfico de Distribuição de Previsões por Sexo**: Uma visualização adicional que detalha a distribuição das previsões feitas pelo modelo.

Conclusão

O código desenvolvido conseguiu extrair informações valiosas sobre a diferença de formação entre os sexos. A análise revelou uma diferença significativa, onde os formandos do sexo masculino são mais numerosos que os do sexo feminino. Esses insights podem ajudar na tomada de decisões acadêmicas e políticas relacionadas à inclusão e apoio ao desenvolvimento de todos os grupos.

Referências

- ⑩ **Bibliotecas Python**: pandas, matplotlib, seaborn, scikit-learn, mlxtend
- ⑩ **Fontes de Dados**: Dados de ingressantes e formandos em formato Excel e CSV