import pandas as pd

# Carregar o arquivo CSV e remover valores nulos

df = pd.read\_csv('/data/ecommerce\_preparados.csv')

df = df.dropna()

# Cálculo das estatísticas para o campo 'Desconto'

media = df['Desconto'].mean()  # Média dos descontos

mediana = df['Desconto'].median()  # Mediana dos descontos

variancia = df['Desconto'].var()  # Variância dos descontos

desvio\_padrao = df['Desconto'].std()  # Desvio padrão dos descontos

# Como a função mode() retorna uma Série, selecionamos a primeira moda

moda = df['Desconto'].mode()[0]  # Moda dos descontos

minimo = df['Desconto'].min()  # Valor mínimo do desconto

quartis = df['Desconto'].quantile([0.25, 0.5, 0.75])  # Quartis dos descontos

maximo = df['Desconto'].max()  # Valor máximo do desconto

# Exibe os resultados

print("Análise Estatística do Campo 'Desconto':")

print(f"Média: {media}")

print(f"Mediana: {mediana}")

print(f"Variância: {variancia}")

print(f"Desvio Padrão: {desvio\_padrao}")

print(f"Moda: {moda}")

print(f"Mínimo: {minimo}")

print("Quartis:")

print(quartis)

print(f"Máximo: {maximo}")

# Exibe as primeiras linhas do DataFrame

print("\nPrimeiras linhas do DataFrame:")

print(df.head())

import pandas as pd

# Carregar o arquivo CSV e remover valores nulos

df = pd.read\_csv('/data/ecommerce\_preparados.csv')

df = df.dropna()

# Calcula a matriz de correlação entre Qtd\_Vendidos\_Cod e N\_Avaliacoes\_MinMax

corr\_n\_avaliacoes = df[['Qtd\_Vendidos\_Cod', 'N\_Avaliacoes\_MinMax']].corr()

# Calcula a matriz de correlação entre Qtd\_Vendidos\_Cod e Nota\_MinMax

corr\_nota = df[['Qtd\_Vendidos\_Cod', 'Nota\_MinMax']].corr()

# Calcula a matriz de correlação entre Qtd\_Vendidos\_Cod e Preco\_MinMax

corr\_preco = df[['Qtd\_Vendidos\_Cod', 'Preco\_MinMax']].corr()

# Exibe os resultados

print("Matriz de Correlação com N\_Avaliacoes\_MinMax:")

print(corr\_n\_avaliacoes)

print("\nMatriz de Correlação com Nota\_MinMax:")

print(corr\_nota)

print("\nMatriz de Correlação com Preco\_MinMax:")

print(corr\_preco)

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression

from sklearn.metrics import mean\_squared\_error, r2\_score

from math import sqrt

# Carrega os dados

df = pd.read\_csv('/data/ecommerce\_preparados.csv')

df = df.dropna()

# Define as variáveis preditoras e a variável alvo

X = df[['N\_Avaliacoes\_MinMax', 'Nota\_MinMax', 'Preco\_MinMax', 'Desconto\_MinMax', 'Temporada\_Cod', 'Marca\_Freq', 'Material\_Freq']]  # Preditor

Y = df['Qtd\_Vendidos\_Cod']

# Divide os dados em conjuntos de treino e teste

x\_train, x\_test, y\_train, y\_test = train\_test\_split(X, Y, test\_size=0.2, random\_state=42)

# Cria e treina o modelo de Regressão Linear

modelo\_lr = LinearRegression()

modelo\_lr.fit(x\_train, y\_train)

# Faz previsões com o conjunto de teste

y\_prev = modelo\_lr.predict(x\_test)

# Avalia o modelo

r2 = r2\_score(y\_test, y\_prev)

print(f'Coeficiente de Determinação - R²: {r2:.2f}')

rmse = sqrt(mean\_squared\_error(y\_test, y\_prev))

print(f"Raiz do Erro Quadrático Médio - RMSE: {rmse:.2f}")

desvio\_padrao = Y.std()

print(f"Desvio Padrão do campo Qtd\_Vendidos\_Cod: {desvio\_padrao:.2f}")