DS2

import pandas as pd

import numpy as np

import seaborn as sns

import matplotlib.pyplot as plt

from sklearn.preprocessing import PowerTransformer

# 1. Create synthetic dataset

np.random.seed(42)

n = 100

df = pd.DataFrame({

'student\_id': np.arange(1, n+1),

'age': np.random.randint(15, 19, size=n),

'gender': np.random.choice(['Male', 'Female'], size=n),

'math\_score': np.random.normal(70, 15, size=n),

'english\_score': np.random.normal(65, 10, size=n),

'science\_score': np.random.normal(75, 12, size=n),

'attendance\_rate': np.random.beta(5, 1, size=n) \* 100

})

# Introduce ~5% missing values

for col in ['math\_score', 'english\_score', 'attendance\_rate']:

df.loc[df.sample(frac=0.05).index, col] = np.nan

# Introduce ~3% outliers

df.loc[df.sample(frac=0.03).index, 'science\_score'] \*= 1.8

df.loc[df.sample(frac=0.03).index, 'attendance\_rate'] += 50

print("INITIAL DATA (first 5 rows):")

print(df.head(), "\n")

# 2. Scan & impute missing values

print("MISSING VALUES BEFORE:")

print(df.isnull().sum(), "\n")

# Impute numeric with mean

means = df[['math\_score', 'english\_score', 'attendance\_rate']].mean()

df['math\_score'] = df['math\_score'].fillna(means['math\_score'])

df['english\_score'] = df['english\_score'].fillna(means['english\_score'])

df['attendance\_rate'] = df['attendance\_rate'].fillna(means['attendance\_rate'])

print("MISSING VALUES AFTER:")

print(df.isnull().sum(), "\n")

#plt.figure(figsize=(10, 4))

sns.boxplot(data=df[['math\_score', 'english\_score', 'science\_score', 'attendance\_rate']])

plt.title("Boxplot Before Outlier Treatment")

plt.show()

# 3. Detect & treat outliers using IQR clipping

num\_cols = ['math\_score', 'english\_score', 'science\_score', 'attendance\_rate']

for col in num\_cols:

Q1, Q3 = df[col].quantile([0.25, 0.75])

IQR = Q3 - Q1

lower, upper = Q1 - 1.5 \* IQR, Q3 + 1.5 \* IQR

df[col] = df[col].clip(lower, upper)

print("OUTLIERS CLIPPED via IQR. Summary stats now:")

print(df[num\_cols].describe(), "\n")

#plt.figure(figsize=(10, 4))

sns.boxplot(data=df[['math\_score', 'english\_score', 'science\_score', 'attendance\_rate']])

plt.title("Boxplot After Outlier Treatment")

plt.show()

# 4. Transform attendance\_rate to reduce skewness using log1p

print("Skewness BEFORE:", df['attendance\_rate'].skew())

# Apply log1p (log(1 + x)) to avoid issues with 0 values

df['attendance\_rate\_transformed'] = np.log10(df['attendance\_rate'])

print("Skewness AFTER:", df['attendance\_rate\_transformed'].skew(), "\n")

fig, axs = plt.subplots(1, 2, figsize=(10, 4))

axs[0].hist(df['attendance\_rate'], bins=15, color='skyblue', edgecolor='black')

axs[0].set\_title('Original Attendance Rate')

axs[1].hist(df['attendance\_rate\_transformed'], bins=15, color='salmon', edgecolor='black')

axs[1].set\_title('Transformed Attendance Rate')

plt.tight\_layout()

plt.show()

print(df[['attendance\_rate', 'attendance\_rate\_transformed']].head())