EDA-Data Normalization

Data normalization is an essential step in Exploratory Data Analysis (EDA) that involves scaling or transforming the data to a common range, making it easier to analyze and compare. Here's a summary of data normalization for EDA:

Why Normalize Data?

1. **Avoid dominance by large values**: Large values can dominate the analysis, leading to skewed distributions and misleading conclusions.
2. **Improve model performance**: Normalized data can improve the performance of machine learning models, as they are less affected by scale differences.
3. **Enhance interpretability**: By scaling data, you can better understand the relationships between variables.

Types of Data Normalization:

1. **Min-Max Scaler (Normalization)**: Rescales values to a common range (e.g., 0-1).
2. **StandardScaler (Standardization)**: Rescales values to have zero mean and unit variance.
3. **Robust Scaler**: Similar to StandardScaler, but more robust to outliers.

Example in Python using Pandas

Let's consider a simple example with a dataset containing exam scores for students:

```
 import pandas as pd

# Create a sample dataset
data = {
    'Student': ['A', 'B', 'C'],
    'Math Score': [90, 85, 95],
    'English Score': [80, 75, 90]
}

df = pd.DataFrame(data)

print("Original Data:")
print(df)

# Normalize data using Min-Max Scaler
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()
df[['Math Score', 'English Score']] = scaler.fit_transform(df[['Math Score', 'English

print("\nNormalized Data (Min-Max Scaler):")
print(df)
```

Output:

```
 Original Data:
  Student  Math Score  English Score
0       A         90.0           80.0
1       B         85.0           75.0
2       C         95.0           90.0

Normalized Data (Min-Max Scaler):
  Student  Math Score  English Score
0       A        0.8333         0.6667
1       B        0.6957         0.5833
2       C        0.9722         0.7333
```

In this example, we normalized the exam scores using a Min-Max Scaler to rescale them between 0 and 1.

Tips and Variations:

- Choose the appropriate normalization technique based on your dataset's characteristics (e.g., skewed distributions may benefit from logarithmic scaling).
- Consider applying normalization only to specific columns or features.
- Be aware of potential loss of information during normalization, especially for categorical variables.

Curated by Brajesh Kumar