exp 8.1

Okay, here's the experiment report with the AIM, theory, algorithm, code, and other sections, using a dataset I'll create:

Experiment Number: 8

AIM: To demonstrate the identification and handling of outliers in a dataset using the Interquartile Range (IQR) method.

Objectives:

- To understand the concept of outliers and their impact on data analysis.
- To learn how to detect outliers using the IQR method.
- To apply different techniques for handling outliers: removal, capping, and transformation.
- To visualize the effect of outlier handling on the data distribution.

Course Outcomes: CO3, CO5 (These likely relate to data preprocessing, outlier detection, and data transformation)

Resource/Tools: Jupyter Notebook, Python with Pandas, NumPy, and Matplotlib libraries.

Theory:

Outliers are data points that significantly deviate from the other observations in a dataset. They can arise due to measurement errors, data entry mistakes, or genuine extreme values. Outliers can distort statistical analyses, affect model performance, and lead to misleading conclusions.

Interquartile Range (IQR) Method:

The IQR method is a robust technique for outlier detection. It's less sensitive to extreme values than methods based on standard deviation.

Algorithm:

1. **Generate the dataset:** Create a dataset with some outliers.
2. **Calculate quartiles:** Determine the first quartile (Q1) and third quartile (Q3).
3. **Calculate IQR:** Calculate the interquartile range (IQR = Q3 - Q1).
4. **Define bounds:** Set lower and upper bounds for outlier detection using:
    - Lower Bound = Q1 - 1.5 * IQR
    - Upper Bound = Q3 + 1.5 * IQR
5. **Identify outliers:** Find data points outside the bounds.

6. **Handle outliers:** Apply appropriate techniques (removal, capping, transformation).

Code:

Python

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# 1. Generate Dataset
np.random.seed(42)
data = {'Value': np.random.normal(50, 10, 100).tolist() + [100, 120, 130]}
# Add outliers
df = pd.DataFrame(data)

# 2. Calculate Quartiles and IQR
Q1 = df['Value'].quantile(0.25)
Q3 = df['Value'].quantile(0.75)
IQR = Q3 - Q1

# 3. Define Bounds
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR

# 4. Identify Outliers
outliers = ((df['Value'] < lower_bound) | (df['Value'] > upper_bound))

# 5. Handle Outliers (Choose ONE method)

# 5.1 Remove Outliers
df_removed = df[~outliers]

# 5.2 Cap Outliers
df_capped = df.copy()
df_capped['Value'] = np.clip(df_capped['Value'], lower_bound, upper_bound)

# 5.3 Transform Data (Log Transformation)
df_transformed = df.copy()
df_transformed['Value'] = np.log1p(df_transformed['Value'])

# Visualize
plt.figure(figsize=(15, 5))
plt.subplot(1, 4, 1)
```

```
df['Value'].hist(bins=10)
plt.title('Original Data')
plt.subplot(1, 4, 2)
df_removed['Value'].hist(bins=10)
plt.title('Outliers Removed')
plt.subplot(1, 4, 3)
df_capped['Value'].hist(bins=10)
plt.title('Outliers Capped')
plt.subplot(1, 4, 4)
df_transformed['Value'].hist(bins=10)
plt.title('Log Transformed')
plt.show()
```

Output:

The code generates histograms to visualize the distribution of the data:

- **Original Data:** Shows the data with outliers.
- **Outliers Removed:** Shows the data after removing outliers.
- **Outliers Capped:** Shows the data after capping outliers.
- **Log Transformed:** Shows the data after applying log transformation.

Result:

The experiment demonstrates the impact of outliers on the data distribution and the effects of different outlier handling techniques. You can observe how each method changes the shape of the distribution and potentially improves the data quality for analysis.

Learning Outcomes:

- Created a dataset with outliers and applied the IQR method for detection.
- Implemented different outlier handling techniques (removal, capping, transformation).
- Visualized the effects of each technique on the data distribution.
- Gained a better understanding of how to deal with outliers in data analysis.
- Developed skills in using Pandas, NumPy, and Matplotlib for data manipulation and visualization.