**Lab 7: Data Cleaning – Handling Missing Values and Outliers**

**Prelab Questions**

1. **Why is data cleaning important before analysis?**

Data cleaning is like the foundation of a house—it sets the stage for everything that comes after. Before diving into any analysis or building models, it's important to make sure the data you're working with is accurate, consistent, and complete. Here's why it's crucial:

1. **Improves Accuracy**: Bad data can skew your results, leading to incorrect conclusions. Clean data ensures that the analysis reflects reality.
2. **Saves Time and Resources**: By cleaning your data beforehand, you avoid wasting time and effort on analyzing flawed information, which could lead to invalid insights or faulty predictions.
3. **Prevents Bias**: Unclean data might introduce bias into your models. For example, if certain data points are missing or incorrect, it can affect how well the model generalizes to real-world scenarios.
4. **What are the different types of missing data?**

Missing data can be tricky, and it often falls into one of three categories:

1. **Missing Completely at Random (MCAR)**: The missing data has no relationship with any other data in the dataset. It's truly random, so you don’t need to worry too much about how it affects your analysis.
2. **Missing at Random (MAR)**: The missing data is related to other observed variables, but not the missing ones. For example, maybe income data is missing more often from younger age groups. You can handle this type of missingness by analyzing the relationship between other variables.
3. **Missing Not at Random (MNAR)**: The missing data is related to the value of the variable itself. If people with higher incomes are less likely to report their income, that’s a case of MNAR, and handling this can be tricky, as it introduces bias.
4. **What are some techniques for handling missing values in a dataset?**

There are several approaches for dealing with missing values, and the right one depends on the nature of the data and the extent of the missingness:

1. **Remove Missing Data**: If only a small portion of your dataset has missing values, you might simply remove those rows or columns. However, this can lead to data loss, so it's a balancing act.
2. **Imputation**: This involves filling in missing values with plausible guesses. Common methods include:
   1. **Mean/Median/Mode Imputation**: Filling in missing numerical values with the mean or median, and categorical values with the mode.
   2. **K-Nearest Neighbors (KNN)**: Imputing values based on similar observations.
   3. **Regression Imputation**: Using relationships between variables to predict the missing values.
3. **Predictive Modeling**: In some cases, you might use machine learning algorithms to predict the missing values based on the other features in your dataset.
4. **Forward/Backward Filling**: In time series data, missing values can sometimes be filled by using the last known value (forward fill) or the next value (backward fill).
5. **How do outliers affect statistical analysis and machine learning models?**

Outliers can mess up the results of your analysis or mess with machine learning models in several ways:

1. **Distorting Mean and Variance**: Outliers can significantly impact the mean and standard deviation, making them unreliable for analysis. They can create misleading impressions of the "average" data.
2. **Model Performance**: Many machine learning algorithms, especially those that rely on distance metrics (like KNN or linear regression), can be heavily influenced by outliers. A few extreme values can skew the model's predictions and affect its ability to generalize.
3. **Violating Assumptions**: Many statistical tests assume a normal distribution of the data. Outliers can distort these assumptions, leading to inaccurate test results.
4. **What methods can be used to detect outliers in a dataset?**

There are a few techniques you can use to spot outliers:

1. **Visual Inspection**: Plotting the data with box plots, scatter plots, or histograms is an easy and effective way to spot outliers. Box plots, for example, clearly show any data points that lie far beyond the "whiskers" of the plot.
2. **Z-Score**: This is a measure of how many standard deviations away a data point is from the mean. If the Z-score is above a certain threshold (usually 3 or -3), the point is considered an outlier.
3. **IQR (Interquartile Range)**: By calculating the range between the first quartile (Q1) and third quartile (Q3), you can determine if any data points fall outside of the typical range. Data points beyond 1.5 times the IQR from the quartiles are often considered outliers.
4. **Isolation Forest**: This is a machine learning method that isolates anomalies by randomly selecting features and splitting data. It's particularly useful when dealing with high-dimensional datasets.
5. **DBSCAN (Density-Based Spatial Clustering of Applications with Noise)**: A clustering algorithm that can help detect outliers by identifying points that don’t fit into any cluster.

**In-Lab Details**

**Objective**:

* **Identify and handle missing values in a dataset.**
* **Detect and treat outliers to improve data quality.**

**PYTHON SCRIPT:**

import pandas as pd

import numpy as np

import seaborn as sns

import matplotlib.pyplot as plt

# Load the customer data

df = pd.read\_csv(r"C:\Users\DAA-02\Downloads\customer\_data.csv")

# Display the first few rows of the dataset

df.head()

# Check for missing values in each column

missing\_values = df.isnull().sum()

print("Missing values in each column:")

print(missing\_values)

# Handling missing values (example: fill with mean for numerical columns)

df['age'].fillna(df['age'].mean(), inplace=True)

df['income'].fillna(df['income'].mean(), inplace=True)

df['purchase\_frequency'].fillna(df['purchase\_frequency'].mode()[0], inplace=True)

# Check again for missing values

missing\_values\_after = df.isnull().sum()

print("Missing values after handling:")

print(missing\_values\_after)

# Create a boxplot to detect outliers in the income column

plt.figure(figsize=(8, 6))

sns.boxplot(x=df['income'])

plt.title("Income Outliers")

plt.show()

# Calculate the IQR for the income column

Q1 = df['income'].quantile(0.25)

Q3 = df['income'].quantile(0.75)

IQR = Q3 - Q1

# Define the outlier bounds

lower\_bound = Q1 - 1.5 \* IQR

upper\_bound = Q3 + 1.5 \* IQR

# Remove rows where income is an outlier

df\_cleaned = df[(df['income'] >= lower\_bound) & (df['income'] <= upper\_bound)]

# Display cleaned data summary

print("Cleaned Data Summary:")

print(df\_cleaned.describe())

# Show cleaned data (no missing values and no outliers)

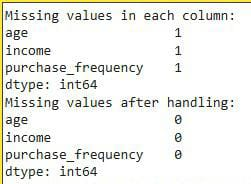
df\_cleaned.head()

**Resources**:

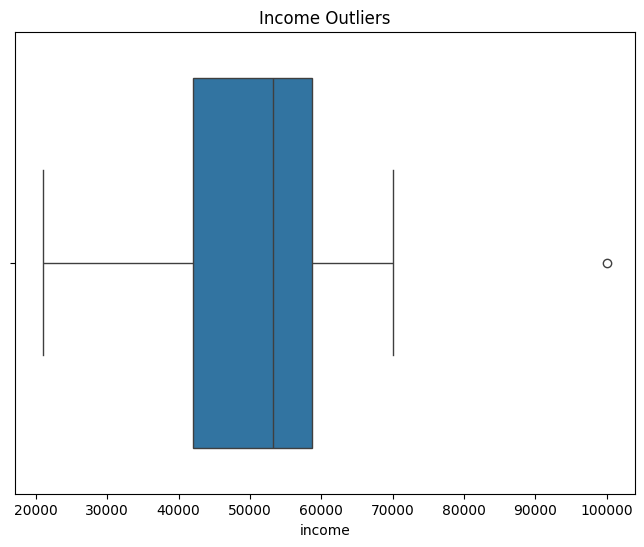
* Python (Jupyter Notebook).
* Libraries: Pandas, NumPy, Seaborn, Matplotlib.
* Dataset: customer\_data.csv containing customer age, income, and purchase frequency.

**Expected Output**:

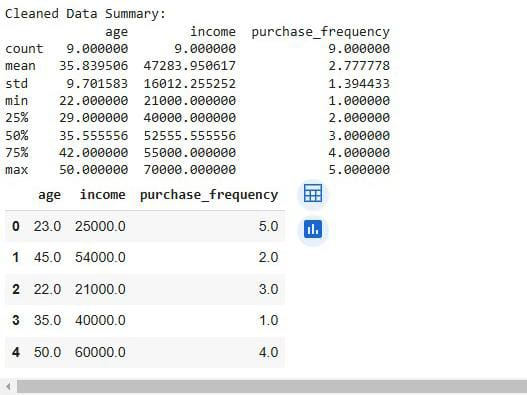
1. **Missing Values Summary**: Displaying columns with missing data.



1. **Boxplot Visualization**: Detecting income outliers.



1. **Cleaned Data Summary**: Dataset with missing values handled and outliers removed.



**Postlab Questions**

1. **Why is it necessary to check for missing values before analysis?**

Checking for missing values before diving into analysis is like making sure your puzzle pieces are all there before starting to assemble it. Missing data can affect the accuracy of your analysis and models, leading to unreliable or biased results. If you don't address missing values upfront, your conclusions could be distorted, and your models might struggle to make meaningful predictions. It’s an important step because missing data can tell you something about your dataset—whether it’s random, systematic, or potentially impactful.

1. **Compare different strategies for handling missing data (mean, median, mode, or removal).**

**When you're faced with missing values, you have a few options to deal with them, each with its pros and cons. Here’s how they stack up:**

1. **Mean Imputation**:
   1. **What it is**: Replacing missing values with the average (mean) of the existing data in the column.
   2. **When to use it**: This works well when your data is fairly symmetrically distributed without extreme values (outliers).
   3. **Downside**: It can distort the data if the variable has outliers, as the mean can be heavily influenced by them. It also doesn’t account for relationships with other variables.
2. **Median Imputation**:
   1. **What it is**: Replacing missing values with the median (the middle value when the data is sorted).
   2. **When to use it**: The median is a great choice when your data has outliers or is skewed, as it’s less sensitive to extreme values than the mean.
   3. **Downside**: Like the mean, it doesn’t capture relationships between features, so it might oversimplify things in some cases.
3. **Mode Imputation**:
   1. **What it is**: Replacing missing values with the most common value (mode) in the data column. This is typically used for categorical variables.
   2. **When to use it**: If your variable is categorical (e.g., color, gender), this is a reasonable option.
   3. **Downside**: It can make the data less diverse and may not be representative of the missing data’s real distribution.
4. **Removal**:
   1. **What it is**: Removing the rows or columns with missing data entirely.
   2. **When to use it**: This is a good option if only a small percentage of data is missing, and removing it doesn’t result in losing too much useful information.
   3. **Downside**: It can lead to data loss, especially if a large chunk of your dataset has missing values. You might also lose valuable insights by discarding incomplete records.
5. **What is the interquartile range (IQR), and how is it used for outlier detection?**

The **Interquartile Range (IQR)** is a measure of statistical dispersion, and it's used for detecting outliers. Here's how it works:

1. **What it is**: The IQR measures the range between the first quartile (Q1) and the third quartile (Q3) in a dataset. In simple terms, it's the range of the middle 50% of the data.
   1. **Q1** is the value below which 25% of the data fall.
   2. **Q3** is the value below which 75% of the data fall.
2. **How it’s used for outlier detection**: Outliers are typically defined as data points that fall more than 1.5 times the IQR below Q1 or above Q3. These values are considered extreme and could potentially be treated as outliers.
   1. **Formula**:
      1. Lower bound: Q1 − 1.5 \* IQR
      2. Upper bound: Q3 + 1.5 \* IQR

Any data points outside this range can be flagged as outliers.

1. **How can extreme outliers affect predictive models like regression?**

Extreme outliers can have a major impact on predictive models like **regression**:

1. **Skewing Results**: In regression, outliers can disproportionately influence the line of best fit, making the model fit the extremes rather than the general trend. This means the model could end up making poor predictions, especially for the majority of data points.
2. **Impact on Coefficients**: Outliers can distort the regression coefficients (like the slope in a linear regression). This means the model may overemphasize or misrepresent the relationship between variables.
3. **Distorting Metrics**: Outliers can mess with key metrics like the **R-squared value** (how well the model fits the data). It could inflate these metrics, giving the false impression that the model is performing better than it actually is.
4. **What alternative techniques can be used to handle outliers instead of removal?**

While removing outliers is one option, it’s often better to try and handle them thoughtfully, especially if you don't want to lose valuable data. Here are a few alternatives:

1. **Transformations**:
   1. **Log Transformation**: Applying a logarithmic transformation can help reduce the effect of outliers by compressing large values, making the distribution more normal.
   2. **Box-Cox Transformation**: This is a more advanced technique that can help stabilize variance and normalize data, particularly for data that’s not normally distributed.
2. **Winsorization**:
   1. This involves replacing outliers with the nearest valid values that fall within a specified range (like replacing extreme values with the highest value within the 95th percentile). It helps in reducing the influence of outliers without completely discarding them.
3. **Imputation**:
   1. If the outliers are due to data errors or noise, you could consider imputing them based on the rest of the data (e.g., using the mean, median, or even prediction models).
4. **Robust Models**:
   1. Some machine learning models, like **robust regression** or **tree-based methods** (like Random Forests), are less sensitive to outliers and can handle them better without skewing results.
5. **Clustering**:
   1. If your dataset has a few extreme outliers, clustering methods like **DBSCAN** can help you identify and treat these outliers by grouping similar data points together and labeling the extreme ones as noise.

DONE BY

Durga S – 22BCS026

Nandhini Priya K K -22BCS073

Nithiyaa T – 22BCS082

Priya Dharshini P – 22BCS094

Rithika S – 22BCS103

Sandhiya G – 22BCS109

Santhiya P – 22BCS111