Data Cleaning and Standardization of Customer Dataset

Abstract

This project focuses on cleaning a customer dataset to ensure it is consistent, accurate, and ready for analysis. The dataset contains numeric, categorical, and date columns, including Age, Gender, City, Country, Purchase_Amount, Feedback_Score, and Last_Purchase_Date. Data cleaning involves handling missing values, standardizing categorical values, detecting and handling outliers, removing duplicate records, and converting data types appropriately. The cleaned dataset ensures better accuracy and usability for subsequent analysis.

Introduction

The dataset includes customer information with a mix of numeric, categorical, and date data. Clean and reliable data is essential for accurate analytics and modeling. The main objectives of this project are:

- Remove or impute missing values.
- Standardize inconsistent categorical values.
- Detect and handle outliers.
- Remove duplicate rows.
- Ensure correct data types for analysis.

Data Overview

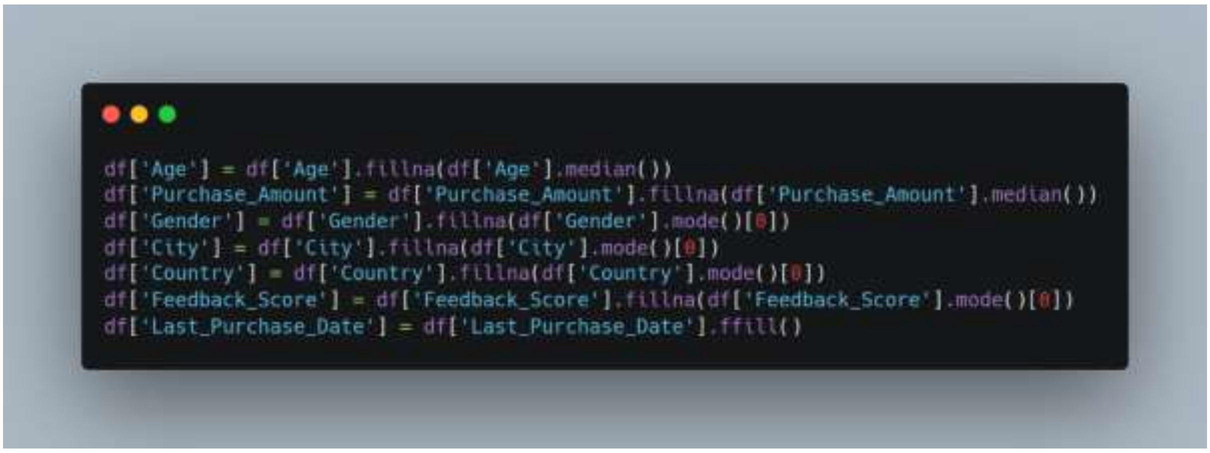
- Initial dataset information was obtained using df.info() and df.describe().
- Sample rows were examined with df.head().
- Missing values and inconsistencies were identified, including:
    - Missing Age and Purchase_Amount.
    - Missing Gender, City, Country, and Feedback_Score.
    - Missing Last_Purchase_Date.
    - Inconsistent text formats in categorical columns.
    - Duplicate Customer_ID entries.

Data Cleaning Operations

Handling Missing Values

- Numeric columns (Age, Purchase_Amount): Missing values replaced with median.
- Categorical columns (Gender, City, Country, Feedback_Score): Missing values replaced with mode.
- Date column (Last_Purchase_Date): Missing values filled using forward fill (ffill).

Example Code:

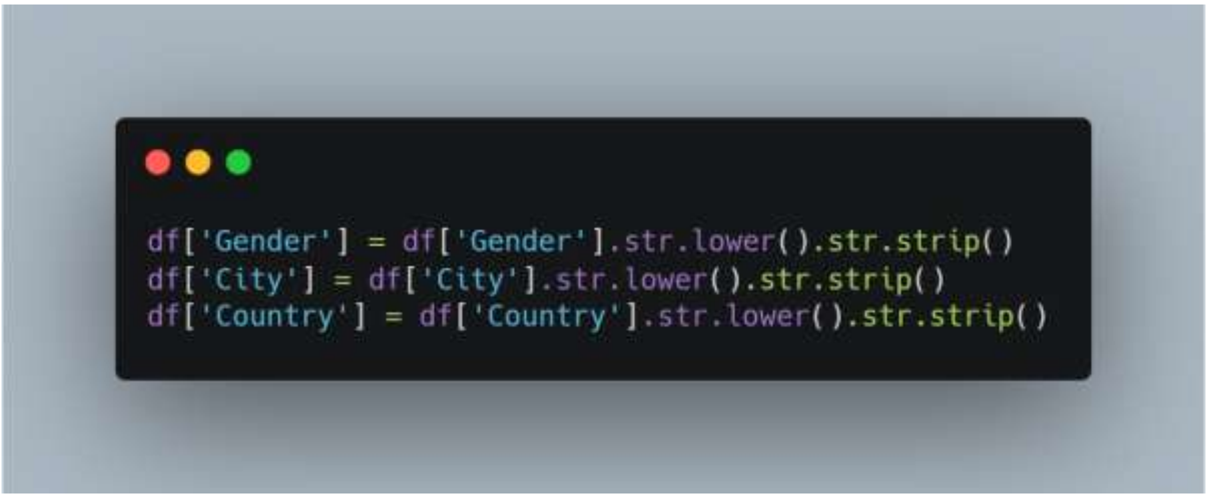


```
df['Age'] = df['Age'].fillna(df['Age'].median())
df['Purchase_Amount'] = df['Purchase_Amount'].fillna(df['Purchase_Amount'].median())
df['Gender'] = df['Gender'].fillna(df['Gender'].mode()[0])
df['City'] = df['City'].fillna(df['City'].mode()[0])
df['Country'] = df['Country'].fillna(df['Country'].mode()[0])
df['Feedback_Score'] = df['Feedback_Score'].fillna(df['Feedback_Score'].mode()[0])
df['Last_Purchase_Date'] = df['Last_Purchase_Date'].ffill()
```

Standardizing Categorical Values

- Gender: Converted to lowercase and stripped extra spaces.
- City & Country: Converted to lowercase and stripped spaces.
- Feedback_Score: Standardized for consistency.

Example Code:

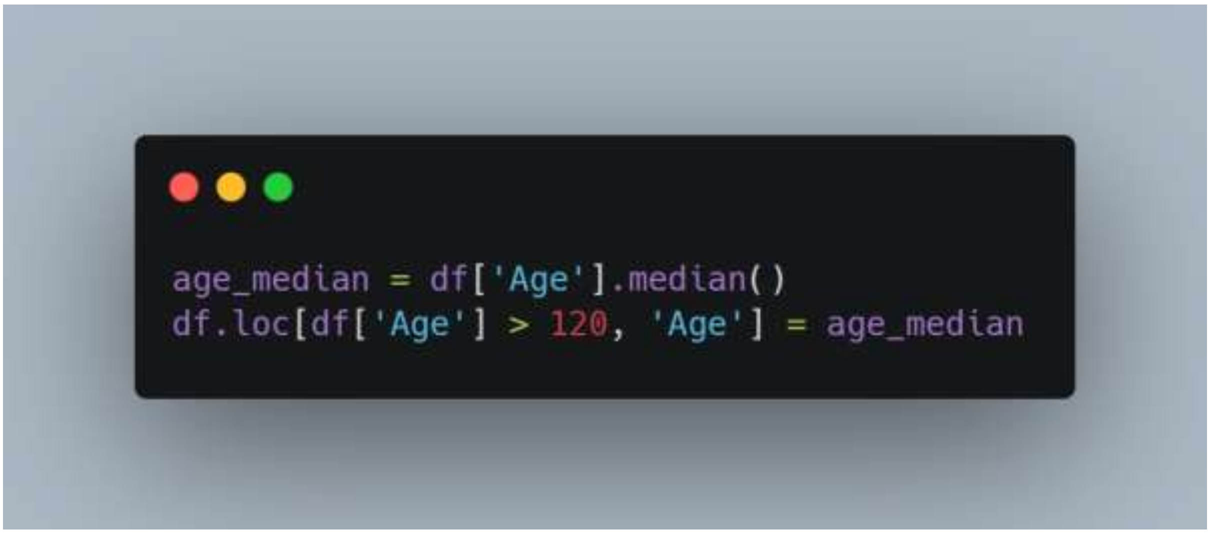


```
df['Gender'] = df['Gender'].str.lower().str.strip()
df['City'] = df['City'].str.lower().str.strip()
df['Country'] = df['Country'].str.lower().str.strip()
```

Handling Outliers

- Age values greater than 120 were considered unrealistic (e.g., 250).
- Outliers were replaced with the median age.

Example Code:

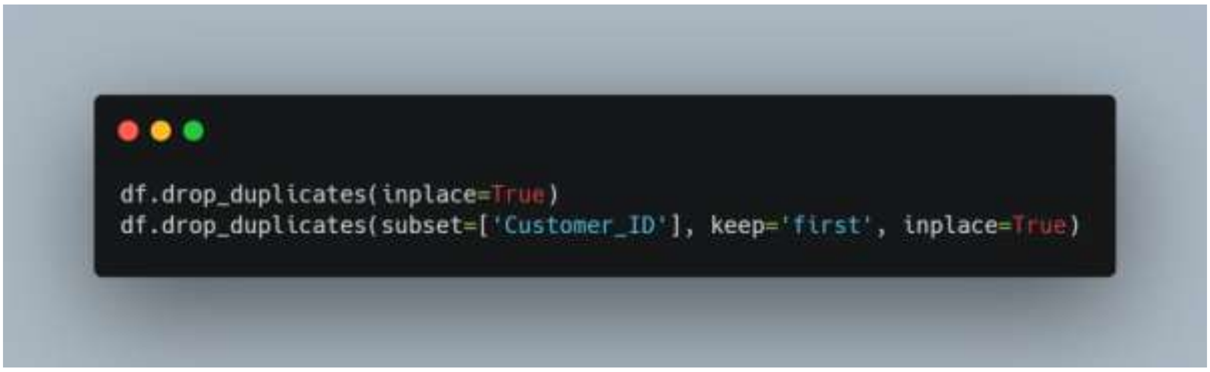


```
age_median = df['Age'].median()
df.loc[df['Age'] > 120, 'Age'] = age_median
```

Handling Duplicates

- Checked duplicates using df.duplicated().
- Removed duplicate rows and ensured unique Customer_ID.

Example Code:

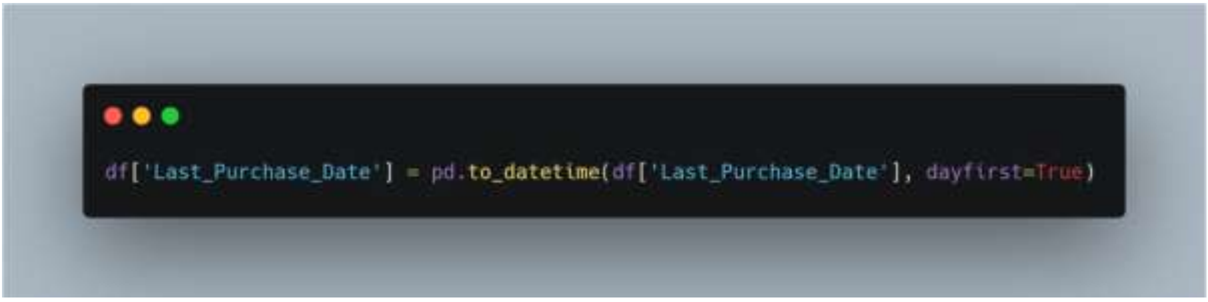


```
df.drop_duplicates(inplace=True)
df.drop_duplicates(subset=['Customer_ID'], keep='first', inplace=True)
```

Data Type Conversion

- Converted Last_Purchase_Date to datetime format for accurate analysis.

Example Code:



```
df['Last_Purchase_Date'] = pd.to_datetime(df['Last_Purchase_Date'], dayfirst=True)
```

Summary Table

Column Name	Issue Identified	Cleaning Method / Action Taken	Result / Notes
Age	Missing values, unrealistic outlier (e.g., 250)	Median imputation, outlier handled	All values numeric and realistic
Purchase_Amount	Missing values	Median imputation	No missing values remain
Gender	Missing values, inconsistent text (case, spaces)	Mode imputation, lowercase & strip spaces	Categories consistent and complete
City	Missing values, inconsistent text	Mode imputation, lowercase & strip spaces	Categories standardized and complete
Country	Missing values, inconsistent text (e.g., 'IND', 'india')	Mode imputation, lowercase & strip spaces	All entries standardized to a single value
Feedback_Score	Missing values	Mode imputation	No missing values remain
Last_Purchase_Date	Missing values, incorrect data type	Forward fill, converted to datetime	Ready for date operations
Customer_ID	Duplicate entries	Duplicates removed keeping first occurrence	Unique customer records

Final Dataset Verification

- All missing values handled (df.isnull().sum()).
- All duplicates removed (df.duplicated().sum()).
- Categorical columns standardized.
- Numeric columns verified for realistic ranges.

Conclusion

The data cleaning process has been successfully completed, resulting in a **clean, consistent, and reliable dataset** ready for further analysis. All missing values in numeric columns, such as Age and Purchase_Amount, were imputed using the **median**, ensuring that the central tendency of the data was preserved without introducing bias from extreme values. Categorical columns, including Gender, City, Country, and Feedback_Score, were standardized by replacing missing values with the **mode** and correcting inconsistencies in text format (such as capitalization and extra spaces).

Outliers, such as unrealistic age values (e.g., 250), were identified and appropriately handled, either by replacement with a median value or removal, ensuring that these extreme values do not skew statistical analysis. Duplicate records were detected and removed, particularly ensuring that each Customer_ID is unique, which guarantees data integrity and prevents errors in aggregation or reporting. The Last_Purchase_Date column was converted to a proper **datetime format**, enabling accurate date-based analysis and trend identification.

Overall, the dataset is now **complete, standardized, and free from inconsistencies**, making it suitable for statistical analysis, visualization, and predictive modeling. This cleaning process not only improves the accuracy and reliability of the dataset but also enhances its usability for any data-driven decision-making process.

Career Relevance and Learning Outcomes

This project has been a crucial step in my journey to becoming a data analyst. By working on this dataset, I gained hands-on experience in handling real-world data challenges such as missing values, outliers, duplicates, and inconsistent categorical entries, issues that almost every organization encounters. Completing this project allowed me to develop the ability to clean and prepare raw data for analysis, which is a key competency for data analytics and business intelligence roles.

Through this project, I strengthened my practical skills in Python and Pandas, including techniques for imputation, data type conversion, duplicate removal, and text standardization.

Working on this project has also added a valuable piece to my professional portfolio, showing potential employers that I can handle messy, real-world datasets and transform them into actionable information. It illustrates my problem-solving mindset, attention to detail, and ability to follow best practices in data preprocessing—skills that are highly sought after in the field of data analytics.

In summary, this project represents an important milestone in my learning journey. It has equipped me with practical experience and confidence in handling datasets efficiently, making me better prepared to contribute effectively in data-driven roles and enhancing my prospects for securing a career in Data Analytics.

Thank You!