IT-462

Exploratory Data Analysis

Assignment-1: Missingno Package

Jash Shah (202201016)

Vraj Gandhi (202201425)

Yatri Rakholiya (202411056)

Introduction:

- Handling missing data is an essential part of data analysis. Missing data generally falls into three categories:
  1. **Missing Completely at Random (MCAR):** No pattern in the missing data. The missingness is independent of any variable in the dataset.
  2. **Missing at Random (MAR):** Missing data is systematically related to other observed data, but not the missing data itself. For example, high-income individuals may be less likely to report their income, making missingness related to other variables like income.
  3. **Not Missing at Random (NMAR):** Missing data is related to the value itself. For example, people with lower income may choose not to disclose their income. NMAR requires domain expertise to confirm and cannot always be inferred from the data itself.
- The `missingno` package in Python helps visualize missing data, aiding in determining how to handle it effectively. These visualizations can offer clues about the type of missing data, enabling data analysts to take appropriate action .
- To install the **missingno** package, use:
  1) `pip install missingno`
  2) `import missingno as msno`

Bar Plot:

- **Purpose:** Displays the count of missing values per column.

- **Explanation:** The bar plot provides a simple overview of how much data is missing in each column. The height of each bar shows the proportion of missing data in that column. This helps identify columns with significant missing data, giving a quick summary for initial exploration.

- **Detailed Insight:** The left y-axis shows data completeness (from 0.0 to 1.0), where 1.0 represents no missing values. The right y-axis shows the number of missing values in actual counts. This dual-scale view allows for easy interpretation, even when dealing with large datasets.

- **Identifying Type of Missing Data:** The bar plot does not directly reveal the type of missing data, but if certain columns have disproportionately higher missing values, it could hint at NMAR. Random or uniform missing values across columns may suggest MCAR.

- **Use Case:** This plot is ideal for getting a quick, high-level overview of which columns require further investigation.
- **Example:** `msno.bar(df)`

Matrix Plot:

- **Purpose:** Provides a matrix-style visualization of missing data across rows.
- **Explanation:** The matrix plot shows missing values as white gaps within the rows of the dataset. This allows you to detect patterns in how missing values are distributed across rows. For example, missing values that cluster at certain points in the dataset could indicate issues with how the data was collected over time or across different groups.
- **Detailed Insight:** A sparkline on the side of the plot provides a summary of data completeness across rows. Rows with no missing values appear on the right of the sparkline, while rows with increasing amounts of missing data shift toward the left.
- **Identifying Type of Missing Data:**
  1) **MCAR:** Missing values appear randomly throughout the dataset without any discernible pattern.
  2) **MAR/NMAR:** Missing values cluster in specific parts of the dataset. For instance, gaps that appear more frequently toward the end of the dataset might indicate MAR, related to time-based factors, or NMAR if specific groups of data have missing values for reasons inherent to the data itself.
- **Use Case:** This plot is particularly useful when working with time-series data or datasets where row order matters, as it reveals whether missing data is concentrated in specific time periods or groups.
- **Example:**
  1) For Normal(Black & White) Visualization: `msno.matrix(df)`
  2) To customize the color palate: `msno.matrix(df,color = (0.2, 0.15, 0.3))`

Heatmap:

- **Purpose:** Shows correlations between missing values across columns.

- **Explanation:** The heatmap displays the relationship between missing values in different columns, showing the strength of the correlation. A high positive correlation (close to +1) indicates that when one column has missing values, another column is also likely to have missing values. A correlation near 0 implies that the missingness in one column is independent of missingness in another, which suggests MCAR.

- **Detailed Insight:** The heatmap is particularly helpful in datasets with many columns. High correlations between missing values in certain columns suggest that they may be related to the same cause, which can help you determine whether the data is MAR. If the correlations are low across the board, this indicates the data is likely MCAR.

- **Identifying Type of Missing Data:**

  1) **MCAR:** Correlations between missing values are near zero, indicating the absence of a relationship between columns.
  2) **MAR:** A moderate to high positive correlation (between 0.3 and 1) indicates that the missingness in one column can be explained by the values in another column.
  3) **NMAR:** This type of missingness is difficult to detect from heatmaps alone, as NMAR typically requires domain knowledge. However, heatmaps can suggest NMAR when no clear correlations are visible, yet certain columns have missing values in a systematic way.

- **Use Case:** This plot is useful when you want to investigate whether missing values in one column are related to missing values in another, helping to distinguish between MCAR and MAR.

- **Example:** `msno.heatmap(df)`

Dendrogram:

- **Purpose:** Shows hierarchical clustering of columns based on missing data patterns.
- **Explanation:** The dendrogram groups columns with similar missing data patterns. Columns that are closely clustered together are more likely to have similar missingness behavior, which might indicate a shared source or cause of missing values. This hierarchical structure is particularly useful for understanding the relationships between variables with missing data.
- **Detailed Insight:** In a dendrogram, the closer the branches are, the more similar the missing data patterns are between columns. For example, if several columns are grouped together, it suggests they share similar patterns of missingness, indicating MAR. Columns that are more separated suggest MCAR, where missingness is random and unrelated.
- **Identifying Type of Missing Data:**

  1) **MCAR:** Columns appear in separate clusters or are more isolated, indicating no relationship in their missing data patterns.
  2) **MAR:** Columns cluster closely together, indicating that their missing values are related.
  3) **NMAR:** If certain columns are isolated with unique missing data patterns that can't be explained by correlations, this might indicate NMAR, especially when external factors influence the missingness.

- **Use Case:** The dendrogram is especially useful for large datasets with many columns. It helps visualize which columns have similar missing data patterns, making it easier to address related columns together.
- **Example:** `msno.dendrogram(df)`

Practical Features of missingno:

In addition to the visualizations, missingno provides several practical features that enhance data cleaning and analysis:

- **Filtering Columns or Rows by Missing Values:** You can filter out columns or rows based on a threshold of missing values, making it easier to focus on only those that require cleaning.
  ```
  msno.dendrogram(df)
  ```
- **Sorting Columns by Missing Values:** Sort columns based on the amount of missing data, helping you prioritize columns for analysis.
  ```
  msno.dendrogram(df)
  ```
- **Resampling for Larger Datasets:** You can increase or decrease the frequency of missing data patterns in matrix plots to simulate larger datasets or simplify visualizations for better clarity.
  ```
  msno.upsample(df, n=2)    #Increasing Freq. of missing data patterns
  msno.downsample(df, n=2)  #Reducing the size of dataset for clarity
  ```

Conclusion:

The missingno package in Python provides a comprehensive suite of tools for visualizing and analyzing missing data. By offering intuitive visualizations like bar plots, matrix plots, heatmaps, and dendrograms, it helps identify patterns of missingness, aiding in determining whether data is MCAR, MAR, or NMAR. Coupled with practical features like filtering, sorting, and geographic plotting, missingno simplifies the entire process of dealing with missing data, making it an essential tool for data scientists and analysts.

GitHub Link: https://github.com/jash0803/DAIICT/tree/main/SEM%205/EDA

References: https://moodle.daiict.ac.in/mod/resource/view.php?id=6157