Project Step - 3
Data Cleaning & Transformation

Harsh Siddhapura
Shivkesh Madasu
Shrenika Soma
Jhansi Alugoju

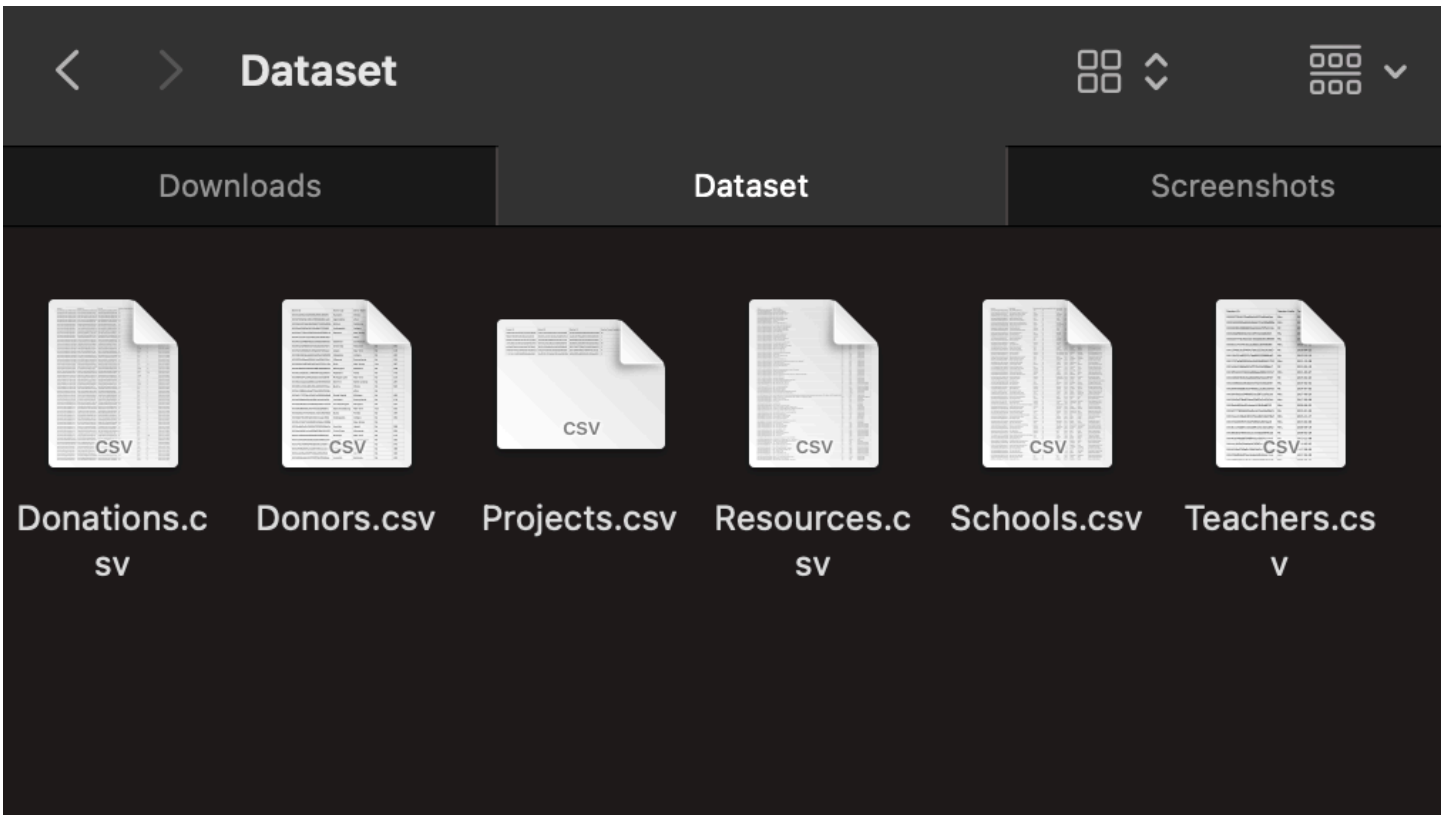
03/10/2024

“Forecasting Philanthropy:
A Predictive Analysis for Donors Supporting Various School Projects in the USA”

Project: Our team will build a regression model that forecasts/predicts analysis for donors to support various projects in different schools in the United States of America. The title of the project is, “Forecasting Philanthropy: A Predictive Analysis for Donors Supporting Various School Projects in the USA”.

++

Save the collected data file as csv.
Google Drive Dataset Link:
https://drive.google.com/drive/folders/1fSaWJX43KUwbZgV0Atm0fDUwaCpfTOU9?usp=sharing



++

1. **A description of the data cleaning steps needed.**

Here is a detailed description of the data cleaning steps performed in the code:

- Define missing values list:
    - A list named missing_values is defined, containing various representations of missing values such as '$', '%', 'null', 'None', '?', empty strings, None, and 'unknown'.

- Replace missing values:
    - The replace() method is used to replace all occurrences of the values in the missing_values list with pd.NA (Pandas' way of representing missing values) in the DataFrame data.

- Check missing values in rows:
    - The isna().sum(axis=1) method calculates the total number of missing values in each row of the DataFrame.
    - The result is then counted to determine how many rows have a specific number of missing values.

- Check missing values in columns:
    - The isna().sum(axis=0) method calculates the total number of missing values in each column of the DataFrame.
    - The count of missing values in columns is then calculated to understand how many columns have a specific number of missing values.

- Drop columns with highest missing values:
    - Two columns, 'Donor City' and 'Donor Zip', are dropped from the DataFrame donors_copy due to having the highest number of missing values.

- Remove rows with missing values:
    - Rows with any missing values are removed from the cleaned DataFrame donors_clean using dropna().

- Re-run data cleaning steps on cleaned data on other files:
    - The data cleaning steps are applied again on the cleaned DataFrame donors_clean to check for any remaining missing values and their distribution in rows and columns.

- Combining data from multiple files:
    - Multiple files containing related data are read into separate DataFrames. These files are in CSV formats.
    - Each DataFrame represents a subset of the overall dataset with potentially overlapping or complementary information.

- Concatenating DataFrames:
    - The individual DataFrames are concatenated or merged together to create a single unified DataFrame that combines all the data.
    - This was achieved using functions like pd.concat() or pd.merge() in pandas, depending on how the data needs to be combined (row-wise or column-wise).

- Dropping Duplicate Rows:
    - After combining the data into a single DataFrame named data, the drop_duplicates() method was applied to remove duplicate rows.
    - Duplicate rows are identified based on all columns in the DataFrame. If two or more rows have exactly the same values across all columns, one of them is considered a duplicate and removed.
    - The inplace=True parameter modifies the original DataFrame data by dropping the duplicate rows directly within it.

- Effect of Dropping Duplicates:
    - Dropping duplicate rows helped in cleaning the dataset by ensuring that each row is unique.
    - It prevents issues like bias in analysis, incorrect statistical calculations, or redundant information in the dataset.
    - The resulting DataFrame after dropping duplicates will have only unique rows, improving the quality and reliability of subsequent analyses or machine learning models.

- Data Integrity and Quality:
    - Removing duplicate rows is a common data cleaning practice to maintain data integrity and ensure accurate analysis results.
    - It helps in eliminating redundant information and maintaining consistency within the dataset.

The results are displayed of missing values in rows, missing values in each column, and total missing values per column are printed before and after cleaning to assess the effectiveness of the data cleaning process.

By following these steps, the code aims to identify, handle, and remove missing values from the dataset to ensure data quality and integrity for further analysis or modeling tasks. It ensures that our dataset is consolidated, free of redundant entries, and ready for further analysis or processing.

++

2. **A description of the data attribute types (Nominal, Ordinal, ... etc.) and the transformation methods used accordingly.**
   In the context of data attribute types and transformation methods used, let's break down the attributes and transformations:

- Categorical Attribute:
  - Attribute Type: Categorical attributes represent qualitative data without any inherent order or ranking.
  - Transformation Method: Frequency encoding is applied to handle large cardinality in the dataset. It replaces each category with its frequency of occurrence in the dataset, which helps reduce dimensionality compared to one-hot encoding.
  - Columns:
    - Project ID
    - Donation ID
    - Donor ID
    - Donation Included Optional Donation
    - School ID
    - Teacher ID
    - Project Type
    - Project Title
    - Project Essay
    - Project Short Description
    - Project Need Statement
    - Project Subject Category Tree
    - Project Subject Subcategory Tree
    - Project Resource Category
    - Donor State
    - Donor Is Teacher
    - School Name
    - School State
    - School Zip
    - School City
    - School County
    - School District
    - Teacher Prefix
    - Resource Item Name

        - Resource Vendor Name

- Ordinal Attribute:
    - Attribute Type: Ordinal attributes have a clear order or ranking among their values.
    - Transformation Method: Unique values are identified, and a dictionary is created to rank them in a specific order. This allows for converting ordinal values into numerical representations based on their predefined order.
    - Columns:
        - Project Grade Level Category
        - Project Current Status
        - School Metro Type

- Interval Attribute:
    - Attribute Type: Interval attributes represent continuous numerical data within a specific range or interval.
    - Transformation Method:
        - pd.to_datetime(d['Last Updated']).min(): Converts the 'Last Updated' column to datetime format and finds the minimum date value in the dataset.
        - pd.to_datetime(d['Last Updated']) - dt.datetime(2010, 5, 21): Calculates the time interval between each date value and a reference date (May 21, 2010).
        - d['Last Updated'] = d['Last Updated'].dt.days: Converts the time interval to days, representing the number of days since May 21, 2010.
    - Columns:
        - Donation Received Date
        - Project Posted Date
        - Project Expiration Date
        - Teacher First Project Posted Date

- Ratio Attribute:

- Attribute Type: Ratio attributes represent numerical data on which various functions can be performed.
- Transformation Method: Creating a new dataframe with ratio attributes.
- Columns:
  - Donation Amount
  - Donor Cart Sequence
  - Teacher Project Posted Sequence
  - Project Cost
  - School Percentage Free Lunch
  - Resource Quantity
  - Resource Unit Price

By understanding the attribute types (categorical, ordinal, interval, ratio) and applying appropriate transformation methods tailored to each type, we can effectively preprocess and transform the data for analysis or modeling purposes. Each transformation method is chosen based on the nature of the attribute and the desired outcome in terms of data representation and analysis requirements.

++

References

1. Kaggle: https://www.kaggle.com/datasets/perkymaster/school-donations/data
2. DonorsChoose: https://www.donorschoose.org/
3. Greater Cedar Rapids Community Foundation: https://www.gcrcf.org/teachers-students/donors-supporting-education/