Project Requiremnt:

Must use Python 3

The input

1. AnomalyData.csv
2. OutlierData.csv
3. MissingData.csv

The Output

1. AnomalyDataSolution.csv
2. OutlierDataSolution.csv
3. MissingDataSolution.csv

Perform graphical and /or non graphical EDA method to understand the data and fixed the data problems.

- Detect and fix errors in AnomalyData.csv
- Detect and remove outliers rows in OutlierData.csv
- Impute the missing values in MissingData.csv

The dataset contains transactional retail data from an online electronics store (DigiCO) located in Melbourne, Australia[1]. The store operation is exclusively online, and it has three warehouses around Melbourne from which goods are delivered to customers.

DESCRIPTION

- A unique id for each order
- A unique id for each customer
- The date the order was made, given in YYYY-MM-DD format
- A string denoting the name of the nearest warehouse to the customer
- Latitude of the customer’s location Longitude of the customer’s location
- An integer denoting the percentage discount to be applied to the order_price.
- A float representing the delivery charges of the order
- A float denoting the total of the order in AUD after all discounts and/or delivery charges are applied.
- A string denoting the season in which the order was placed. Refer to this link for details about how seasons are defined.

Notes:

1. The output *csv* files **must** have the exact same columns as the input.
2. There is at least one anomaly in the dataset from each category of the data anomalies (i.e., syntactic, semantic, and coverage).
3. In the file *AnomalyData.csv,* any row can carry no more than one anomaly.
4. There are no data anomalies in the file *OutlierData.csv,* only outliers. Similarly, there are no data anomalies other than missing value problems in the file *MissingData.csv*
5. The retail store has three different warehouses in Melbourne (see warehouses.csv for their locations)
6. The retail store focuses only on 10 branded items and sells them at competitive prices.
7. A useful python package to solve linear equations is numpy.linalg
8. The store has different business rules depending on the season to match the different demands of each season. For example, delivery charge is calculated using a linear model which differs depending on the season. The model depends linearly (but in different ways for each season) on:

   - Distance between customer and nearest warehouse
   - Whether the customer wants an expedited delivery
   - Whether the customer was happy with his/her last purchase (if no previous purchase, it is assumed that the customer is happy)

9. To check whether a customer is happy with their last order, the customer's latest review is classified using a sentiment analysis classifier. *Sentiment Intensity Analyzer* from nltk.sentiment.vader is used to obtain the polarity score. A sentiment is considered positive if it has a 'compound' polarity score of 0.05 or higher and is considered negative otherwise.
10. If the customer provided a coupon during purchase, the coupon discount percentage will be applied to the order price before adding the delivery charges (i.e. the delivery charges will never be discounted).
11. Also, we know that the following attributes are always correct (i.e. don't look for any errors in dirty data for them):

    - coupon_discount
    - delivery_charges
    - The ordered quantity values in the shopping_cart attribute

The cleaning task must be explained in a well-formatted (with appropriate sections and subsections). Please Explain the complete EDA to examine the data, your methodology to find the data anomalies and the suggested approach to fix those anomalies.