Machine Learning Task

Data Description:

Legacy data (*legacy_data.csv*):

Each record in the dataset represents an individual order that a company receives.

Brief description of the features:

- ***'OrderID'*** - Represents a unique ID assigned to every incoming order
- ***'OrderPrice'*** - Represents the price of the gift opted by the sender
- ***'Amount_Charged'*** - Represents the total amount charged for that order
- ***'Rating'*** - Represents the rating given by the gift recipient to the gift they received
- ***'Preferences'*** - Represents the preferences of the gift recipient based on which a gift box will be curated

Sample Output (*sample_output.csv*):

Each record represents a curated gift box (a gift box is a combination of individual gifts) assigned to each incoming order

Brief description of the features:

- ***'OrderID'*** - Represents a unique ID assigned to every incoming order
- ***'GiftBox'*** - Represents a curated gift box assigned to an order. Products in gift box are represented by a unique number assigned to them known as product ids'. A typical entry in this column is usually a string containing a combination of product ids' (Eg. '3456,4700')

IMPORTANT NOTE: If you find any product id or multiple product ids' which is/are present as a part of the gift box (legacy data and sample output) but absent in the inventory then you can consider that entire specific record in the legacy data and sample output to be invalid and you can drop the same from being considered for model development

Inventory (*inventory.csv*):

Each record in this dataset details the product information about a specific product

Brief description of the features:

- ***'ProductId'*** - Represents an unique number assigned to every product in the inventory
- ***'ProductName'***- Represents the name of the product
- ***'Cost'*** - Represents the value at which the product is procured from vendors
- ***'Price'*** - Represents the value at which the product is sold to the client
- ***'Rating'*** - Represents the average rating of the product given by the clients
- ***'ProductTag'*** - Represents tag(s) assigned to a product to categorise and segregate them broadly

Test Input (*test_input.csv*):

Each record represents a set of inputs we would need a curated gift box to be assigned to

Brief description of the features:

- ***'OrderID'*** - Represents a unique ID assigned to every incoming order
- ***'OrderPrice'*** - Represents the price of the gift opted by the sender
- ***'Preferences'*** - Represents the preferences of the gift recipient based on which a gift box will be curated

IMPORTANT NOTE: You can assume amount charged within a reasonable range and create your column for the same in test input

Task Description:

- Design a recommender system algorithm that can recommend curated gift boxes to each incoming order based on the features detailed in legacy data and inventory datasets.
- The recommender algorithm (that contains the embedded recommender model) should take in the user preferences and gift price as explicit inputs and come up with a minimum of three curated gift boxes as recommendation.
- The recommended gift boxes should be highly rated and the margin percentage obtained from each box should be more than 20%.

NOTE: Formula for margin percent calculation (All prices are in $):
Margin = Total amount charged - Total cost - Shipping fee - Strip fee
Total amount charged = total amount charged ('Amount_Charged' column in the inventory) for that specific order
Total cost = summation of cost ('Cost' column in the inventory) of individual products in a gift box
Shipping fee = 9.5$ in all cases
Strip fee = ((Total amount charged*2.9)/100)+0.3
Margin percent = (Margin * 100)/Total amount charged

Desired Solution Structure:

The solution to entail the following steps

***Data Interpretation and Visualisation*:**

1. Perform exploratory data analysis on the legacy data provided and note the inferences drawn
2. Outlier detection
3. Visualise the the legacy data and note key insights from your observations
4. Visualise the contents of the inventory and note key insights from your observations

***Data cleaning and feature engineering*:**

1. Employ standard cleaning techniques to clean the features you find to be relevant
2. Engineer the features in a way you find to be suitable for drawing maximum performance from the model

***Model Building and Algorithm development*:**

1. Select a suitable algorithm from a plethora of it available from standard python machine learning libraries to build and train a suitable model that can recommend curated gift boxes. Note: Clearly state the reason(s) for selecting a specific algorithm
2. Use a serialisation library/framework to make the model to be portable as a binary entity

***Performance Analysis*:**

1. Employ appropriate performance metrics to clearly denote the performance of the model on a test dataset (provided with the task)
2. Infer your observations on how the model can positively impact the ratings and overall efficiency

Important Instructions:

1. The Solution has to be submitted in two Jupyter notebooks/google colab notebooks. The first notebook should contain the Data Interpretation and Visualisation followed by Data cleaning and feature engineering (named as ‘data_analysis.ipynb'). The second notebook should contain Model Building and Algorithm development followed by Performance Analysis (named as ‘algorithm.ipynb’).
2. The solution notebooks described in the above step should clearly have the output visible (i.e all the cells have to be compulsorily executed)
3. Denote a brief description of the step just above every code snippet about it’s functionality and include comments for individual lines of code wherever appropriate
4. Use python version 3.6 and above for development
5. You can use any standard ML algorithm(s) for model building. If you feel you can develop a better solution using Deep Learning you are free to do so and the same solution structure is to be followed in this case as well as detailed in the heading marked ‘Desired Solution Structure’