Sprint II

Data Pre Processing

Date	5 November 2022
Team ID	PNT2022TMID27576
Project Name	DemandEst - AI powered Food Demand Forecaster

Screenshots:

Data Preprocessing:

1. **Importing the libraries**
2. **Reading the Dataset**
3. **Exploratory Data analysis**
4. **Checking for null values.**
5. **Reading and merging.csv files.**
6. **Dropping columns.**
7. **Label encoding**
8. **Data visualization**
9. **Splitting the dataset into dependent and independent variable.**
10. **Split the dataset into train set and test set**

Importing the Libraries

In [16]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

Reading the Dataset

In [5]:
```
train =  pd.read_csv("../Dataset/train.csv")
test = pd.read_csv("../Dataset/test.csv")
fulfilment_center = pd.read_csv("../Dataset/fulfilment_center_info.csv")
meal_info = pd.read_csv("../Dataset/meal_info.csv")
```

Exploratory Data Analysis

In [6]:
```
train.head()
```

Out[6]:

	id	week	center_id	meal_id	checkout_price	base_price	emailer_for_promotion	homepage_featured	num_orders
0	1379560	1	55	1885	136.83	152.29	0	0	177
1	1466964	1	55	1993	136.83	135.83	0	0	270
2	1346989	1	55	2539	134.86	135.86	0	0	189
3	1338232	1	55	2139	339.50	437.53	0	0	54
4	1448490	1	55	2631	243.50	242.50	0	0	40

In [7]:
```
train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 456548 entries, 0 to 456547
Data columns (total 9 columns):
 #   Column                Non-Null Count   Dtype  
---  ------                --------------   -----  
 0   id                    456548 non-null  int64  
 1   week                  456548 non-null  int64  
 2   center_id             456548 non-null  int64  
 3   meal_id               456548 non-null  int64  
 4   checkout_price        456548 non-null  float64
 5   base_price            456548 non-null  float64
 6   emailer_for_promotion  456548 non-null  int64  
 7   homepage_featured     456548 non-null  int64  
 8   num_orders            456548 non-null  int64  
dtypes: float64(2), int64(7)
memory usage: 31.3 MB
```

In [10]:
```
fulfilment_center.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 77 entries, 0 to 76
Data columns (total 5 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   center_id    77 non-null     int64  
 1   city_code    77 non-null     int64  
 2   region_code  77 non-null     int64  
 3   center_type  77 non-null     object 
 4   op_area      77 non-null     float64
dtypes: float64(1), int64(3), object(1)
memory usage: 3.1+ KB
```

In [12]:
```
meal_info.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 51 entries, 0 to 50
Data columns (total 3 columns):
 #   Column    Non-Null Count  Dtype 
---  ------    --------------  ----- 
 0   meal_id   51 non-null     int64 
 1   category  51 non-null     object
 2   cuisine   51 non-null     object
dtypes: int64(1), object(2)
memory usage: 1.3+ KB
```

In [13]:
```
fulfilment_center.head()
```

Out[13]:

	center_id	city_code	region_code	center_type	op_area
0	11	679	56	TYPE_A	3.7
1	13	590	56	TYPE_B	6.7
2	124	590	56	TYPE_C	4.0
3	66	648	34	TYPE_A	4.1
4	94	632	34	TYPE_C	3.6

In [14]:
```
meal_info.head()
```

Out[14]:

	meal_id	category	cuisine
0	1885	Beverages	Thai
1	1993	Beverages	Thai
2	2539	Beverages	Thai
3	1248	Beverages	Indian
4	2631	Beverages	Indian

Checking for Null Values

In [17]:
```
train.isnull().sum()
```

Out[17]:
```
id                      0
week                    0
center_id               0
meal_id                 0
checkout_price          0
base_price              0
emailer_for_promotion   0
homepage_featured       0
num_orders              0
dtype: int64
```

Reading and Merging .csv files

In [19]:
```
merged_train = pd.merge(train, meal_info, on = "meal_id", how = "outer")
merged_train = pd.merge(merged_train, fulfilment_center, on = "center_id", how = "outer")
merged_train.head()
```

Out[19]:

	id	week	center_id	meal_id	checkout_price	base_price	emailer_for_promotion	homepage_featured	num_orders	category	cuisine	city_code	regic
0	1379560	1	55	1885	136.83	152.29	0	0	177	Beverages	Thai	647	
1	1018704	2	55	1885	135.83	152.29	0	0	323	Beverages	Thai	647	
2	1196273	3	55	1885	132.92	133.92	0	0	96	Beverages	Thai	647	
3	1116527	4	55	1885	135.86	134.86	0	0	163	Beverages	Thai	647	
4	1343872	5	55	1885	146.50	147.50	0	0	215	Beverages	Thai	647	

Dropping Columns

In [20]:
```
merged_train = merged_train.drop(["center_id","meal_id"], axis = 1)
merged_train.head()
```

Out[20]:

	id	week	checkout_price	base_price	emailer_for_promotion	homepage_featured	num_orders	category	cuisine	city_code	region_code	center_type
0	1379560	1	136.83	152.29	0	0	177	Beverages	Thai	647	56	TYPE_C
1	1018704	2	135.83	152.29	0	0	323	Beverages	Thai	647	56	TYPE_C
2	1196273	3	132.92	133.92	0	0	96	Beverages	Thai	647	56	TYPE_C
3	1116527	4	135.86	134.86	0	0	163	Beverages	Thai	647	56	TYPE_C
4	1343872	5	146.50	147.50	0	0	215	Beverages	Thai	647	56	TYPE_C

In [22]:
```
cols = merged_train.columns.tolist()
print(cols)
```

```
['id', 'week', 'checkout_price', 'base_price', 'emailer_for_promotion', 'homepage_featured', 'num_orders', 'category', 'cuisin
e', 'city_code', 'region_code', 'center_type', 'op_area']
```

In [24]:
```
cols = cols[:2] + cols[9:] + cols[7:9] + cols[2:7]
print(cols)
```

```
['id', 'week', 'base_price', 'emailer_for_promotion', 'homepage_featured', 'num_orders', 'cuisine', 'checkout_price', 'city_cod
e', 'region_code', 'center_type', 'op_area', 'category']
```

In [25]:
```
train_final = merged_train[cols]
train_final.dtypes
```

Out[25]:
```
id                        int64
week                      int64
base_price              float64
emailer_for_promotion     int64
homepage_featured         int64
num_orders                int64
cuisine                  object
checkout_price          float64
city_code                 int64
region_code               int64
center_type              object
op_area                 float64
category                 object
dtype: object
```

Label Encoding

In [27]:
```
from sklearn.preprocessing import LabelEncoder
lb1 = LabelEncoder()
train_final["center_type"] = lb1.fit_transform(train_final["center_type"])

lb2 = LabelEncoder()
train_final["category"] = lb1.fit_transform(train_final["category"])

lb3 = LabelEncoder()
train_final["cuisine"] = lb1.fit_transform(train_final["cuisine"])

train_final.head()
```

Out[27]:

	id	week	base_price	emailer_for_promotion	homepage_featured	num_orders	cuisine	checkout_price	city_code	region_code	center_type	op_area
0	1379560	1	152.29	0	0	177	3	136.83	647	56	2	2.0
1	1018704	2	152.29	0	0	323	3	135.83	647	56	2	2.0
2	1196273	3	133.92	0	0	96	3	132.92	647	56	2	2.0
3	1116527	4	134.86	0	0	163	3	135.86	647	56	2	2.0
4	1343872	5	147.50	0	0	215	3	146.50	647	56	2	2.0

In [29]:
```
train_final.shape
```

Out[29]:
```
(456548, 13)
```

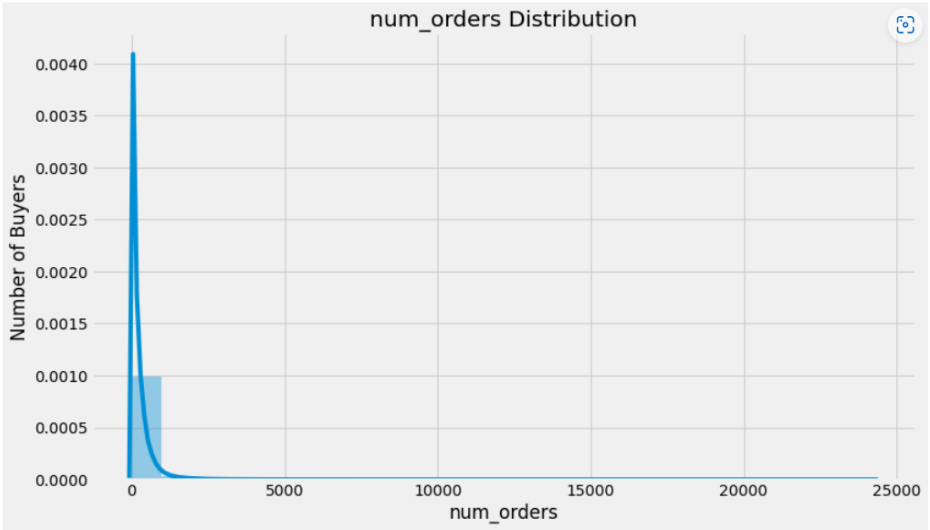
Data Visualization

Univariate Analysis

In [30]:
```
plt.style.use('fivethirtyeight')
plt.figure(figsize=(12,7))
sns.distplot(train_final['num_orders'], bins = 25)
plt.xlabel("num_orders")
plt.ylabel("Number of Buyers")
plt.title("num_orders Distribution")
```

```
C:\Users\joeki\anaconda3\lib\site-packages\seaborn\distributions.py:2557: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[30]:
```
Text(0.5, 1.0, 'num_orders Distribution')
```

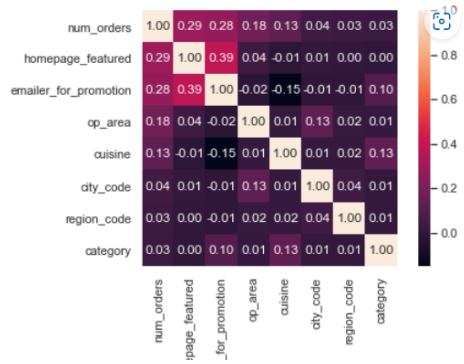


Bivariate Analysis

In [31]:
```
train_final2 = train_final.drop(['id'], axis=1)
correlation = train_final2.corr(method='pearson')
columns = correlation.nlargest(8, 'num_orders').index
columns
```

Out[31]:
```
Index(['num_orders', 'homepage_featured', 'emailer_for_promotion', 'op_area',
       'cuisine', 'city_code', 'region_code', 'category'],
      dtype='object')
```

In [32]:
```
correlation_map = np.corrcoef(train_final2[columns].values.T)
sns.set(font_scale=1.0)
heatmap = sns.heatmap(correlation_map, cbar=True, annot=True, square=True, fmt='.2f', yticklabels=columns.values, xticklabels=col
plt.show()
```



Splitting the Dataset into Dependent and Independent Variable

In [33]:
```
features = columns.drop(['num_orders'])
train_final3 = train_final[features]
X = train_final3.values
y = train_final['num_orders'].values
train_final3.head()
```

Out[33]:

	homepage_featured	emailer_for_promotion	op_area	cuisine	city_code	region_code	category
0	0	0	2.0	3	647	56	0
1	0	0	2.0	3	647	56	0
2	0	0	2.0	3	647	56	0
3	0	0	2.0	3	647	56	0
4	0	0	2.0	3	647	56	0

Split the Dataset Into Train Set And Test Set

In [37]:
```
from sklearn.model_selection import train_test_split
X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.25)
X_train
```

Out[37]:
```
array([[  0. ,   0. ,   2.9, ..., 526. ,  34. ,   0. ],
       [  0. ,   0. ,   3.9, ..., 620. ,  77. ,   5. ],
       [  0. ,   1. ,   4. , ..., 526. ,  34. ,   0. ],
       ...,
       [  0. ,   0. ,   5.1, ..., 590. ,  56. ,  12. ],
       [  0. ,   0. ,   2.8, ..., 699. ,  85. ,   0. ],
       [  1. ,   1. ,   4.2, ..., 615. ,  34. ,   2. ]])
```