predicting-sales-of-a-retail-store

July 18, 2023

[1]:
```
import pandas as pd      #importing dependencies
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.preprocessing import OneHotEncoder
from sklearn.model_selection import train_test_split
from sklearn.metrics import r2_score
from sklearn import linear_model
```

[2]:
```
df1 = pd.read_csv(r'C:\Users\lenovo\Downloads\train_data.csv',encoding =␣
 ↪"unicode escape")   #loading required datasets
df2 = pd.read_csv(r'C:\Users\lenovo\Downloads\date_to_week_id_map.csv',encoding␣
 ↪= "unicode escape")
df3 = pd.read_csv(r'C:\Users\lenovo\Downloads\product_prices.csv',encoding =␣
 ↪"unicode escape")
```

[3]:
```
df_productprices = pd.merge(df3,df2,on=['week_id']) # merging dataset based on␣
 ↪common column
df_main = pd.
 ↪merge(df1,df_productprices,on=['outlet','product_identifier','date'])
```

0.0.1 BASIC EDA

[4]:
```
df_main
```

[4]:

	date	product_identifier	department_identifier	\
0	2012-01-01	74	11	
1	2012-01-01	337	11	
2	2012-01-01	423	12	
3	2012-01-01	432	12	
4	2012-01-01	581	21	
…	…	…	…	
394995	2014-02-28	2932	33	
394996	2014-02-28	2935	33	
394997	2014-02-28	3004	33	
394998	2014-02-28	3008	33	
394999	2014-02-28	3021	33	

```
                category_of_product  outlet        state  sales  week_id  \
0                            others     111  Maharashtra      0       49
1                            others     111  Maharashtra      1       49
2                            others     111  Maharashtra      0       49
3                            others     111  Maharashtra      0       49
4       fast_moving_consumer_goods     111  Maharashtra      0       49
…                              …     …          …    …       …
394995             drinks_and_food     333       Kerala      2      161
394996             drinks_and_food     333       Kerala      8      161
394997             drinks_and_food     333       Kerala      0      161
394998             drinks_and_food     333       Kerala      0      161
394999             drinks_and_food     333       Kerala      0      161

        sell_price
0             2.94
1             7.44
2             0.97
3             4.97
4             4.88
…              …
394995        2.78
394996        0.20
394997        2.50
394998        1.98
394999        2.08

[395000 rows x 9 columns]
```

[5]:
```
df = df_main.copy()
```

[6]:
```
def columns_info(df_main):      # Summary of dataset
    colms= []
    dtypes=[]
    unique=[]
    nunique= []
    nulls = []

    for colm in df_main.columns:
        colms.append(colm)
        dtypes.append(df_main[colm].dtypes)
        unique.append(df_main[colm].unique())
        nunique.append(df_main[colm].nunique())
        nulls.append(df_main[colm].isna().sum())

    return pd.DataFrame ({'columns' :colms,
                          'Datatypes': dtypes,
```

```
                            'Unique Values': unique,
                            'No. Of Unique values' : nunique,
                             'Missing values' : nulls})
columns_info(df_main)
```

[6]:

	columns	Datatypes	Unique Values	No. Of Unique values	Missing values
0	date	object	[2012-01-01, 2012-01-02, 2012-01-03, 2012-01-0…	790	0
1	product_identifier	int64	[74, 337, 423, 432, 581, 611, 631, 659, 743, 7…	50	0
2	department_identifier	int64	[11, 12, 21, 22, 31, 33]	6	0
3	category_of_product	object	[others, fast_moving_consumer_goods, drinks_an…	3	0
4	outlet	int64	[111, 112, 113, 114, 221, 222, 223, 331, 332, …	10	0
5	state	object	[Maharashtra, Telangana, Kerala]	3	0
6	sales	int64	[0, 1, 3, 2, 9, 5, 8, 18, 12, 28, 4, 6, 27, 7,…	126	0
7	week_id	int64	[49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 6…	113	0
8	sell_price	float64	[2.94, 7.44, 0.97, 4.97, 4.88, 2.84, 6.97, 3.9…	107	0

[7]:
```
df_main['Month'] = pd.to_datetime(df_main['date']).dt.month #adding a month␣
↪component using date column
```

[8]:
```
df_main = df_main.drop(columns=["date","week_id"]) #Dropping the unwanted␣
↪columns
df_main
```

[8]:

	product_identifier	department_identifier	category_of_product
0	74	11	others

```
1                     337                    11                      others
2                     423                    12                      others
3                     432                    12                      others
4                     581                    21   fast_moving_consumer_goods
…                     …                      …                       …
394995                2932                   33              drinks_and_food
394996                2935                   33              drinks_and_food
394997                3004                   33              drinks_and_food
394998                3008                   33              drinks_and_food
394999                3021                   33              drinks_and_food

        outlet        state  sales  sell_price  Month
0          111  Maharashtra      0        2.94      1
1          111  Maharashtra      1        7.44      1
2          111  Maharashtra      0        0.97      1
3          111  Maharashtra      0        4.97      1
4          111  Maharashtra      0        4.88      1
…          …            …      …           …      …
394995     333       Kerala      2        2.78      2
394996     333       Kerala      8        0.20      2
394997     333       Kerala      0        2.50      2
394998     333       Kerala      0        1.98      2
394999     333       Kerala      0        2.08      2

[395000 rows x 8 columns]
```

[9]:
```
df_main.describe()     #getting statistical values of dataset
```

[9]:
```
       product_identifier  department_identifier         outlet  \
count       395000.000000          395000.000000  395000.000000
mean          1509.960000              24.460000     211.200000
std            809.799518               6.337863      91.161291
min             74.000000              11.000000     111.000000
25%            926.000000              21.000000     113.000000
50%           1325.000000              22.000000     221.500000
75%           1753.000000              31.000000     331.000000
max           3021.000000              33.000000     333.000000

                sales     sell_price          Month
count  395000.000000  395000.000000  395000.000000
mean        1.228919       4.987644       6.143038
std         3.595266       3.874444       3.576092
min         0.000000       0.050000       1.000000
25%         0.000000       2.680000       3.000000
50%         0.000000       3.980000       6.000000
75%         1.000000       6.480000       9.000000
max       293.000000      44.360000      12.000000
```

[10]:
```
correlation_matrix = df_main.corr()
correlation_matrix
```

```
C:\Users\LENOVO\AppData\Local\Temp\ipykernel_964\222542439.py:1: FutureWarning:
The default value of numeric_only in DataFrame.corr is deprecated. In a future
version, it will default to False. Select only valid columns or specify the
value of numeric_only to silence this warning.
  correlation_matrix = df_main.corr()
```

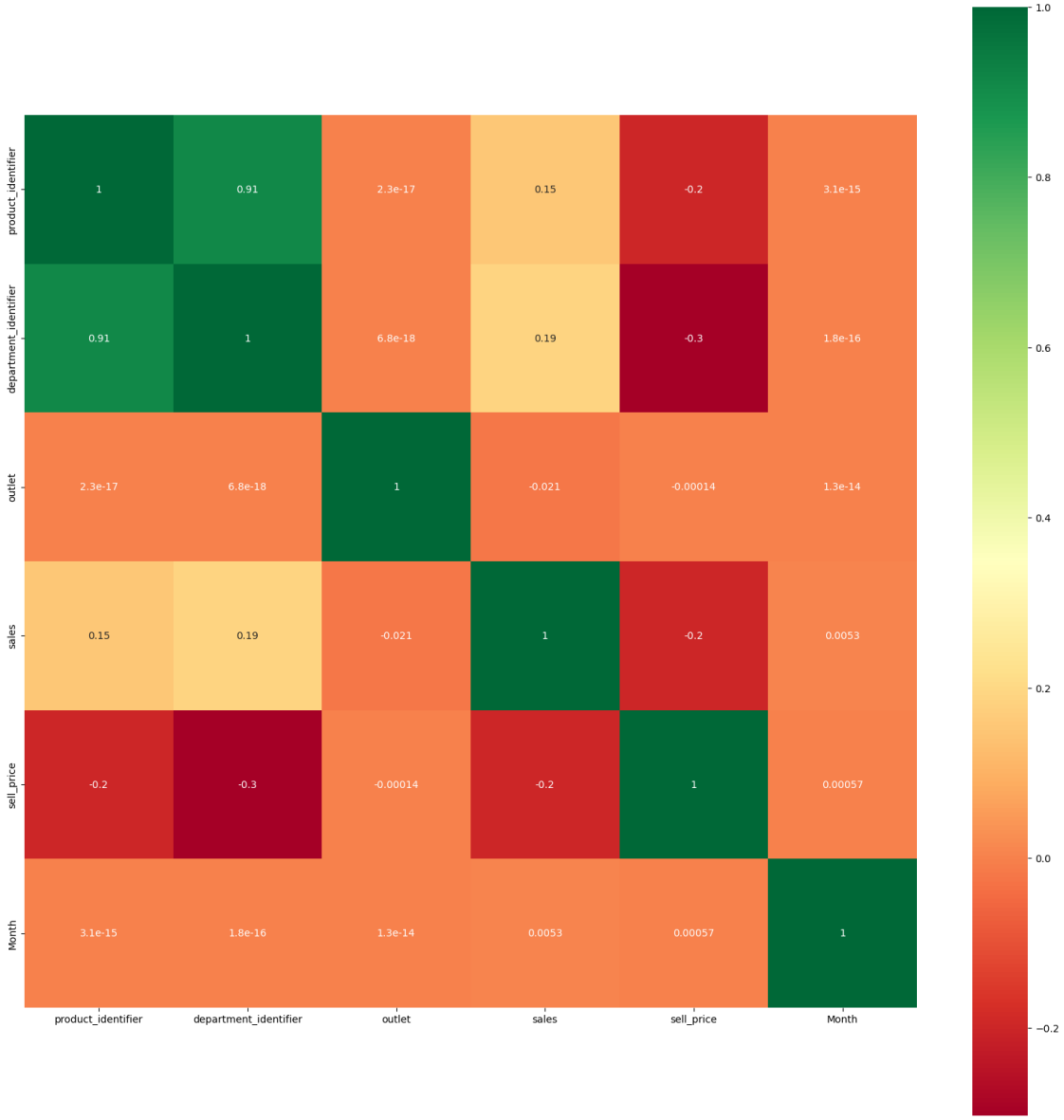
[10]:

	product_identifier	department_identifier	\
product_identifier	1.000000e+00	9.099899e-01	
department_identifier	9.099899e-01	1.000000e+00	
outlet	2.286007e-17	6.817693e-18	
sales	1.528933e-01	1.900381e-01	
sell_price	-2.010739e-01	-3.028601e-01	
Month	3.090176e-15	1.808373e-16	

	outlet	sales	sell_price	Month
product_identifier	2.286007e-17	0.152893	-0.201074	3.090176e-15
department_identifier	6.817693e-18	0.190038	-0.302860	1.808373e-16
outlet	1.000000e+00	-0.021005	-0.000140	1.284597e-14
sales	-2.100456e-02	1.000000	-0.198098	5.262983e-03
sell_price	-1.403172e-04	-0.198098	1.000000	5.675647e-04
Month	1.284597e-14	0.005263	0.000568	1.000000e+00

[11]:
```
plt.figure(figsize=(20,20))
p=sns.heatmap(df_main.corr(), annot=True,cmap='RdYlGn',square=True)
```

```
C:\Users\LENOVO\AppData\Local\Temp\ipykernel_964\63796267.py:2: FutureWarning:
The default value of numeric_only in DataFrame.corr is deprecated. In a future
version, it will default to False. Select only valid columns or specify the
value of numeric_only to silence this warning.
  p=sns.heatmap(df_main.corr(), annot=True,cmap='RdYlGn',square=True)
```



[12]:
```
df_main.drop('department_identifier',axis=1,inplace=True)  # droppinh higher␣
↪co-related column
```

[13]:
```
prefix_col = ['category_of_product', 'state']          #converting categorical␣
↪values into numericals
dummy_col = ['category_of_product', 'state']
df_main = pd.get_dummies(df_main, prefix = prefix_col, columns = dummy_col)
df_main
```

[13]:

	product_identifier	outlet	sales	sell_price	Month	\
0	74	111	0	2.94	1	

```
1                     337     111      1        7.44      1
2                     423     111      0        0.97      1
3                     432     111      0        4.97      1
4                     581     111      0        4.88      1
…                    …     …    …        …    …
394995               2932     333      2        2.78      2
394996               2935     333      8        0.20      2
394997               3004     333      0        2.50      2
394998               3008     333      0        1.98      2
394999               3021     333      0        2.08      2

        category_of_product_drinks_and_food  \
0                                         0
1                                         0
2                                         0
3                                         0
4                                         0
…                                        …
394995                                    1
394996                                    1
394997                                    1
394998                                    1
394999                                    1

        category_of_product_fast_moving_consumer_goods  \
0                                                    0
1                                                    0
2                                                    0
3                                                    0
4                                                    1
…                                                   …
394995                                               0
394996                                               0
394997                                               0
394998                                               0
394999                                               0

        category_of_product_others  state_Kerala  state_Maharashtra  \
0                                1             0                  1
1                                1             0                  1
2                                1             0                  1
3                                1             0                  1
4                                0             0                  1
…                               …            …                 …
394995                           0             1                  0
394996                           0             1                  0
394997                           0             1                  0
```

```
394998                          0              1                    0
394999                          0              1                    0

        state_Telangana
0                     0
1                     0
2                     0
3                     0
4                     0
…                   …
394995                0
394996                0
394997                0
394998                0
394999                0

[395000 rows x 11 columns]
```

[14]:
```
X = df_main.drop(columns= 'sales',axis=1)  #Splitting target & features
Y = df_main['sales']
```

[15]:
```
print(X)
```

```
        product_identifier  outlet  sell_price  Month  \
0                       74     111        2.94      1
1                      337     111        7.44      1
2                      423     111        0.97      1
3                      432     111        4.97      1
4                      581     111        4.88      1
…                      …       …          …     …
394995                2932     333        2.78      2
394996                2935     333        0.20      2
394997                3004     333        2.50      2
394998                3008     333        1.98      2
394999                3021     333        2.08      2

        category_of_product_drinks_and_food  \
0                                         0
1                                         0
2                                         0
3                                         0
4                                         0
…                                       …
394995                                    1
394996                                    1
394997                                    1
394998                                    1
```

```
394999                                    1

        category_of_product_fast_moving_consumer_goods  \
0                                                    0
1                                                    0
2                                                    0
3                                                    0
4                                                    1
…                                                  …
394995                                               0
394996                                               0
394997                                               0
394998                                               0
394999                                               0

        category_of_product_others  state_Kerala  state_Maharashtra  \
0                                1             0                  1
1                                1             0                  1
2                                1             0                  1
3                                1             0                  1
4                                0             0                  1
…                              …             …                  …
394995                           0             1                  0
394996                           0             1                  0
394997                           0             1                  0
394998                           0             1                  0
394999                           0             1                  0

        state_Telangana
0                     0
1                     0
2                     0
3                     0
4                     0
…                   …
394995                0
394996                0
394997                0
394998                0
394999                0

[395000 rows x 10 columns]
```

[16]:
```
print(Y)
```

```
0         0
1         1
2         0
```

```
3         0
4         0
         ..
394995    2
394996    8
394997    0
394998    0
394999    0
Name: sales, Length: 395000, dtype: int64
```

[17]:
```
X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size=0.3,␣
↪random_state=42)
```

[18]:
```
print(X.shape, X_train.shape, X_test.shape)
```

```
(395000, 10) (276500, 10) (118500, 10)
```

[19]:
```
reg = linear_model.LinearRegression()
ml_model1 = reg.fit(X_train, Y_train)
```

[20]:
```
print ("R^2 is: \n", ml_model1.score(X_train, Y_train))
```

```
R^2 is:
 0.06601050367518246
```

[21]:
```
from sklearn.tree import DecisionTreeRegressor
```

[22]:
```
reg2 = DecisionTreeRegressor()
ml_model2 = reg2.fit(X_train, Y_train)
```

[23]:
```
print ("R^2 is: \n", ml_model2.score(X_train, Y_train))
```

```
R^2 is:
 0.4290958264240916
```

[24]:
```
from sklearn.ensemble import RandomForestRegressor
```

[25]:
```
reg3 = RandomForestRegressor()
ml_model3 = reg3.fit(X_train, Y_train)
```

[26]:
```
print ("R^2 is: \n", ml_model3.score(X_train, Y_train))
```

```
R^2 is:
 0.42893549567727296
```

[27]:
```
pip install xgboost
```

```
Requirement already satisfied: xgboost in c:\users\lenovo\anaconda3\lib\site-
packages (1.7.6)
Requirement already satisfied: numpy in c:\users\lenovo\anaconda3\lib\site-
packages (from xgboost) (1.24.3)
Requirement already satisfied: scipy in c:\users\lenovo\anaconda3\lib\site-
packages (from xgboost) (1.10.1)
Note: you may need to restart the kernel to use updated packages.
```

[28]:
```
import xgboost as xgb
```

[29]:
```
reg4 = xgb.XGBRegressor()
ml_model4 = reg4.fit(X_train, Y_train)
```

[30]:
```
print ("R^2 is: \n", ml_model4.score(X_train, Y_train))
```

```
R^2 is:
 0.4175445431250323
```

0.0.2 Creating a subset from main dataset to check if r2 score increases

[31]:
```
df['date'] = pd.to_datetime(df['date'])
Subset = df[(df['date'].dt.year == 2013)& (df['state'] == 'Maharashtra')]
Subset
```

[31]:

	date	product_identifier	department_identifier	category_of_product	outlet	state	sales	week_id
183000	2013-01-01	74	11	others	111	Maharashtra	0	101
183001	2013-01-01	337	11	others	111	Maharashtra	3	101
183002	2013-01-01	423	12	others	111	Maharashtra	0	101
183003	2013-01-01	432	12	others	111	Maharashtra	0	101
183004	2013-01-01	581	21	fast_moving_consumer_goods	111	Maharashtra	1	101
…	…	…	…	…	…	…	…	…
365195	2013-12-31	2932	33	drinks_and_food	114	Maharashtra	2	153
365196	2013-12-31	2935	33	drinks_and_food	114	Maharashtra	4	153
365197	2013-12-31	3004	33	drinks_and_food	114	Maharashtra	2	153
365198	2013-12-31	3008	33					
365199	2013-12-31	3021	33					

```
365198             drinks_and_food     114  Maharashtra      0      153
365199             drinks_and_food     114  Maharashtra      0      153

        sell_price
183000        3.43
183001        6.98
183002        0.97
183003        4.97
183004        4.88
…              …
365195        2.78
365196        0.20
365197        2.50
365198        1.98
365199        2.08

[73000 rows x 9 columns]
```

[32]:
```
Subset1 = Subset.copy()
```

[33]:
```
prefix_col = ['category_of_product']
dummy_col = ['category_of_product']
Subset1 = pd.get_dummies(Subset1, prefix = prefix_col, columns = dummy_col)
Subset1
```

[33]:
```
             date  product_identifier  department_identifier  outlet  \
183000 2013-01-01                  74                     11     111
183001 2013-01-01                 337                     11     111
183002 2013-01-01                 423                     12     111
183003 2013-01-01                 432                     12     111
183004 2013-01-01                 581                     21     111
…             …                   …                      …      …
365195 2013-12-31                2932                     33     114
365196 2013-12-31                2935                     33     114
365197 2013-12-31                3004                     33     114
365198 2013-12-31                3008                     33     114
365199 2013-12-31                3021                     33     114

              state  sales  week_id  sell_price  \
183000  Maharashtra      0      101        3.43
183001  Maharashtra      3      101        6.98
183002  Maharashtra      0      101        0.97
183003  Maharashtra      0      101        4.97
183004  Maharashtra      1      101        4.88
…               …    …       …          …
365195  Maharashtra      2      153        2.78
365196  Maharashtra      4      153        0.20
```

```
365197  Maharashtra      2      153        2.50
365198  Maharashtra      0      153        1.98
365199  Maharashtra      0      153        2.08

        category_of_product_drinks_and_food  \
183000                                    0
183001                                    0
183002                                    0
183003                                    0
183004                                    0
…                                       …
365195                                    1
365196                                    1
365197                                    1
365198                                    1
365199                                    1

        category_of_product_fast_moving_consumer_goods  \
183000                                               0
183001                                               0
183002                                               0
183003                                               0
183004                                               1
…                                                  …
365195                                               0
365196                                               0
365197                                               0
365198                                               0
365199                                               0

        category_of_product_others
183000                           1
183001                           1
183002                           1
183003                           1
183004                           0
…                              …
365195                           0
365196                           0
365197                           0
365198                           0
365199                           0

[73000 rows x 11 columns]
```

[34]:
```
Subset1['Month'] = pd.to_datetime(Subset1['date']).dt.month #adding a month component using date column
```

```
Subset1
```

[34]:
```
             date  product_identifier  department_identifier  outlet  \
183000 2013-01-01                  74                     11     111
183001 2013-01-01                 337                     11     111
183002 2013-01-01                 423                     12     111
183003 2013-01-01                 432                     12     111
183004 2013-01-01                 581                     21     111
…             …                   …                      …      …
365195 2013-12-31                2932                     33     114
365196 2013-12-31                2935                     33     114
365197 2013-12-31                3004                     33     114
365198 2013-12-31                3008                     33     114
365199 2013-12-31                3021                     33     114

              state  sales  week_id  sell_price  \
183000  Maharashtra      0      101        3.43
183001  Maharashtra      3      101        6.98
183002  Maharashtra      0      101        0.97
183003  Maharashtra      0      101        4.97
183004  Maharashtra      1      101        4.88
…               …    …       …          …
365195  Maharashtra      2      153        2.78
365196  Maharashtra      4      153        0.20
365197  Maharashtra      2      153        2.50
365198  Maharashtra      0      153        1.98
365199  Maharashtra      0      153        2.08

        category_of_product_drinks_and_food  \
183000                                    0
183001                                    0
183002                                    0
183003                                    0
183004                                    0
…                                       …
365195                                    1
365196                                    1
365197                                    1
365198                                    1
365199                                    1

        category_of_product_fast_moving_consumer_goods  \
183000                                               0
183001                                               0
183002                                               0
183003                                               0
183004                                               1
```

```
…                                                 …
365195                                              0
365196                                              0
365197                                              0
365198                                              0
365199                                              0

        category_of_product_others  Month
183000                           1      1
183001                           1      1
183002                           1      1
183003                           1      1
183004                           0      1
…                              …    …
365195                           0     12
365196                           0     12
365197                           0     12
365198                           0     12
365199                           0     12

[73000 rows x 12 columns]
```

[35]:
```
Subset1 = Subset1.drop(columns=["date","week_id","state"]) #Dropping the␣
 ↪unwanted columns
Subset1
```

[35]:
```
        product_identifier  department_identifier  outlet  sales  sell_price  \
183000                  74                     11     111      0        3.43
183001                 337                     11     111      3        6.98
183002                 423                     12     111      0        0.97
183003                 432                     12     111      0        4.97
183004                 581                     21     111      1        4.88
…                      …                       …      …    …        …
365195                2932                     33     114      2        2.78
365196                2935                     33     114      4        0.20
365197                3004                     33     114      2        2.50
365198                3008                     33     114      0        1.98
365199                3021                     33     114      0        2.08

        category_of_product_drinks_and_food  \
183000                                    0
183001                                    0
183002                                    0
183003                                    0
183004                                    0
…                                       …
365195                                    1
```

```
365196                                    1
365197                                    1
365198                                    1
365199                                    1

        category_of_product_fast_moving_consumer_goods  \
183000                                                0
183001                                                0
183002                                                0
183003                                                0
183004                                                1
…                                                   …
365195                                                0
365196                                                0
365197                                                0
365198                                                0
365199                                                0

        category_of_product_others  Month
183000                           1      1
183001                           1      1
183002                           1      1
183003                           1      1
183004                           0      1
…                              …    …
365195                           0     12
365196                           0     12
365197                           0     12
365198                           0     12
365199                           0     12

[73000 rows x 9 columns]
```

[36]:
```
X = Subset1.drop(columns= 'sales',axis=1)  #Splitting target & features
Y = Subset1['sales']
```

[37]:
```
print(X)
```

```
        product_identifier  department_identifier  outlet  sell_price  \
183000                  74                     11     111        3.43
183001                 337                     11     111        6.98
183002                 423                     12     111        0.97
183003                 432                     12     111        4.97
183004                 581                     21     111        4.88
…                      …                       …      …          …
365195                2932                     33     114        2.78
365196                2935                     33     114        0.20
```

```
365197                3004                    33     114        2.50
365198                3008                    33     114        1.98
365199                3021                    33     114        2.08

        category_of_product_drinks_and_food  \
183000                                    0
183001                                    0
183002                                    0
183003                                    0
183004                                    0
…                                       …
365195                                    1
365196                                    1
365197                                    1
365198                                    1
365199                                    1

        category_of_product_fast_moving_consumer_goods  \
183000                                               0
183001                                               0
183002                                               0
183003                                               0
183004                                               1
…                                                  …
365195                                               0
365196                                               0
365197                                               0
365198                                               0
365199                                               0

        category_of_product_others  Month
183000                           1      1
183001                           1      1
183002                           1      1
183003                           1      1
183004                           0      1
…                              …    …
365195                           0     12
365196                           0     12
365197                           0     12
365198                           0     12
365199                           0     12

[73000 rows x 8 columns]
```

[38]:
```
print(Y)
```

```
183000    0
```

```
183001    3
183002    0
183003    0
183004    1
          ..
365195    2
365196    4
365197    2
365198    0
365199    0
Name: sales, Length: 73000, dtype: int64
```

[39]:
```
X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size=0.3,␣
↪random_state=42)
```

[40]:
```
print(X.shape, X_train.shape, X_test.shape)
```

```
(73000, 8) (51100, 8) (21900, 8)
```

[41]:
```
reg5 = linear_model.LinearRegression()
ml_model5 = reg5.fit(X_train, Y_train)
```

[42]:
```
print ("R^2 is: \n", ml_model5.score(X_train, Y_train))
```

```
R^2 is:
 0.08579840819308904
```

[43]:
```
reg6 = RandomForestRegressor()
ml_model6 = reg6.fit(X_train, Y_train)
```

[44]:
```
print ("R^2 is: \n", ml_model6.score(X_train, Y_train))
```

```
R^2 is:
 0.5473490362132806
```

[ ]:
```

```