Exploratory Data Analysis

```python
import numpy as np
import pandas as pd

#importing the dataset
df1=pd.read_csv(r'C:\Users\swati\Desktop\python\Product.csv')

df1.head() #Display the first 5 rows of the dataset
```

```
   User_ID Product_ID Gender   Age  Occupation City_Category  \
0  1000001  P00069042      F  0-17          10             A   
1  1000001  P00248942      F  0-17          10             A   
2  1000001  P00087842      F  0-17          10             A   
3  1000001  P00085442      F  0-17          10             A   
4  1000002  P00285442      M   55+          16             C   

  Stay_In_Current_City_Years  Marital_Status  Product_Category_1  \
0                          2               0                   3   
1                          2               0                   1   
2                          2               0                  12   
3                          2               0                  12   
4                         4+               0                   8   

   Product_Category_2  Product_Category_3  Purchase  New  
0                 NaN                 NaN      8370  NaN  
1                 6.0                14.0     15200  NaN  
2                 NaN                 NaN      1422  NaN  
3                14.0                 NaN      1057  NaN  
4                 NaN                 NaN      7969  NaN  
```

```python
df1.tail() #Display the last 5 rows of the dataset
```

```
        User_ID Product_ID Gender    Age  Occupation City_Category  \
550063  1006033  P00372445      M  51-55          13             B   
550064  1006035  P00375436      F  26-35           1             C   
550065  1006036  P00375436      F  26-35          15             B   
550066  1006038  P00375436      F    55+           1             C   
550067  1006039  P00371644      F  46-50           0             B   

       Stay_In_Current_City_Years  Marital_Status  Product_Category_1
\
550063                          1               1                  20

550064                          3               0                  20

550065                         4+               1                  20

550066                          2               0                  20
```

```
550067                         4+               1                  20


        Product_Category_2  Product_Category_3  Purchase  New
550063                 NaN                 NaN       368  NaN
550064                 NaN                 NaN       371  NaN
550065                 NaN                 NaN       137  NaN
550066                 NaN                 NaN       365  NaN
550067                 NaN                 NaN       490  NaN
```

```
df1.info() #Get information about data types, missing values, and
memory usage:
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 550068 entries, 0 to 550067
Data columns (total 13 columns):
 #   Column                      Non-Null Count   Dtype
---  ------                      --------------   -----
 0   User_ID                     550068 non-null  int64
 1   Product_ID                  550068 non-null  object
 2   Gender                      550068 non-null  object
 3   Age                         550068 non-null  object
 4   Occupation                  550068 non-null  int64
 5   City_Category               550068 non-null  object
 6   Stay_In_Current_City_Years  550068 non-null  object
 7   Marital_Status              550068 non-null  int64
 8   Product_Category_1          550068 non-null  int64
 9   Product_Category_2          376430 non-null  float64
 10  Product_Category_3          166821 non-null  float64
 11  Purchase                    550068 non-null  int64
 12  New                         0 non-null       float64
dtypes: float64(3), int64(5), object(5)
memory usage: 54.6+ MB
```

```
df1.describe() #Generate summary statistics for numerical columns:
```

```
            User_ID     Occupation  Marital_Status  Product_Category_1
\
count  5.500680e+05  550068.000000   550068.000000       550068.000000

mean   1.003029e+06       8.076707        0.409653            5.404270

std    1.727592e+03       6.522660        0.491770            3.936211

min    1.000001e+06       0.000000        0.000000            1.000000

25%    1.001516e+06       2.000000        0.000000            1.000000

50%    1.003077e+06       7.000000        0.000000            5.000000
```

```
75%    1.004478e+06      14.000000        1.000000          8.000000

max    1.006040e+06      20.000000        1.000000         20.000000


       Product_Category_2  Product_Category_3       Purchase  New
count       376430.000000       166821.000000  550068.000000  0.0
mean             9.842329           12.668243    9263.968713  NaN
std              5.086590            4.125338    5023.065394  NaN
min              2.000000            3.000000      12.000000  NaN
25%              5.000000            9.000000    5823.000000  NaN
50%              9.000000           14.000000    8047.000000  NaN
75%             15.000000           16.000000   12054.000000  NaN
max             18.000000           18.000000   23961.000000  NaN
```

```python
df1.shape #Get the number of rows and columns in the dataset
```

```
(550068, 13)
```

```python
df1.drop(['New'],axis=1,inplace=True) #drop the coloumn
```

```python
df1.head()
```

```
   User_ID Product_ID Gender   Age  Occupation City_Category   \
0  1000001  P00069042      F  0-17          10             A
1  1000001  P00248942      F  0-17          10             A
2  1000001  P00087842      F  0-17          10             A
3  1000001  P00085442      F  0-17          10             A
4  1000002  P00285442      M   55+          16             C

  Stay_In_Current_City_Years  Marital_Status  Product_Category_1  \
0                          2               0                   3
1                          2               0                   1
2                          2               0                  12
3                          2               0                  12
4                         4+               0                   8

   Product_Category_2  Product_Category_3  Purchase
0                 NaN                 NaN      8370
1                 6.0                14.0     15200
2                 NaN                 NaN      1422
3                14.0                 NaN      1057
4                 NaN                 NaN      7969
```

```python
##Handling categorical feature Gender
df1['Gender']=df1['Gender'].map({'F':0,'M':1}) #used Dictionaries
```

```python
df1.head()
```

```
   User_ID Product_ID  Gender   Age  Occupation City_Category  \
0  1000001  P00069042       0  0-17          10             A
1  1000001  P00248942       0  0-17          10             A
```

```
2  1000001  P00087842       0  0-17          10             A
3  1000001  P00085442       0  0-17          10             A
4  1000002  P00285442       1   55+          16             C

  Stay_In_Current_City_Years  Marital_Status  Product_Category_1  \
0                          2               0                   3   
1                          2               0                   1   
2                          2               0                  12   
3                          2               0                  12   
4                         4+               0                   8   

   Product_Category_2  Product_Category_3  Purchase  
0                 NaN                 NaN      8370  
1                 6.0                14.0     15200  
2                 NaN                 NaN      1422  
3                14.0                 NaN      1057  
4                 NaN                 NaN      7969  
```

```
df1['Age'].unique() # that is used to retrieve the unique values in
the 'Age' column of the DataFrame df1.
```

```
array(['0-17', '55+', '26-35', '46-50', '51-55', '36-45', '18-25'],
      dtype=object)
```