shopping-eda

April 23, 2024

[1]:
```
''' importing the required libraries for analysis '''


import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

[2]:
```
''' Reading the datafile and creating dataframe df_shop '''

df_shop = pd.read_csv('shopping.csv')
df_shop
```

[2]:

	Administrative	Administrative_Duration	Informational	\
0	0	0.0	0	
1	0	0.0	0	
2	0	0.0	0	
3	0	0.0	0	
4	0	0.0	0	
…	…	…	…	
12325	3	145.0	0	
12326	0	0.0	0	
12327	0	0.0	0	
12328	4	75.0	0	
12329	0	0.0	0	

	Informational_Duration	ProductRelated	ProductRelated_Duration	\
0	0.0	1	0.000000	
1	0.0	2	64.000000	
2	0.0	1	0.000000	
3	0.0	2	2.666667	
4	0.0	10	627.500000	
…	…	…	…	
12325	0.0	53	1783.791667	
12326	0.0	5	465.750000	
12327	0.0	6	184.250000	
12328	0.0	15	346.000000	

```
12329                    0.0               3                21.250000

       BounceRates  ExitRates  PageValues  SpecialDay Month  OperatingSystems  \
0         0.200000   0.200000    0.000000         0.0   Feb                 1
1         0.000000   0.100000    0.000000         0.0   Feb                 2
2         0.200000   0.200000    0.000000         0.0   Feb                 4
3         0.050000   0.140000    0.000000         0.0   Feb                 3
4         0.020000   0.050000    0.000000         0.0   Feb                 3
…              …          …           …         …    …                   …
12325     0.007143   0.029031   12.241717         0.0   Dec                 4
12326     0.000000   0.021333    0.000000         0.0   Nov                 3
12327     0.083333   0.086667    0.000000         0.0   Nov                 3
12328     0.000000   0.021053    0.000000         0.0   Nov                 2
12329     0.000000   0.066667    0.000000         0.0   Nov                 3

       Browser  Region  TrafficType        VisitorType  Weekend  Revenue
0            1       1            1  Returning_Visitor    False    False
1            2       1            2  Returning_Visitor    False    False
2            1       9            3  Returning_Visitor    False    False
3            2       2            4  Returning_Visitor    False    False
4            3       1            4  Returning_Visitor     True    False
…          …       …            …                  …      …      …
12325        6       1            1  Returning_Visitor     True    False
12326        2       1            8  Returning_Visitor     True    False
12327        2       1           13  Returning_Visitor     True    False
12328        2       3           11  Returning_Visitor    False    False
12329        2       1            2        New_Visitor     True    False

[12330 rows x 18 columns]
```

[3]:
```
''' Checking for null values in df '''

df_shop.isnull().sum()
```

[3]:
```
Administrative             0
Administrative_Duration    0
Informational              0
Informational_Duration     0
ProductRelated             0
ProductRelated_Duration    0
BounceRates                0
ExitRates                  0
PageValues                 0
SpecialDay                 0
Month                      0
OperatingSystems           0
Browser                    0
```

```
Region                   0
TrafficType              0
VisitorType              0
Weekend                  0
Revenue                  0
dtype: int64
```

[4]:
```
''' getting the numeric information about df '''

df_shop.describe()
```

[4]:

	Administrative	Administrative_Duration	Informational	\
count	12330.000000	12330.000000	12330.000000	
mean	2.315166	80.818611	0.503569	
std	3.321784	176.779107	1.270156	
min	0.000000	0.000000	0.000000	
25%	0.000000	0.000000	0.000000	
50%	1.000000	7.500000	0.000000	
75%	4.000000	93.256250	0.000000	
max	27.000000	3398.750000	24.000000	

	Informational_Duration	ProductRelated	ProductRelated_Duration	\
count	12330.000000	12330.000000	12330.000000	
mean	34.472398	31.731468	1194.746220	
std	140.749294	44.475503	1913.669288	
min	0.000000	0.000000	0.000000	
25%	0.000000	7.000000	184.137500	
50%	0.000000	18.000000	598.936905	
75%	0.000000	38.000000	1464.157214	
max	2549.375000	705.000000	63973.522230	

	BounceRates	ExitRates	PageValues	SpecialDay	\
count	12330.000000	12330.000000	12330.000000	12330.000000	
mean	0.022191	0.043073	5.889258	0.061427	
std	0.048488	0.048597	18.568437	0.198917	
min	0.000000	0.000000	0.000000	0.000000	
25%	0.000000	0.014286	0.000000	0.000000	
50%	0.003112	0.025156	0.000000	0.000000	
75%	0.016813	0.050000	0.000000	0.000000	
max	0.200000	0.200000	361.763742	1.000000	

	OperatingSystems	Browser	Region	TrafficType
count	12330.000000	12330.000000	12330.000000	12330.000000
mean	2.124006	2.357097	3.147364	4.069586
std	0.911325	1.717277	2.401591	4.025169
min	1.000000	1.000000	1.000000	1.000000
25%	2.000000	2.000000	1.000000	2.000000

```
50%            2.000000      2.000000      3.000000      2.000000
75%            3.000000      2.000000      4.000000      4.000000
max            8.000000     13.000000      9.000000     20.000000
```

[5]:
```
''' checking number of values for column 'Administrative' '''

df_shop['Administrative'].value_counts()
```

[5]:
```
Administrative
0     5768
1     1354
2     1114
3      915
4      765
5      575
6      432
7      338
8      287
9      225
10     153
11     105
12      86
13      56
14      44
15      38
16      24
17      16
18      12
19       6
24       4
22       4
23       3
21       2
20       2
27       1
26       1
Name: count, dtype: int64
```

[6]:
```
''' checking number of values for column 'Productrelated' '''

df_shop['ProductRelated'].value_counts()
```

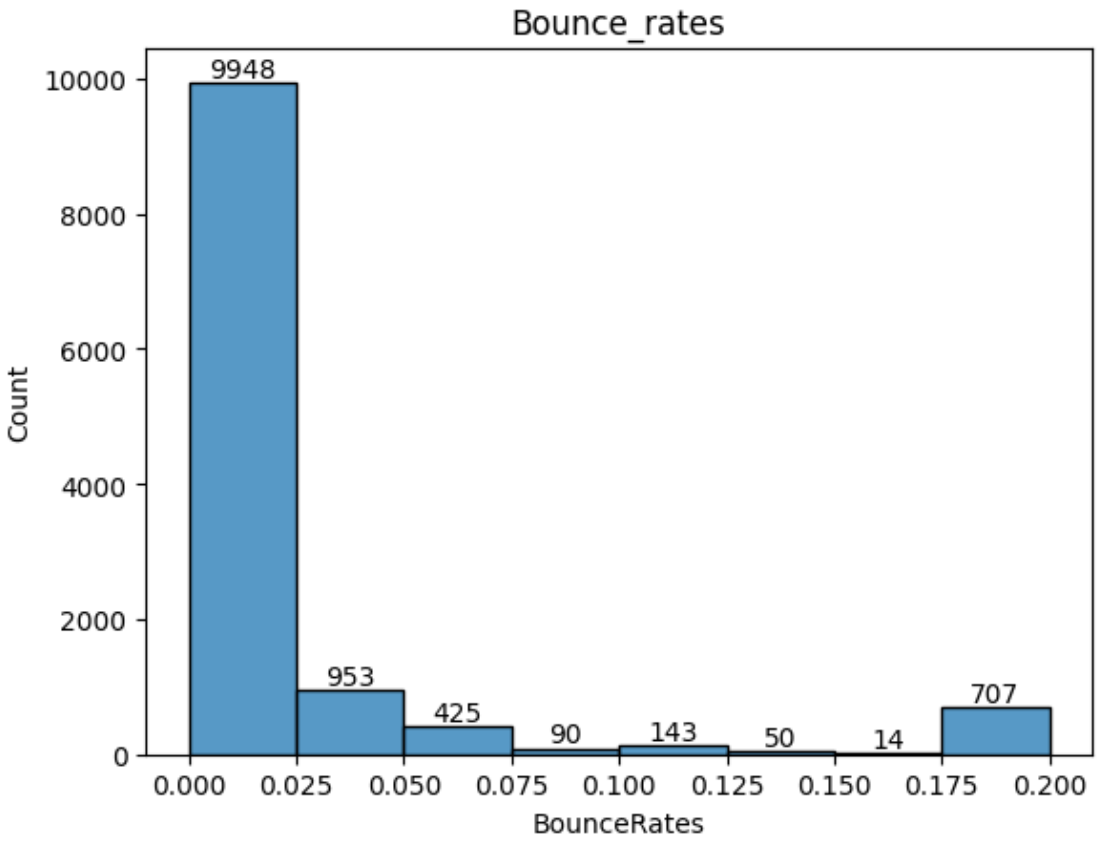
[6]:
```
ProductRelated
1      622
2      465
3      458
4      404
```

```
6      396
       …
243      1
409      1
262      1
414      1
192      1
Name: count, Length: 311, dtype: int64
```

[7]:
```
''' checking number of values for column 'Administrative_duration' '''

df_shop['Administrative_Duration'].value_counts()
```

[7]:
```
Administrative_Duration
0.000000      5903
4.000000        56
5.000000        53
7.000000        45
11.000000       42
                …
68.014286        1
362.300000       1
90.700000        1
760.900000       1
150.357143       1
Name: count, Length: 3335, dtype: int64
```

[8]:
```
''' Plotting the bounce rates percentage for given data '''

plt.title('Bounce_rates')
y = sns.histplot(df_shop['BounceRates'],bins = 8)
y.bar_label(y.containers[0])
plt.show()
```



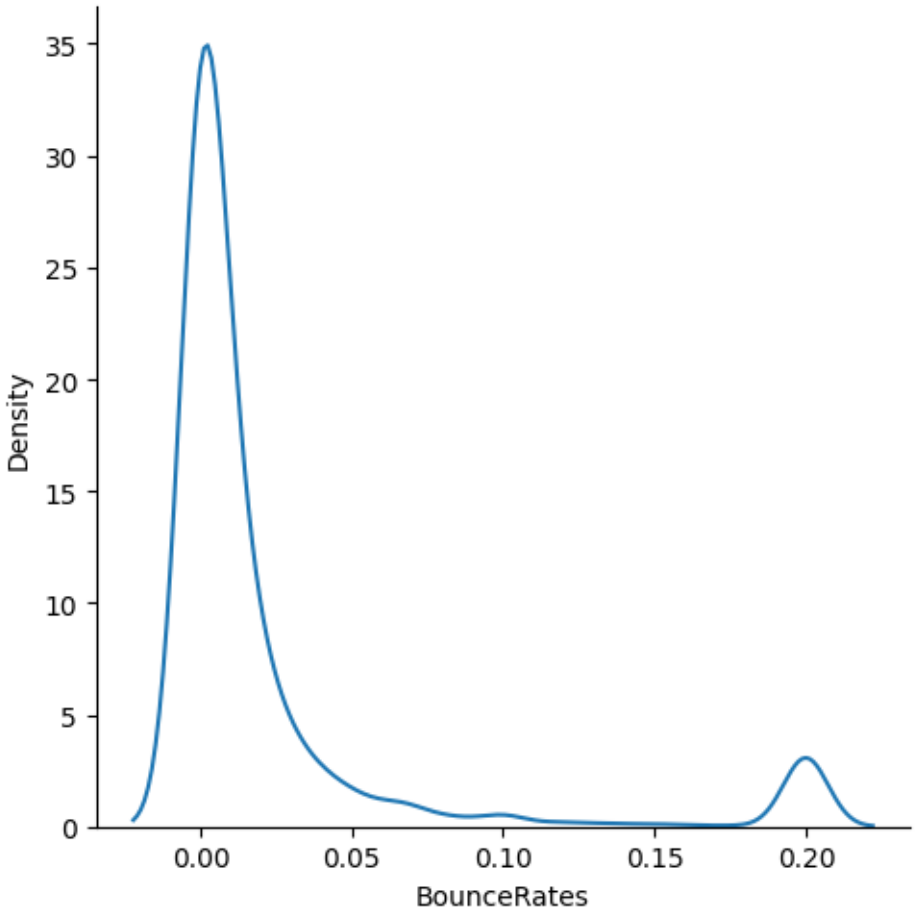
The above histogram shows us the percentage count of bounce rates which implies that the bounce rate pecentage is very high at point '0' , which is a good sign as average bouncerate of a company in ecommerce is below 20%.

we can see small hike in bounce rate for 0.25% and gradually decreasing and raise at end at 20%.

traking the necesary steps and improving customer experience and decraese load time of webpage increases customer association with products ansd decreases bounce rates.

[9]:
```
''' plotting the distribution of bounce rates '''

plt.figure(figsize = (10,6))
sns.displot(df_shop['BounceRates'],kind = 'kde')

plt.show()
```

```
<Figure size 1000x600 with 0 Axes>
```



clearly we can see that data is right skewed or positvely skewed with depth in right side of the data.

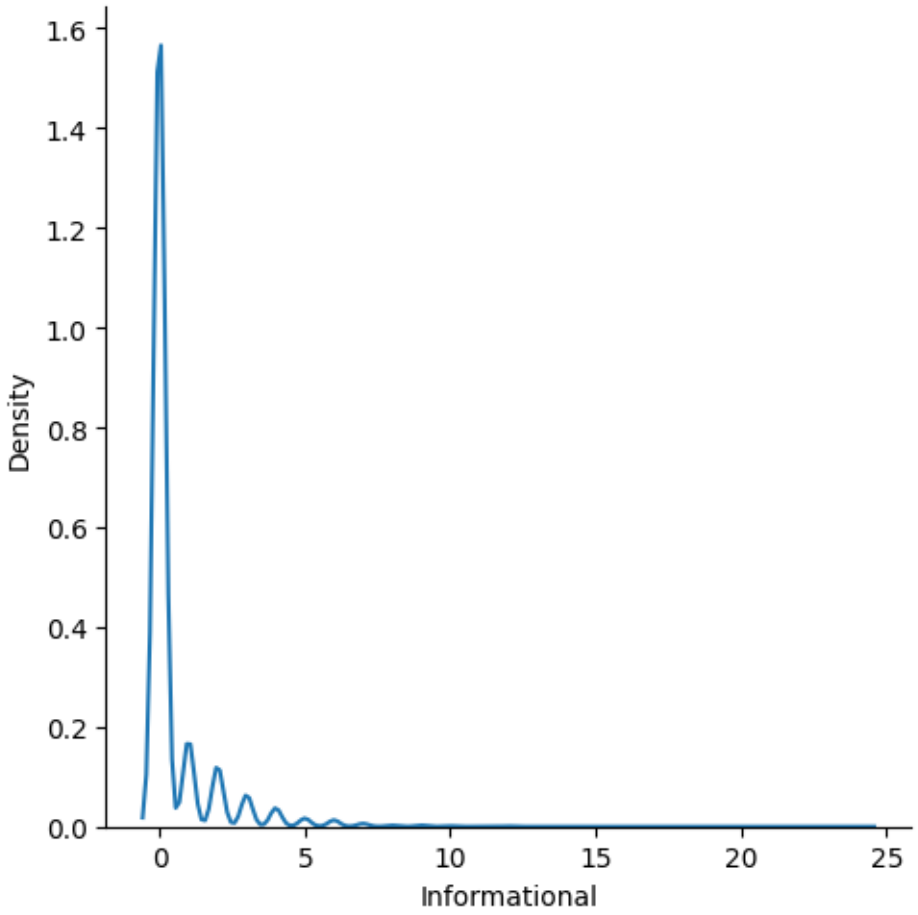
so we calculate median of the data as mean of the data will be larger and not be a average value , while median will the middle value for this data.

[10]:
```
''' calculatrig the median for boub=ncerates column '''

bounce_rate  = df_shop['BounceRates']
bounce_rate.median()
```

[10]:
```
0.0031124675
```

A bounce rate between 20% and 45% is generally considered to be a good benchmark range for ecommerce.(source:Google). So, here the bounce rate is 0.311 ~ 0.3% which is good percentage.

[11]:
```
''' plotting the distribution of informtional catgory'''

sns.displot(df_shop['Informational'],kind = 'kde')
plt.show()
```



here , the data is positively skewed, so we can calculate the median as for posituvely skewed data the mean is greater than usual it won't gives the middle value.

So we use median for the middle value whuch indicates 50% of data lies below median and 50% above means indicates average.

[12]:
```
df_shop['Informational'].median()
```

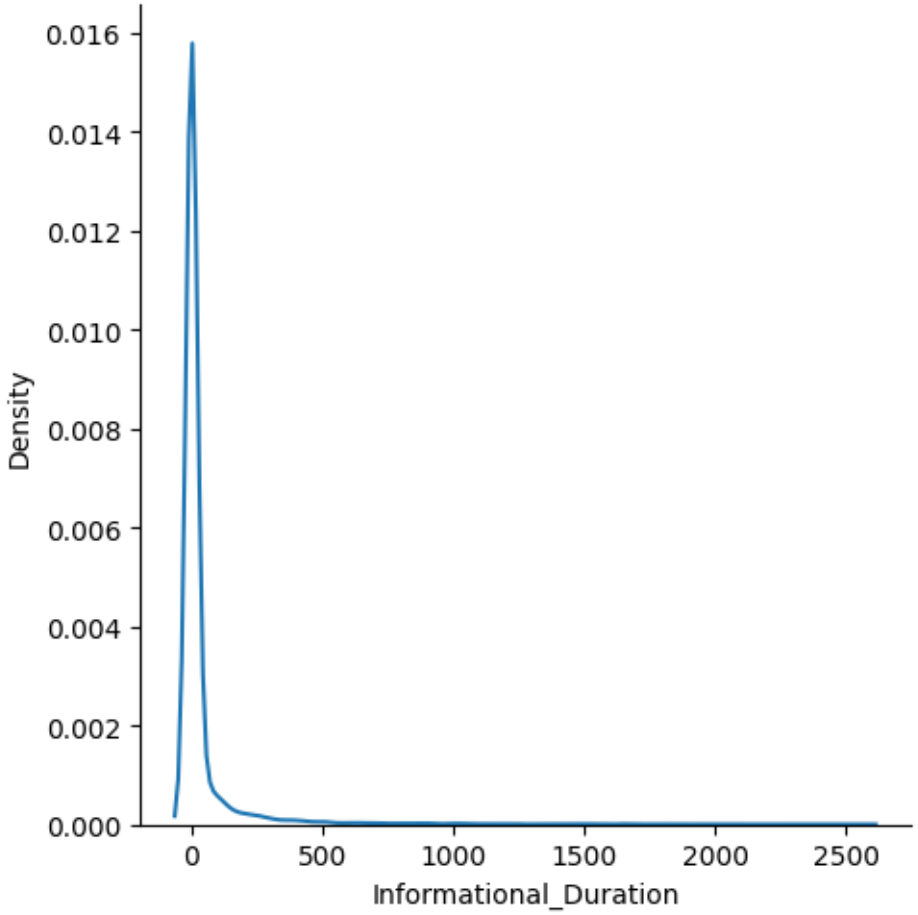
[12]:
```
0.0
```

1. here the median is '0' whch indicates the user visiting informational page is usually very low.
2. Website should improve with respect to informational pages UI,content and other factors that leads to decrease in cutomer attraction.

.

Here we are plotting for the distribution of Informational_duartion which indicates the time spent by visitors

[13]:
```
''' plotting the distribution of Informational duration '''

y = sns.displot(df_shop['Informational_Duration'],kind = 'kde')

plt.show()
```



.

[14]:
```
''' fetching the number of informational category duration time '''

df_shop['Informational_Duration'].value_counts()
```

[14]:
```
Informational_Duration
0.00      9925
9.00        33
7.00        26
10.00       26
6.00        26
```

```
          …
246.80       1
274.00       1
13.40        1
223.15       1
211.25       1
Name: count, Length: 1258, dtype: int64
```

[15]:
```
''' finding the medain for informational category pages time spent '''

df_shop['Informational_Duration'].median()
```

[15]:
```
0.0
```

1. Here from the above plot we can see that most of the visitors spend '0' time on informational pages.
2. It is very low for as the median is also at '0'.hence care msut me taken for informational pages betterment which increases the visitors spent time for a ecom website.
3. Informtional pages are important for customer as it helps in the detail view of product and company norms.
4. pages should be information rich and has to be more user friendly by easing UI experinece, which can help customers to spent more time for informational pages.

[16]:
```
df_shop
```

[16]:
```
       Administrative  Administrative_Duration  Informational  \
0                   0                      0.0              0
1                   0                      0.0              0
2                   0                      0.0              0
3                   0                      0.0              0
4                   0                      0.0              0
…                 …                        …              …
12325               3                    145.0              0
12326               0                      0.0              0
12327               0                      0.0              0
12328               4                     75.0              0
12329               0                      0.0              0

       Informational_Duration  ProductRelated  ProductRelated_Duration  \
0                         0.0               1                 0.000000
1                         0.0               2                64.000000
2                         0.0               1                 0.000000
3                         0.0               2                 2.666667
4                         0.0              10               627.500000
…                       …                  …                       …
12325                     0.0              53              1783.791667
12326                     0.0               5               465.750000
```

```
12327                    0.0               6              184.250000
12328                    0.0              15              346.000000
12329                    0.0               3               21.250000

       BounceRates  ExitRates  PageValues  SpecialDay Month  OperatingSystems  \
0         0.200000   0.200000    0.000000         0.0   Feb                 1
1         0.000000   0.100000    0.000000         0.0   Feb                 2
2         0.200000   0.200000    0.000000         0.0   Feb                 4
3         0.050000   0.140000    0.000000         0.0   Feb                 3
4         0.020000   0.050000    0.000000         0.0   Feb                 3
…              …          …           …           …    …                  …
12325     0.007143   0.029031   12.241717         0.0   Dec                 4
12326     0.000000   0.021333    0.000000         0.0   Nov                 3
12327     0.083333   0.086667    0.000000         0.0   Nov                 3
12328     0.000000   0.021053    0.000000         0.0   Nov                 2
12329     0.000000   0.066667    0.000000         0.0   Nov                 3

       Browser  Region  TrafficType        VisitorType  Weekend  Revenue
0            1       1            1  Returning_Visitor    False    False
1            2       1            2  Returning_Visitor    False    False
2            1       9            3  Returning_Visitor    False    False
3            2       2            4  Returning_Visitor    False    False
4            3       1            4  Returning_Visitor     True    False
…          …       …            …                  …       …       …
12325        6       1            1  Returning_Visitor     True    False
12326        2       1            8  Returning_Visitor     True    False
12327        2       1           13  Returning_Visitor     True    False
12328        2       3           11  Returning_Visitor    False    False
12329        2       1            2        New_Visitor     True    False

[12330 rows x 18 columns]
```

[17]:
```
''' finding the corelation for numerics in data '''

corelation =␣
 ↪df_shop[['Administrative','Administrative_Duration','Informational',
        ␣
 ↪'Informational_Duration','ProductRelated','ProductRelated_Duration','BounceRates','ExitRate
         ,'OperatingSystems','Browser','Region','TrafficType',]].corr('pearson')
corelation
```

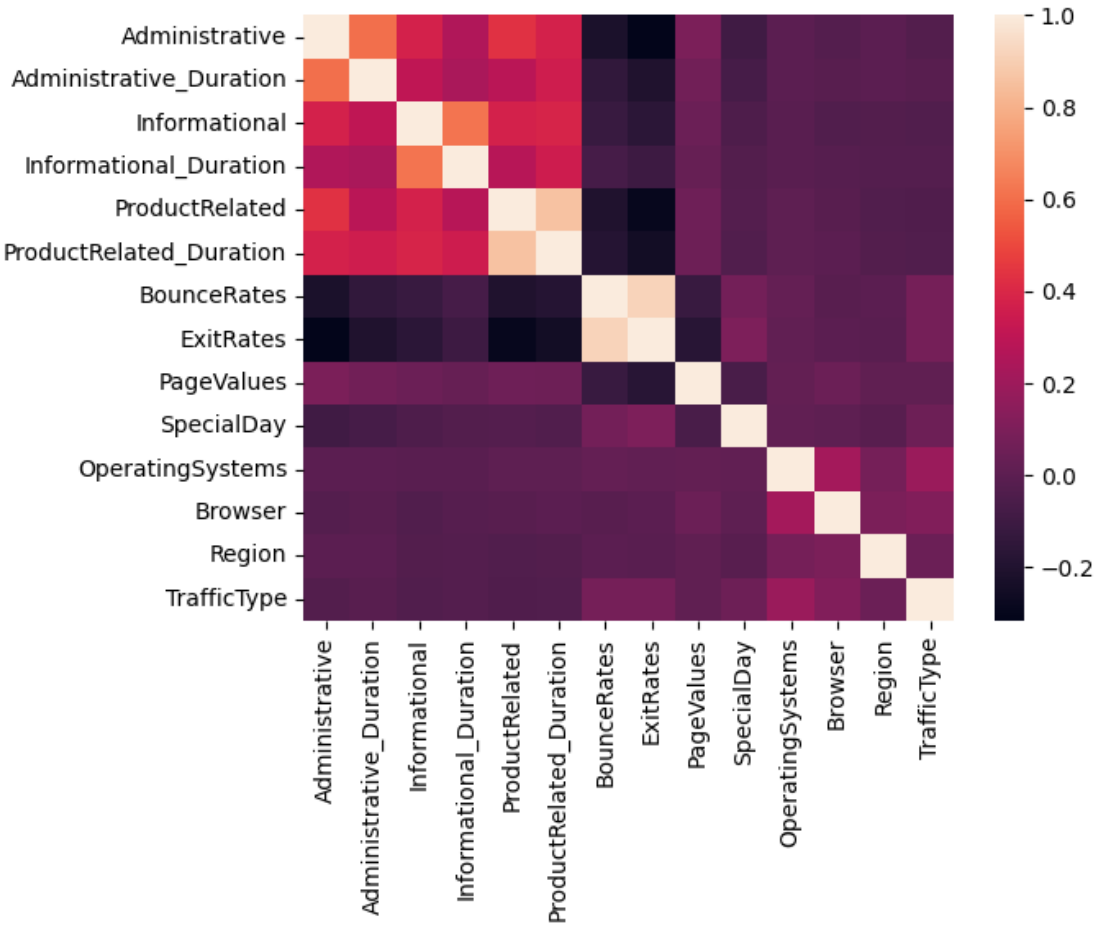
[17]:
```
                         Administrative  Administrative_Duration  \
Administrative                 1.000000                 0.601583
Administrative_Duration        0.601583                 1.000000
Informational                  0.376850                 0.302710
Informational_Duration         0.255848                 0.238031
ProductRelated                 0.431119                 0.289087
```

```
ProductRelated_Duration        0.373939                0.355422
BounceRates                   -0.223563               -0.144170
ExitRates                     -0.316483               -0.205798
PageValues                     0.098990                0.067608
SpecialDay                    -0.094778               -0.073304
OperatingSystems              -0.006347               -0.007343
Browser                       -0.025035               -0.015392
Region                        -0.005487               -0.005561
TrafficType                   -0.033561               -0.014376

                         Informational  Informational_Duration  \
Administrative                0.376850                0.255848
Administrative_Duration       0.302710                0.238031
Informational                 1.000000                0.618955
Informational_Duration        0.618955                1.000000
ProductRelated                0.374164                0.280046
ProductRelated_Duration       0.387505                0.347364
BounceRates                  -0.116114               -0.074067
ExitRates                    -0.163666               -0.105276
PageValues                    0.048632                0.030861
SpecialDay                   -0.048219               -0.030577
OperatingSystems             -0.009527               -0.009579
Browser                      -0.038235               -0.019285
Region                       -0.029169               -0.027144
TrafficType                  -0.034491               -0.024675

                         ProductRelated  ProductRelated_Duration  BounceRates  \
Administrative                 0.431119                 0.373939    -0.223563
Administrative_Duration        0.289087                 0.355422    -0.144170
Informational                  0.374164                 0.387505    -0.116114
Informational_Duration         0.280046                 0.347364    -0.074067
ProductRelated                 1.000000                 0.860927    -0.204578
ProductRelated_Duration        0.860927                 1.000000    -0.184541
BounceRates                   -0.204578                -0.184541     1.000000
ExitRates                     -0.292526                -0.251984     0.913004
PageValues                     0.056282                 0.052823    -0.119386
SpecialDay                    -0.023958                -0.036380     0.072702
OperatingSystems               0.004290                 0.002976     0.023823
Browser                       -0.013146                -0.007380    -0.015772
Region                        -0.038122                -0.033091    -0.006485
TrafficType                   -0.043064                -0.036377     0.078286

                         ExitRates  PageValues  SpecialDay  OperatingSystems  \
Administrative           -0.316483    0.098990   -0.094778         -0.006347
Administrative_Duration  -0.205798    0.067608   -0.073304         -0.007343
Informational            -0.163666    0.048632   -0.048219         -0.009527
Informational_Duration   -0.105276    0.030861   -0.030577         -0.009579
```

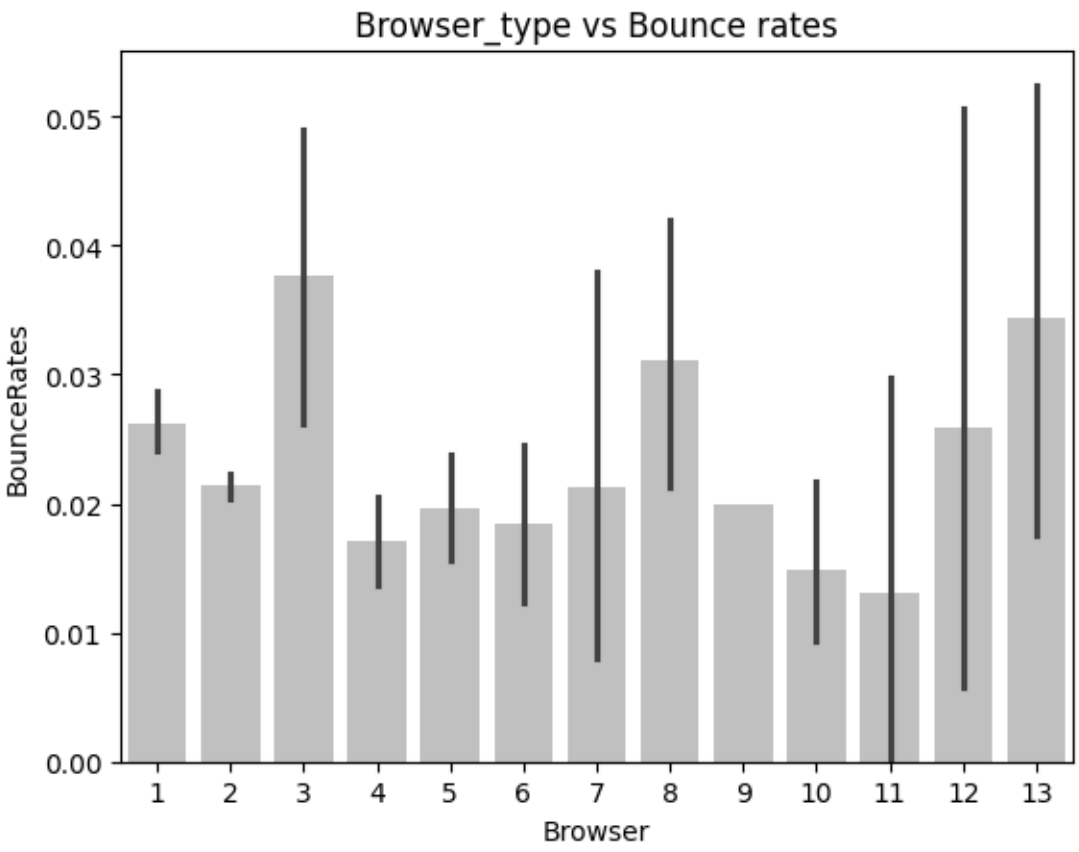
```
ProductRelated           -0.292526    0.056282   -0.023958          0.004290
ProductRelated_Duration  -0.251984    0.052823   -0.036380          0.002976
BounceRates               0.913004   -0.119386    0.072702          0.023823
ExitRates                 1.000000   -0.174498    0.102242          0.014567
PageValues               -0.174498    1.000000   -0.063541          0.018508
SpecialDay                0.102242   -0.063541    1.000000          0.012652
OperatingSystems          0.014567    0.018508    0.012652          1.000000
Browser                  -0.004442    0.045592    0.003499          0.223013
Region                   -0.008907    0.011315   -0.016098          0.076775
TrafficType               0.078616    0.012532    0.052301          0.189154

                           Browser    Region  TrafficType
Administrative           -0.025035 -0.005487    -0.033561
Administrative_Duration  -0.015392 -0.005561    -0.014376
Informational            -0.038235 -0.029169    -0.034491
Informational_Duration   -0.019285 -0.027144    -0.024675
ProductRelated           -0.013146 -0.038122    -0.043064
ProductRelated_Duration  -0.007380 -0.033091    -0.036377
BounceRates              -0.015772 -0.006485     0.078286
ExitRates                -0.004442 -0.008907     0.078616
PageValues                0.045592  0.011315     0.012532
SpecialDay                0.003499 -0.016098     0.052301
OperatingSystems          0.223013  0.076775     0.189154
Browser                   1.000000  0.097393     0.111938
Region                    0.097393  1.000000     0.047520
TrafficType               0.111938  0.047520     1.000000
```

[18]:
```
''' plotting a heatmap for corelation of numerical data in df '''

sns.heatmap(corelation,cbar = True)
```

[18]:
```
<Axes: >
```



[19]:
```
''' Plotting bar graphs for browser and percentage of bounce rates '''
plt.title('Browser_type vs Bounce rates')
y = sns.barplot(data = df_shop,x = "Browser",y = 'BounceRates',color = 'silver')

plt.show()
```



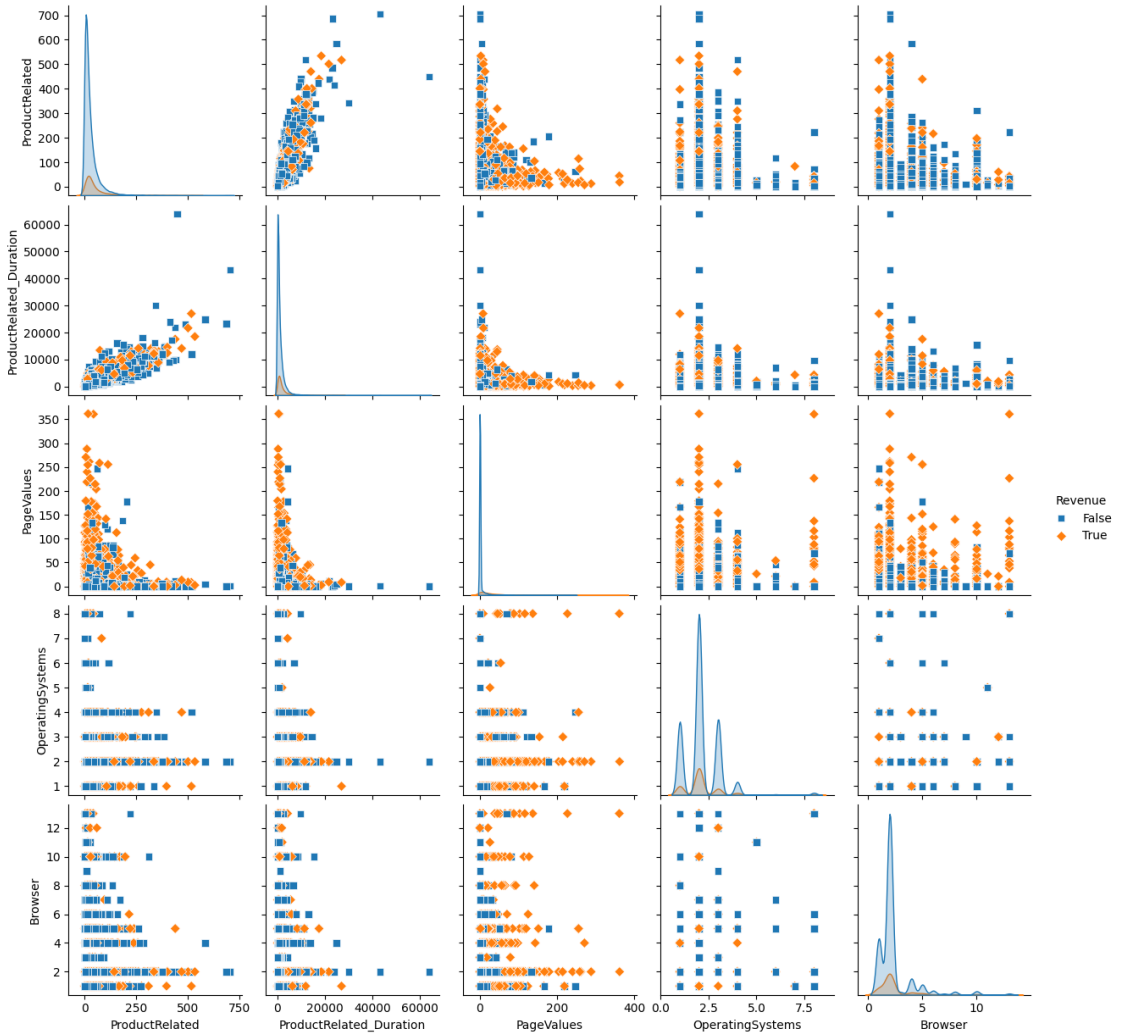
[20]:
```
''' fetching column names from dataframe '''

df_shop.columns
```

[20]:
```
Index(['Administrative', 'Administrative_Duration', 'Informational',
       'Informational_Duration', 'ProductRelated', 'ProductRelated_Duration',
       'BounceRates', 'ExitRates', 'PageValues', 'SpecialDay', 'Month',
       'OperatingSystems', 'Browser', 'Region', 'TrafficType', 'VisitorType',
       'Weekend', 'Revenue'],
      dtype='object')
```

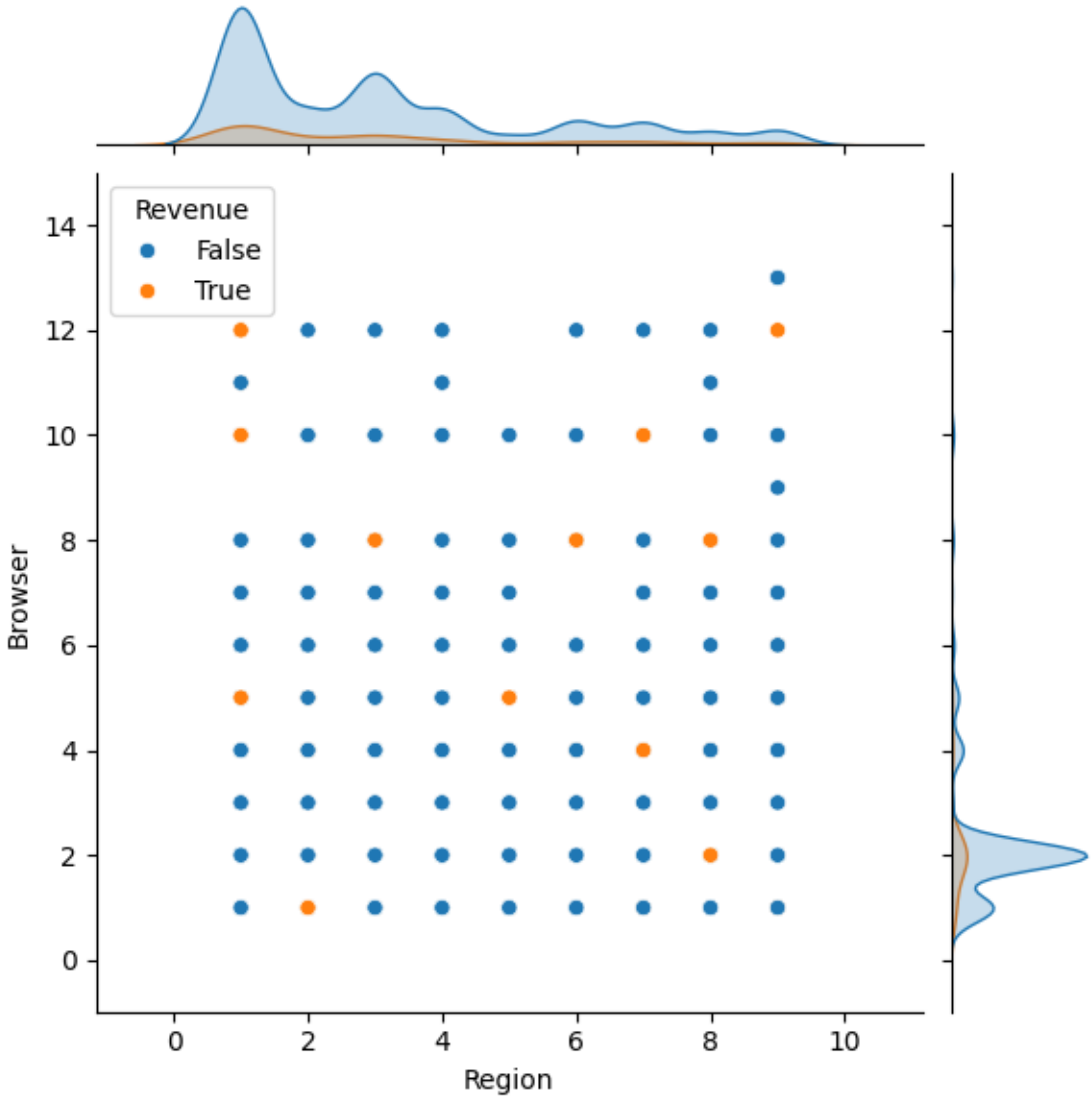
[21]:
```
''' creating anew datframe from the columns '''

a =␣
 ↪df_shop[['ProductRelated','ProductRelated_Duration','PageValues','OperatingSystems',␣
 ↪'Browser','Revenue']]
```

[22]:
```
''' plotting pair plot for given columns in df against rvenue genrated '''

sns.pairplot(a,hue = 'Revenue',markers=["s", "D"])
```

```
plt.show()
```



[23]:
```
''' plotting a joint plot for region wise browser an revenue generated '''

sns.jointplot(data = df_shop,x = 'Region',y = 'Browser',hue = 'Revenue')
plt.show()
```



In the above joint plot we can see that region wise browser conversion rates i.e, for every region and browser type used by customer we are plotting the conversion rates(revenue).

Regions (1,7,8)has good conversion rate than other regionscompared.

In region 1 - (5,10.12) are broser types having positive conversion rates.

In region 7 - (4,10) are browser types having positive conversion rates.

In region 8 - (2,8) are browser types having positive conc=version rates.

Rest all region has not a good conversion rates. mostof them has single positive conversion rats w..r.t browser type

[24]:
```
''' making copy of df '''

df_shop1 = df_shop.copy()
```

[25]:
```
''' Encoding the 1 and 0 as True and false for revenue column '''

df_shop1['dist_rev'] = df_shop['Revenue'].apply(lambda x: 1 if x == True else 0␣
 ↪)
df_shop1
```

[25]:

	Administrative	Administrative_Duration	Informational	\
0	0	0.0	0	
1	0	0.0	0	
2	0	0.0	0	
3	0	0.0	0	
4	0	0.0	0	
…	…	…	…	
12325	3	145.0	0	
12326	0	0.0	0	
12327	0	0.0	0	
12328	4	75.0	0	
12329	0	0.0	0	

	Informational_Duration	ProductRelated	ProductRelated_Duration	\
0	0.0	1	0.000000	
1	0.0	2	64.000000	
2	0.0	1	0.000000	
3	0.0	2	2.666667	
4	0.0	10	627.500000	
…	…	…	…	
12325	0.0	53	1783.791667	
12326	0.0	5	465.750000	
12327	0.0	6	184.250000	
12328	0.0	15	346.000000	
12329	0.0	3	21.250000	

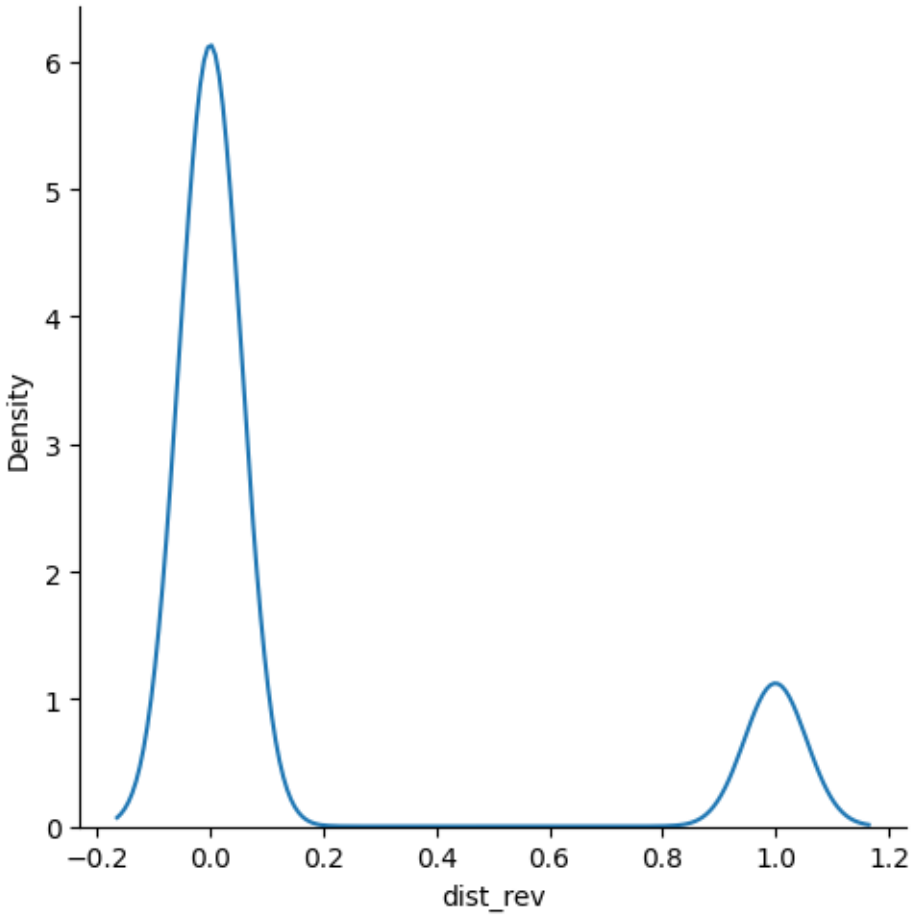
	BounceRates	ExitRates	PageValues	SpecialDay	Month	OperatingSystems	\
0	0.200000	0.200000	0.000000	0.0	Feb	1	
1	0.000000	0.100000	0.000000	0.0	Feb	2	
2	0.200000	0.200000	0.000000	0.0	Feb	4	
3	0.050000	0.140000	0.000000	0.0	Feb	3	
4	0.020000	0.050000	0.000000	0.0	Feb	3	
…	…	…	…	…	…	…	
12325	0.007143	0.029031	12.241717	0.0	Dec	4	
12326	0.000000	0.021333	0.000000	0.0	Nov	3	
12327	0.083333	0.086667	0.000000	0.0	Nov	3	
12328	0.000000	0.021053	0.000000	0.0	Nov	2	

```
12329     0.000000   0.066667    0.000000         0.0   Nov                3

       Browser  Region  TrafficType        VisitorType  Weekend  Revenue  \
0            1       1            1  Returning_Visitor    False    False
1            2       1            2  Returning_Visitor    False    False
2            1       9            3  Returning_Visitor    False    False
3            2       2            4  Returning_Visitor    False    False
4            3       1            4  Returning_Visitor     True    False
…          …     …          …                  …      …      …
12325        6       1            1  Returning_Visitor     True    False
12326        2       1            8  Returning_Visitor     True    False
12327        2       1           13  Returning_Visitor     True    False
12328        2       3           11  Returning_Visitor    False    False
12329        2       1            2        New_Visitor     True    False

       dist_rev
0             0
1             0
2             0
3             0
4             0
…           …
12325         0
12326         0
12327         0
12328         0
12329         0

[12330 rows x 19 columns]
```

.

[26]:
```
''' plotting the distribution of revenue '''


sns.displot(df_shop1['dist_rev'],kind = 'kde')
```

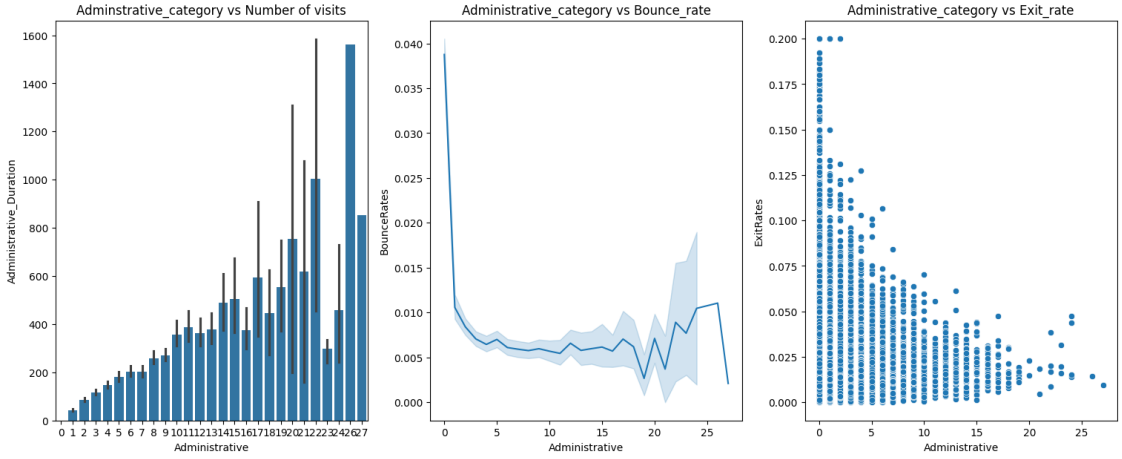
[26]:
```
<seaborn.axisgrid.FacetGrid at 0x77fca39009a0>
```



1. Here we are plotting the distribution of the target varible 'Revenue'.
2. clearly we can see that plot is mostly 'Positively skewed' or 'Right skewed' 3. which indicates that data the revenue or purchases are in nehgative aspect that most of the visitors are not converting.
3. The convertion rate is low.

[27]:
```
''' plotting the Adminstrative category column against bouncerate,exitrate and␣
 ↪visits '''

plt.figure(figsize = (25,7))
plt.subplot(1,4,1)
plt.title('Adminstrative_category vs Number of visits')
sns.barplot(data = df_shop,x = 'Administrative',y= 'Administrative_Duration')
plt.subplot(1,4,2)
plt.title('Administrative_category vs Bounce_rate')
sns.lineplot(data = df_shop,x = 'Administrative', y = 'BounceRates')
plt.subplot(1,4,3)
plt.title('Administrative_category vs Exit_rate')
```

```
sns.scatterplot(data = df_shop,x = 'Administrative',y = 'ExitRates')
plt.show()
```



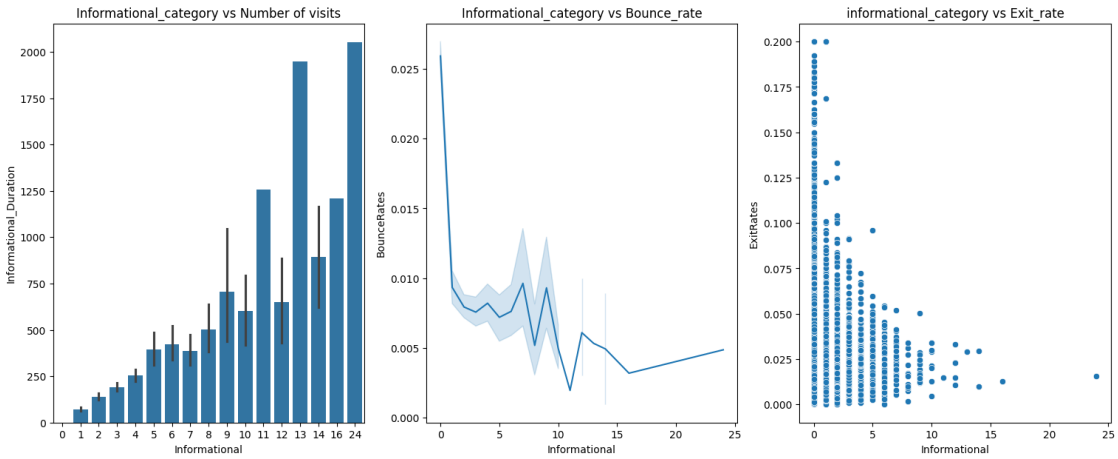
here we plotted the Adminsitrative categories w.r.t Number of visits,bounce rate and Exit rates. 1. Firstly we plotted administrative pages count with duration,where number of visitor spend more time with pages which are visited more .i.e more time a person visits a page, he/she spends more time on that page.

2. In the second line plot we showed the bounce rate w.r.t to administartive categories, where we can see that data is right skewed.i.e more percentage of bounce rate is concentarated at or below origin which is a not good metric in case of bounce rate , which resembles the visitors exit the page immediately. Without performing action. The bounce rate is more at landing page(1-5) categories must be develop in such a way that visitors spend a good amount of time and make some purchase or search for a product.

3. In the third scatter plot we showed the Exit rates w.r.t Administrative categories count, here the exit pages are more concentarted from 1-5 which is not a good sigh beacuse visitors are not drilling down much into website which causes the exit rate more. As the count of Adminsitrative pages visits increases the exit rate decreases, which indicates most of pages with lesser interaction like from(1-10) are higher in case of exit rates as they are not quite customer satsifaction. Care must be taken to these pages of lesser count for better performance and drill down details of product purchase providing to customers or visitors.

[28]:
```
''' plotting the informational category column against bouncerate,exitrate and␣
 ↪visits '''

plt.figure(figsize = (25,7))
plt.subplot(1,4,1)
plt.title('Informational_category vs Number of visits')
sns.barplot(data = df_shop,x = 'Informational',y= 'Informational_Duration')
plt.subplot(1,4,2)
plt.title('Informational_category vs Bounce_rate')
```

```
sns.lineplot(data = df_shop,x = 'Informational', y = 'BounceRates')
plt.subplot(1,4,3)
plt.title('informational_category vs Exit_rate')
sns.scatterplot(data = df_shop,x = 'Informational',y = 'ExitRates')
plt.show()
```



here we plotted the Informational categories w.r.t Number of visits,bounce rate and Exit rates. 1. Firstly we plotted informational pages count with duration,where number of visitor spend more time with pages which are visited more .i.e more time a person visits a page, he/she spends more time on that page.

2. In the second line plot we showed the bounce rate w.r.t to informational category count, where we can see that data is right skewed.i.e more percentage of bounce rate is concentarated at or below origin which is a not good metric in case of bounce rate , which resembles the visitors exit the page immediately. Without performing action for pages with less vistors. The bounce rate is more at landing page(1-5) categories must be develop in such a way that visitors spend a good amount of time and make some purchase or search for a product.Although the more number of visitors ages are good performers incase of Bounce rate which is less compared to pages which are less visited.

3. In the third scatter plot we showed the Exit rates w.r.t Informational categories count, here the exit pages are more concentarted from 1-5 which is not a good sigh beacuse visitors are not drilling down much into website which causes the exit rate more. As the count of Adminsitrative pages visits increases the exit rate decreases, which indicates most of pages with lesser interaction like from(1-10) are higher in case of exit rates as they are not quite customer satsifaction. Care must be taken to these pages of lesser count for better performance and drill down details of product purchase providing to customers or visitors. As we go on along x-axis the number of page visits increases and exit rate deceases. The pages with less visits must be taken care.

[29]:
```
''' creating bins for product category pages '''

bin = [0,100,200,300,400,500,600,750]
```

```
label = ['0-100','101-200','201-300','301-400','401-500','501-600','600+']

df_shop1['product_range'] = pd.cut(df_shop['ProductRelated'],bins = bin,labels
 ↪= label)
df_shop1.head()
```

[29]:

	Administrative	Administrative_Duration	Informational	\
0	0	0.0	0	
1	0	0.0	0	
2	0	0.0	0	
3	0	0.0	0	
4	0	0.0	0	

	Informational_Duration	ProductRelated	ProductRelated_Duration	\
0	0.0	1	0.000000	
1	0.0	2	64.000000	
2	0.0	1	0.000000	
3	0.0	2	2.666667	
4	0.0	10	627.500000	

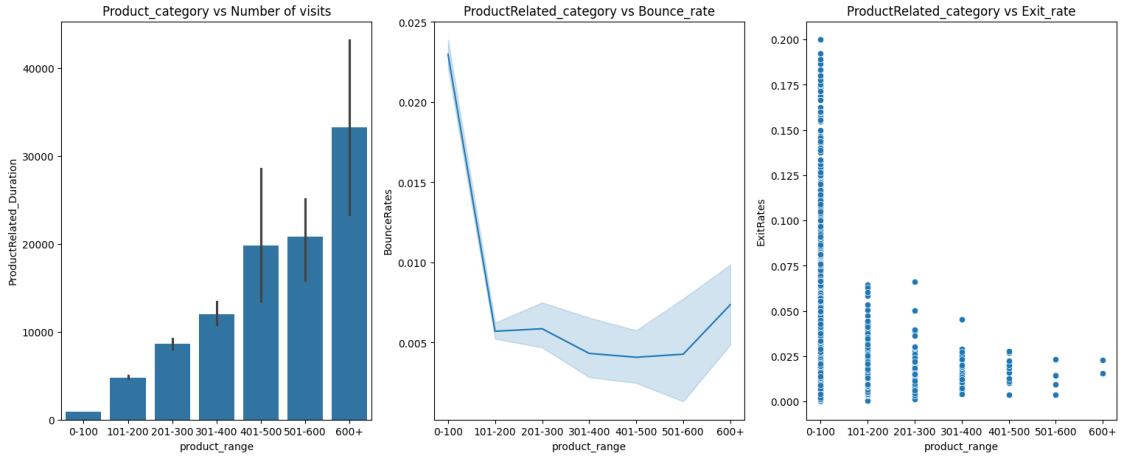
	BounceRates	ExitRates	PageValues	SpecialDay	Month	OperatingSystems	\
0	0.20	0.20	0.0	0.0	Feb	1	
1	0.00	0.10	0.0	0.0	Feb	2	
2	0.20	0.20	0.0	0.0	Feb	4	
3	0.05	0.14	0.0	0.0	Feb	3	
4	0.02	0.05	0.0	0.0	Feb	3	

	Browser	Region	TrafficType	VisitorType	Weekend	Revenue	\
0	1	1	1	Returning_Visitor	False	False	
1	2	1	2	Returning_Visitor	False	False	
2	1	9	3	Returning_Visitor	False	False	
3	2	2	4	Returning_Visitor	False	False	
4	3	1	4	Returning_Visitor	True	False	

	dist_rev	product_range
0	0	0-100
1	0	0-100
2	0	0-100
3	0	0-100
4	0	0-100

[30]:
```
''' plotting the product range column againstbouncerate,exitrate and visits '''

plt.figure(figsize = (25,7))
plt.subplot(1,4,1)
plt.title('Product_category vs Number of visits')
sns.barplot(data = df_shop1,x = 'product_range',y= 'ProductRelated_Duration')
```

```
plt.subplot(1,4,2)
plt.title('ProductRelated_category vs Bounce_rate')
sns.lineplot(data = df_shop1,x = 'product_range', y = 'BounceRates')
plt.subplot(1,4,3)
plt.title('ProductRelated_category vs Exit_rate')
sns.scatterplot(data = df_shop1,x = 'product_range',y = 'ExitRates')
plt.show()
```



here we plotted the product categories count w.r.t Number of visits,bounce rate and Exit rates. 1. Firstly we plotted informational pages count with duration,where number of visitor spend more time with pages which are visited more .i.e more time a person visits a page, he/she spends more time on that page.

2. In the second line plot we showed the bounce rate w.r.t to product category count, where we can see that data is right skewed.i.e more percentage of bounce rate is concentarated at or below origin which is a not good metric in case of bounce rate , which resembles the visitors exit the page immediately. Without performing action for pages with less vistors. The bounce rate is more at landing page(100-200) categories must be develop in such a way that visitors spend a good amount of time and make some purchase or search for a product.Although the more number of visitors ages are good performers incase of Bounce rate which is less compared to pages which are less visited,the bounce rate has been gradually increased from product range 501 onwards.

3. In the third scatter plot we showed the Exit rates w.r.t product categories count, here the exit pages are more concentarted from 1-100 which is not a good sigh beacuse visitors are not drilling down much into website which causes the exit rate more. As the count of Adminsitrative pages visits increases the exit rate decreases, which indicates most of pages with lesser interaction like from(1-100) are higher in case of exit rates as they are not quite customer satsifaction. Care must be taken to these pages of lesser count for better performance and drill down details of product purchase providing to customers or visitors. As we go on along x-axis the number of page visits increases and exit rate deceases. The pages with less visits must be taken care.

[31]:
```
''' filtering the Special day and its distribution against revenue '''

c = df_shop1[['SpecialDay','dist_rev']]
c
```

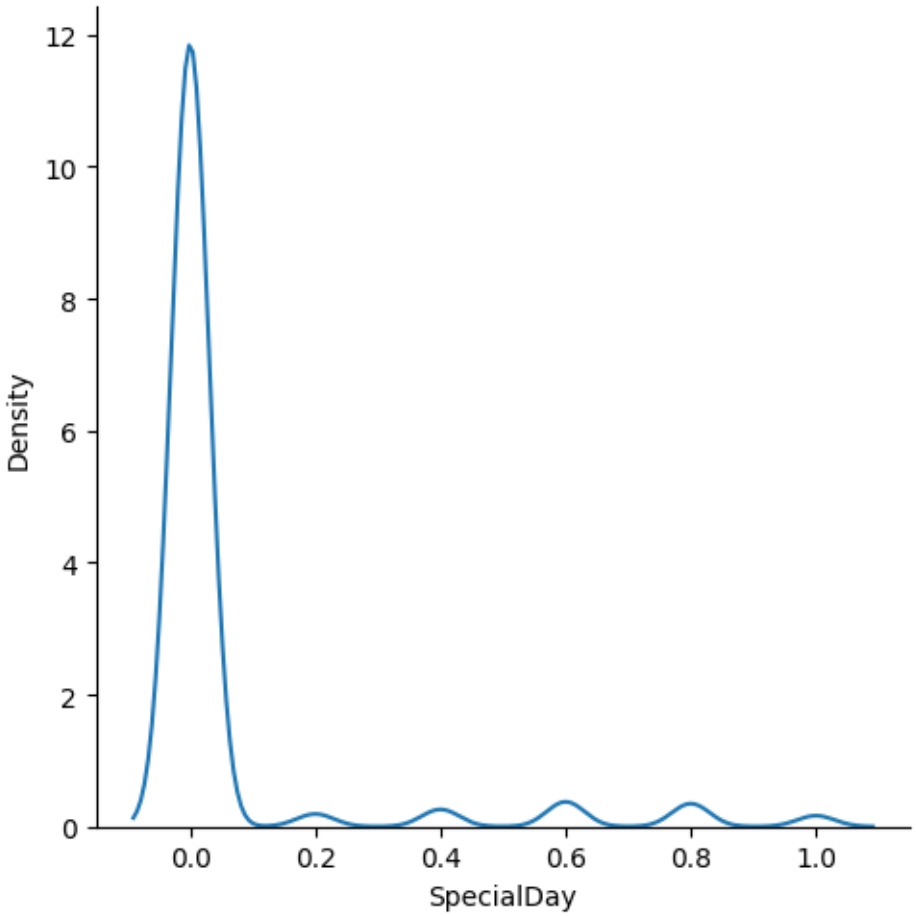
[31]:

	SpecialDay	dist_rev
0	0.0	0
1	0.0	0
2	0.0	0
3	0.0	0
4	0.0	0
…	…	…
12325	0.0	0
12326	0.0	0
12327	0.0	0
12328	0.0	0
12329	0.0	0

[12330 rows x 2 columns]

[32]:
```
''' plotting the distribution of Special day '''

sns.displot(df_shop['SpecialDay'],kind = 'kde')
```

[32]:
```
<seaborn.axisgrid.FacetGrid at 0x77fca1defd60>
```

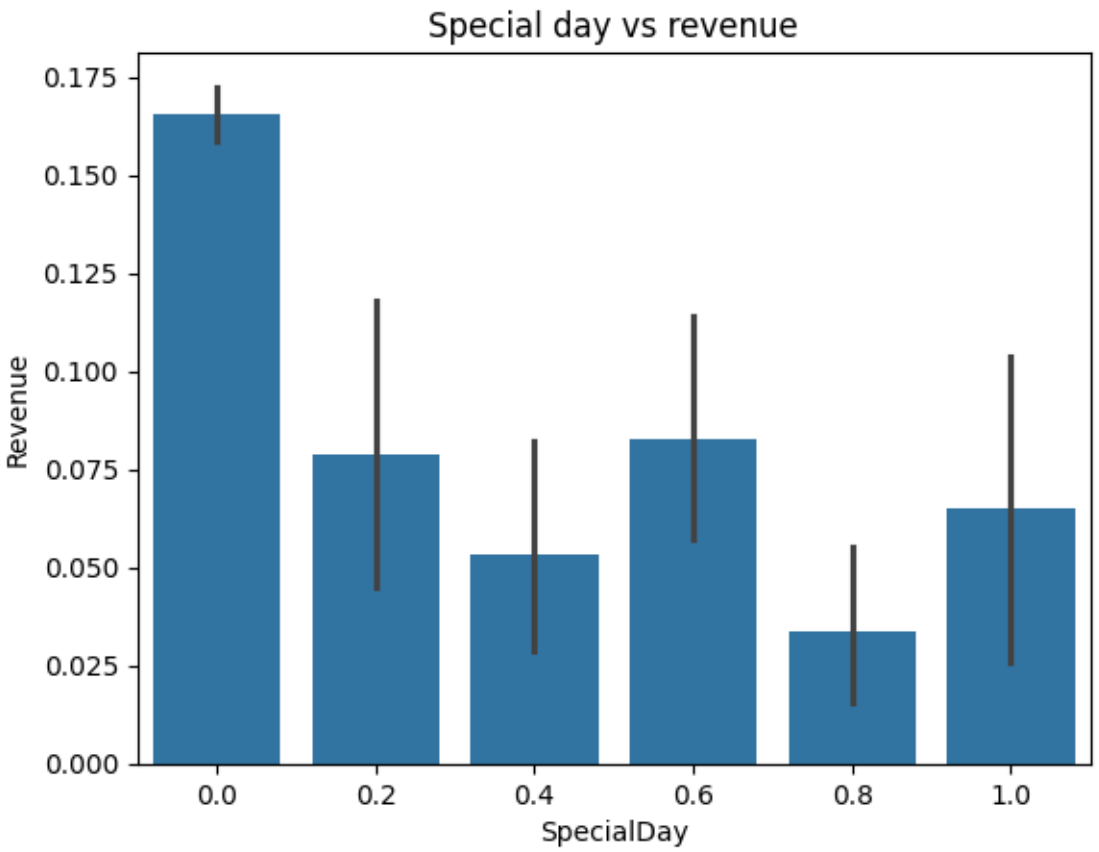


[33]:
```
''' finding the correlation betrween special day and revenue '''

c.corr()
```

[33]:
```
            SpecialDay  dist_rev
SpecialDay    1.000000 -0.082305
dist_rev     -0.082305  1.000000
```

[34]:
```
''' plotting a bra plot for special day vs reveue generated '''

plt.title('Special day vs revenue')
sns.barplot(data = df_shop1,x = 'SpecialDay',y = 'Revenue')
```

[34]:
```
<Axes: title={'center': 'Special day vs revenue'}, xlabel='SpecialDay',
ylabel='Revenue'>
```



here we plotted the bargraph for Special day column and the revenue gernerated. 1. Clearly, we can see that not a Special day - 0 has generated more revenue than the days which are near to Special day or day itself a Special day . 2. We can see the distribution of Special day from above dislot where the data is right skewed. 3. we also clulate dthe corelation between the special day and revenue which came to be negative i.e, Revenue is increaes on non Special day and revenue decreases on a Specila day.

[35]:
```
''' filtering the required categorical columns '''

e = df_shop[['Administrative','Informational','ProductRelated']]
```

[36]:
```
''' checking for customers who visited in all three above categories of␣
 ↪Adminsitartive,productrelated,informational '''

df_shop1['all_visits'] = e.all(axis = 1).apply(lambda x : 1 if x > 0 else 0)
```

[37]:
```
''' customers who visted all three caregorical pages '''

df_shop1[df_shop1['all_visits'] == 1]
```

[37]:

	Administrative	Administrative_Duration	Informational	\
29	1	6.000000	1	
57	4	56.000000	2	
103	2	31.000000	1	
109	6	326.250000	4	
161	2	58.000000	2	
…	…	…	…	
12287	8	167.910714	6	
12307	2	305.125000	3	
12311	1	0.000000	2	
12312	7	150.357143	1	
12313	3	16.000000	3	

	Informational_Duration	ProductRelated	ProductRelated_Duration	\
29	0.00	45	1582.750000	
57	120.00	36	998.741667	
103	16.00	36	2083.530952	
109	94.00	128	5062.213753	
161	22.00	31	829.166667	
…	…	…	…	
12287	547.75	111	6340.152381	
12307	368.25	27	1121.250000	
12311	211.25	144	4627.489571	
12312	9.00	221	11431.001240	
12313	86.00	15	2773.500000	

	BounceRates	ExitRates	PageValues	SpecialDay	…	OperatingSystems	\
29	0.043478	0.050821	54.179764	0.4	…	3	
57	0.000000	0.014736	19.447079	0.2	…	2	
103	0.000000	0.013510	0.000000	0.8	…	2	
109	0.000855	0.017918	0.000000	0.0	…	2	
161	0.030303	0.040606	0.000000	0.0	…	1	
…	…	…	…	…	…	…	
12287	0.003361	0.009432	44.219794	0.0	…	3	
12307	0.020000	0.042857	39.519807	0.0	…	3	
12311	0.001361	0.020664	0.000000	0.0	…	2	
12312	0.011149	0.021904	1.582473	0.0	…	2	
12313	0.000000	0.030000	78.811725	0.0	…	2	

	Browser	Region	TrafficType	VisitorType	Weekend	Revenue	\
29	2	1	1	Returning_Visitor	False	False	
57	2	4	1	Returning_Visitor	False	False	
103	2	4	3	Returning_Visitor	False	False	
109	5	1	3	Returning_Visitor	False	False	
161	1	1	1	Returning_Visitor	True	False	
…	…	…	…	…	…	…	
12287	2	6	2	Returning_Visitor	False	False	

```
12307        2       1            2  Returning_Visitor   False    False
12311        2       1            2  Returning_Visitor   False     True
12312        5       1            2  Returning_Visitor    True     True
12313        2       1            2  Returning_Visitor   False     True

       dist_rev  product_range all_visits
29            0          0-100          1
57            0          0-100          1
103           0          0-100          1
109           0        101-200          1
161           0          0-100          1
…           …            …        …
12287         0        101-200          1
12307         0          0-100          1
12311         1        101-200          1
12312         1        201-300          1
12313         1          0-100          1

[2167 rows x 21 columns]
```

[38]:
```
'''number customers who  visit all  three categorical columns '''

df_shop1[df_shop1['all_visits'] == 1].size
```

[38]:
```
45507
```

[39]:
```
'''number customers who doesn't visit all  three categorical columns '''

df_shop1[df_shop1['all_visits'] == 0]
```

[39]:
```
       Administrative  Administrative_Duration  Informational  \
0                   0                      0.0              0
1                   0                      0.0              0
2                   0                      0.0              0
3                   0                      0.0              0
4                   0                      0.0              0
…                 …                        …              …
12325               3                    145.0              0
12326               0                      0.0              0
12327               0                      0.0              0
12328               4                     75.0              0
12329               0                      0.0              0

       Informational_Duration  ProductRelated  ProductRelated_Duration  \
0                         0.0               1                 0.000000
1                         0.0               2                64.000000
2                         0.0               1                 0.000000
```

```
3                        0.0              2                2.666667
4                        0.0             10              627.500000
…                        …               …                  …
12325                    0.0             53             1783.791667
12326                    0.0              5              465.750000
12327                    0.0              6              184.250000
12328                    0.0             15              346.000000
12329                    0.0              3               21.250000

       BounceRates  ExitRates  PageValues  SpecialDay  … OperatingSystems  \
0         0.200000   0.200000    0.000000         0.0  …                 1
1         0.000000   0.100000    0.000000         0.0  …                 2
2         0.200000   0.200000    0.000000         0.0  …                 4
3         0.050000   0.140000    0.000000         0.0  …                 3
4         0.020000   0.050000    0.000000         0.0  …                 3
…            …          …          …          …  …              …
12325     0.007143   0.029031   12.241717         0.0  …                 4
12326     0.000000   0.021333    0.000000         0.0  …                 3
12327     0.083333   0.086667    0.000000         0.0  …                 3
12328     0.000000   0.021053    0.000000         0.0  …                 2
12329     0.000000   0.066667    0.000000         0.0  …                 3

       Browser  Region  TrafficType        VisitorType Weekend  Revenue  \
0            1       1            1  Returning_Visitor   False    False
1            2       1            2  Returning_Visitor   False    False
2            1       9            3  Returning_Visitor   False    False
3            2       2            4  Returning_Visitor   False    False
4            3       1            4  Returning_Visitor    True    False
…          …     …           …                  …     …      …
12325        6       1            1  Returning_Visitor    True    False
12326        2       1            8  Returning_Visitor    True    False
12327        2       1           13  Returning_Visitor    True    False
12328        2       3           11  Returning_Visitor   False    False
12329        2       1            2        New_Visitor    True    False

       dist_rev  product_range all_visits
0             0          0-100          0
1             0          0-100          0
2             0          0-100          0
3             0          0-100          0
4             0          0-100          0
…           …              …         …
12325         0          0-100          0
12326         0          0-100          0
12327         0          0-100          0
12328         0          0-100          0
12329         0          0-100          0
```

```
[10163 rows x 21 columns]
```

1. Here, we created a binary feature(0 & 1) out of of our data wher customer visit all three categorical columns of Administrative,Informational and ProductRelated.
2. it shows 1 if customer visit all three categorical pages else it will shows 0

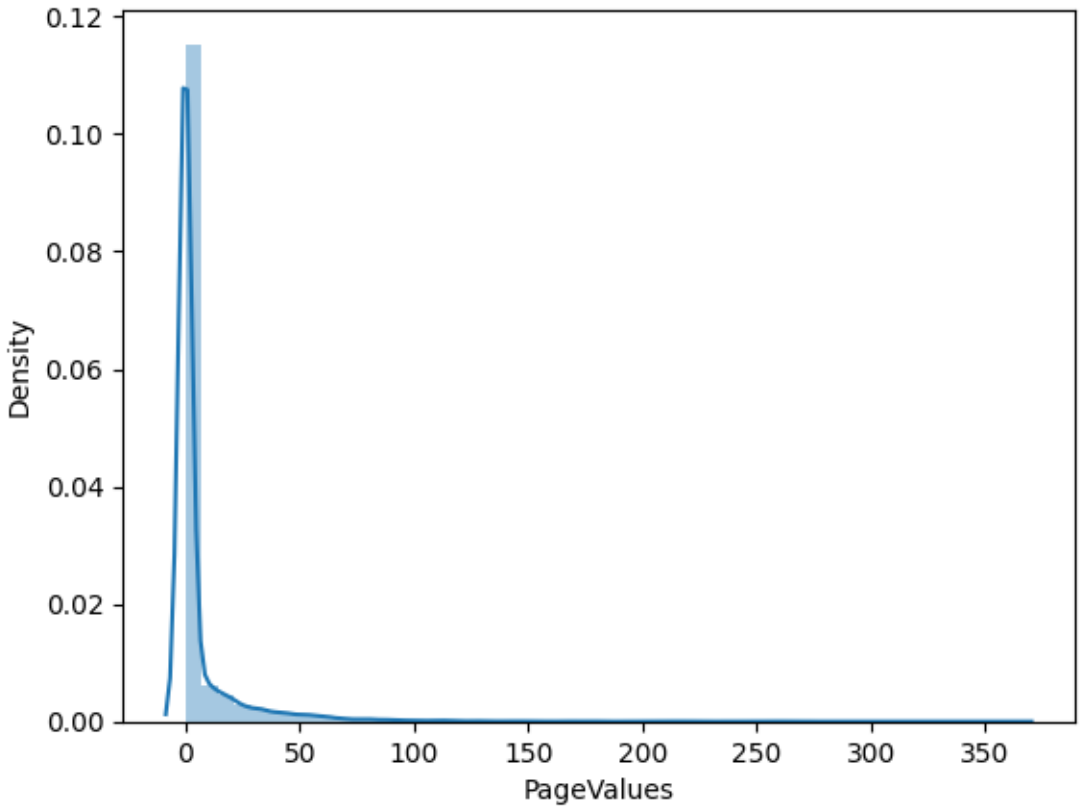
1. here the number of people who visited all three catgorical columns = 45507
2. the number of people who not visited all three categorical columns = 213423
3. By adding these two metrics we get the total number of people visited i.e size of dataframe = 258930

[39]:

[40]:
```
''' plottimg the distribution of pagevalues '''

sns.distplot(df_shop['PageValues'],kde = True)
```

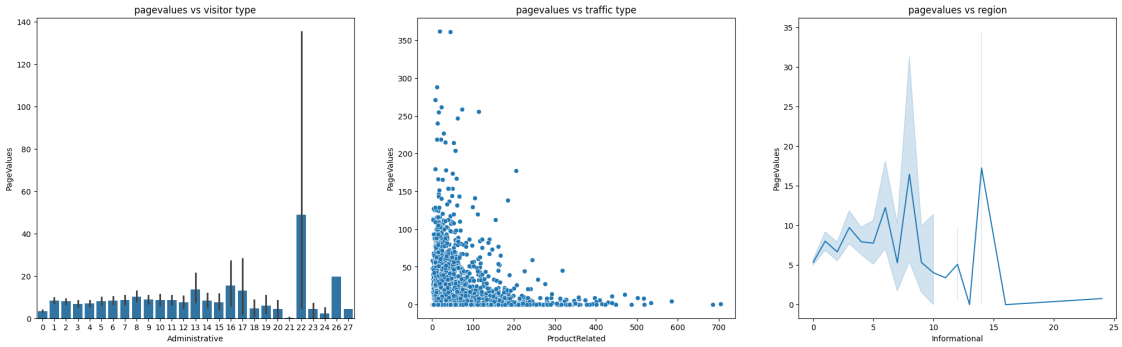
```
<ipython-input-40-8bbb9655f574>:3: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df_shop['PageValues'],kde = True)
```

[40]: `<Axes: xlabel='PageValues', ylabel='Density'>`



[41]:
```
''' filtering the required columns '''

g = df_shop[['PageValues','VisitorType','TrafficType','Region']]
```

[45]:
```
''' plttoing various plots for pages values against␣
 ↪vistortyoe,traffictype,region '''

plt.figure(figsize = (25,7))
plt.subplot(1,3,1)
plt.title('pagevalues vs visitor type')
sns.barplot(data = df_shop,y = 'PageValues',x = 'Administrative')
plt.subplot(1,3,2)
plt.title('pagevalues vs traffic type')
sns.scatterplot(data = df_shop,y = 'PageValues',x = 'ProductRelated')
plt.subplot(1,3,3)
plt.title('pagevalues vs region')
sns.lineplot(data = df_shop, y = 'PageValues',x = 'Informational')
plt.show()
```



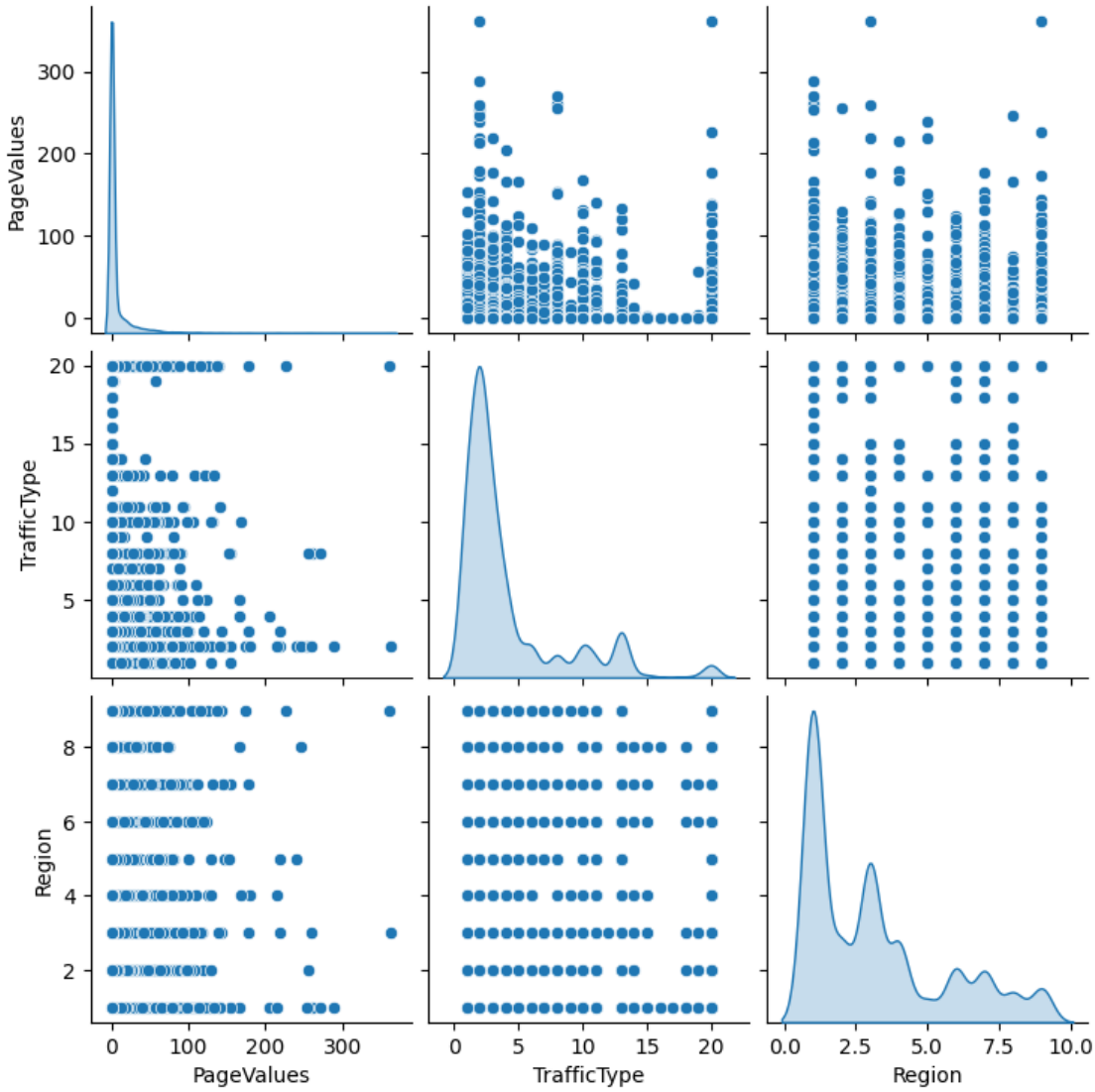
In the above plots we can observe pagevalues vs different parameters like Administrative category pages, informationalcategory pages and productt related category are plotted.

Page Value is the average value for a page that a user visited before landing on the goal page or completing an Ecommerce transaction (or both).

1. in the first plot(Administrative vs pagevalues) the categorical pages of administrative 22 and 26 has the higher pagevalues.rest of the pages are alomost of same values
2. in the second plot(Product_related vs pagevalues) the pages where visitors are not more concentrted of visiting, pagevalue is higher. product related pages from(0-200) are pages with higher pagevalues.
3. in the third poot(Informational vs pagevalues) the informational pages from (0-15) are with higher pages valuee. There is gradual increase in page values from 0- 5 an d the a sudden raise between (5-10) like pges 7-8 and informational pages with 15 also has high pagesvalues aslike 7-8.

[46]:
```
''' plotting pairplots for above three categorical caolumns against page values␣
↪'''

plt.figure(figsize = (20,9))
sns.pairplot(g,diag_kind = 'kde')
```

[46]:
```
<seaborn.axisgrid.PairGrid at 0x77fca3039ba0>

<Figure size 2000x900 with 0 Axes>
```



[47]:
```
df_shop.columns
```

[47]:
```
Index(['Administrative', 'Administrative_Duration', 'Informational',
       'Informational_Duration', 'ProductRelated', 'ProductRelated_Duration',
       'BounceRates', 'ExitRates', 'PageValues', 'SpecialDay', 'Month',
       'OperatingSystems', 'Browser', 'Region', 'TrafficType', 'VisitorType',
       'Weekend', 'Revenue'],
      dtype='object')
```

[48]:
```
''' filtering the amount of time spent on categorical columns of adminsitrative␣
 ↪informational productrealated '''

df_shop[['Administrative_Duration','Informational_Duration','ProductRelated_Duration']]
```

[48]:
```
       Administrative_Duration  Informational_Duration  \
0                          0.0                     0.0
1                          0.0                     0.0
2                          0.0                     0.0
3                          0.0                     0.0
4                          0.0                     0.0
…                          …                       …
12325                    145.0                     0.0
12326                      0.0                     0.0
12327                      0.0                     0.0
12328                     75.0                     0.0
12329                      0.0                     0.0

       ProductRelated_Duration
0                     0.000000
1                    64.000000
2                     0.000000
3                     2.666667
4                   627.500000
…                          …
12325              1783.791667
12326               465.750000
12327               184.250000
12328               346.000000
12329                21.250000

[12330 rows x 3 columns]
```

[50]:
```
''' fetching the minimum and maximum duration values for catgory column -␣
 ↪Administrative '''

df_shop['Administrative_Duration'].max(),df_shop['Administrative_Duration'].
 ↪min()
```

[50]:
```
(3398.75, 0.0)
```

[52]:
```
''' fetching the minimum and maximum duration values for catgory column -␣
 ↪informational '''

df_shop['Informational_Duration'].min(),df_shop['Informational_Duration'].max()
```

[52]:
```
(0.0, 2549.375)
```

[64]:
```
''' fetching the minimum and maximum duration values for catgory column -␣
 ↪productrelated '''
```

```
df_shop['ProductRelated_Duration'].min(),df_shop['ProductRelated_Duration'].
 ↪max()
```

[64]: (0.0, 63973.52223)

As we can see the duration columns([‘Administrative_Duration’,‘Informational_Duration’,‘ProductRelated_Durat
replicate the time spent by visitor for each category.

here we can observe the values for duartion are pretty large so we can create bins for these values and can have the visualization of duration against the conversion rate which is revenue generated by making any purchase.

[65]:
```
''' creating bins for duration of customer for column adminstarive category '''

bin1 = [-1,500,1000,1500,2000,2500,3000,3500]
label1 =␣
 ↪['<500','501-1000','1001-1500','1501-2000','2001-2500','2501-3000','3001+']
df_shop1['ad_bin'] = pd.cut(df_shop['Administrative_Duration'],bins = bin1,␣
 ↪labels = label1)
```

[66]:
```
''' creating bins for duration of customer for column informational category '''


bin2 = [-1,500,1000,1500,2000,2500,3000]
label2 = ['<500','501-1000','1001-1500','1501-2000','2001-2500','2500+']
df_shop1['id_bin'] = pd.cut(df_shop['Informational_Duration'],bins = bin2,␣
 ↪labels = label2)
```

[67]:
```
''' creating bins for duration of customer for column productrelated category␣
 ↪'''


bin3 = [-1,1000,2000,3000,4000,5000,6000,7000]
label3 =␣
 ↪['<1000','1001-2000','2001-3000','3001-4000','4001-5000','5000-6000','6000+']
df_shop1['pr_bin'] = pd.cut(df_shop['ProductRelated_Duration'],bins = bin3,␣
 ↪labels = label3)
```

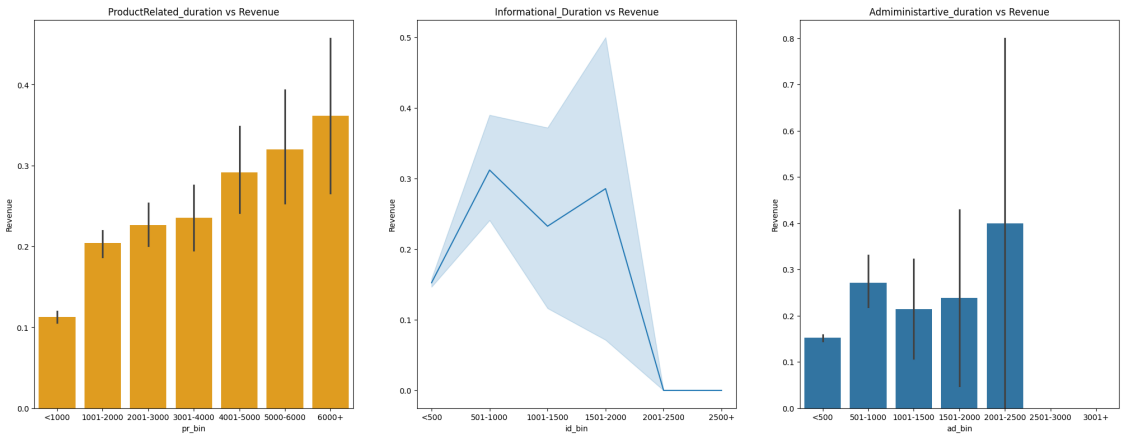
[68]:
```
''' querying  revenue created  for all three columns '''

df_shop1[['ad_bin','id_bin','pr_bin','Revenue']]
```

[68]:

	ad_bin	id_bin	pr_bin	Revenue
0	<500	<500	<1000	False
1	<500	<500	<1000	False
2	<500	<500	<1000	False
3	<500	<500	<1000	False
4	<500	<500	<1000	False

```
...       ...     ...          ...       ...
12325    <500    <500   1001-2000     False
12326    <500    <500       <1000     False
12327    <500    <500       <1000     False
12328    <500    <500       <1000     False
12329    <500    <500       <1000     False

[12330 rows x 4 columns]
```

[69]:
```
''' plotting revenue vs informational,productrelated and adminstrative␣
 ↪categories '''

plt.figure(figsize = (25,9))


plt.subplot(1,3,1)
plt.title('ProductRelated_duration vs Revenue')
sns.barplot(data = df_shop1,x = 'pr_bin',y = 'Revenue',color = 'orange')


plt.subplot(1,3,2)
plt.title('Informational_Duration vs Revenue')
sns.lineplot(data = df_shop1,x = 'id_bin', y = 'Revenue')


plt.subplot(1,3,3)
plt.title('Adminimistartive_duration vs Revenue')
sns.barplot(data = df_shop1,x = 'ad_bin', y = 'Revenue')


plt.show()
```

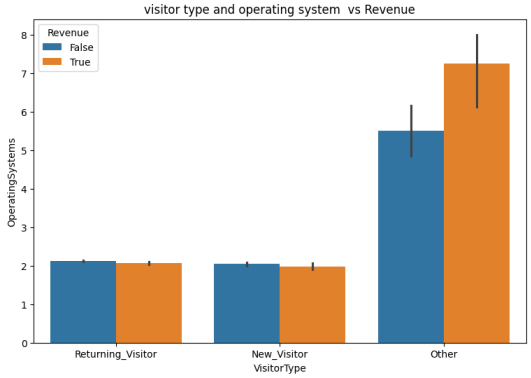


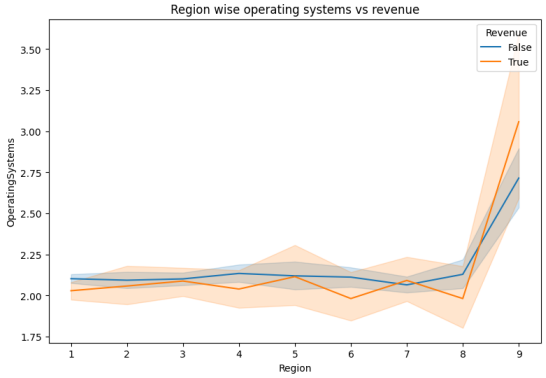
in the above three plots we see the revenue generated by time spending on categorical pages - administartive pages,informational pages and productrelated pages.

1. In the first plot(product vs revenue) as long as customer spends more time in the page has more rate for conversions.the bins from 4000 - 6000+ has increased the revenue than other bins of lesser duration.
2. In the second plot(informational vs revenue) the more and more customer spends on informational pages the conversion rates has been decreased, the ideal duration that is the average duration in inforamtional pages makes the god conversion rates.
3. in the third plot(Administrative vs revenue) the revenue is more generated from the median or middle value of duration of time spent by customer on thsi categorical pages. we can see eventually the conversion rated dipped down with more time spent.

[70]:
```
''' filtering columns operatingsystems,visitor type,region,reveneue '''

i = df_shop[['OperatingSystems','VisitorType','Region','Revenue']]
```

[71]:
```
''' plotting visitor type and operating system vs revenue and region operating␣
 ↪system and revenue '''

plt.figure(figsize = (20,6))
plt.subplot(1,2,1)
plt.title('visitor type and operating system  vs Revenue')
sns.barplot(data = df_shop, y = 'OperatingSystems',x = 'VisitorType', hue =␣
 ↪'Revenue'  )

plt.subplot(1,2,2)
plt.title('Region wise operating systems vs revenue')
sns.lineplot(data = df_shop, x = 'Region',y = 'OperatingSystems',hue =␣
 ↪'Revenue')

plt.show()
```





the above we can see plots are visitor type operating sysytem and revenue generated and the second one is region wise revenue ge=nerated by operating sysytme : 1. from the first plot we can see that the new an old users has almost similar ratio of conversion to happen and conversion not to happen. while for other users the conversion rate is high along with non conversion which is also high.

2. In the second plot the regions from 1 - 8 almost has similar way of conversion rates of customers of having a conversion or not having a conversion, the non conversion is slightly higher for 1-8 regions, whereas for region number 9 the conversion rate is far higher than other regions alongside with non conversion rates which is also higher than other region convesion rates.

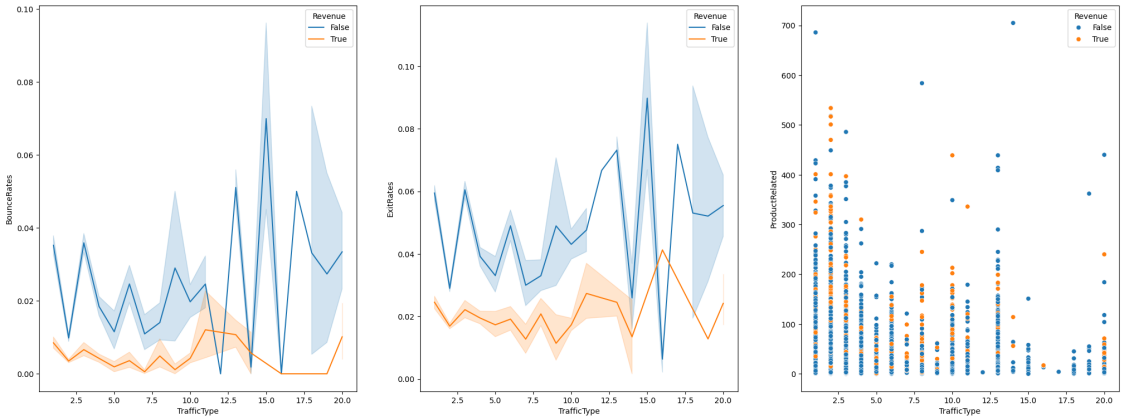
[72]:
```
''' creating new datafram by filtering required columns from df '''

j  =␣
 ↪df_shop[['TrafficType','Administrative','Informational','ProductRelated','Revenue']]
k = df_shop[['TrafficType','BounceRates','ExitRates','Revenue']]
```

[73]:
```
'''' minimum an dmaximum values for traffic columns '''

df_shop['TrafficType'].min(),df_shop['TrafficType'].max()
```

[73]:
```
(1, 20)
```

[74]:
```
''' plotting revenue for each traffic type and parmeters␣
 ↪bouncerate,exitrate,productrelated '''

plt.figure(figsize = (25,9))

plt.subplot(1,3,1)
sns.lineplot(data =df_shop,x = 'TrafficType', y = 'BounceRates',hue = 'Revenue')

plt.subplot(1,3,2)
sns.lineplot(data =df_shop, x = 'TrafficType',y = 'ExitRates',hue = 'Revenue')

plt.subplot(1,3,3)
sns.scatterplot(data = df_shop, x = 'TrafficType',y = 'ProductRelated',hue =␣
 ↪'Revenue')
```

[74]:
```
<Axes: xlabel='TrafficType', ylabel='ProductRelated'>
```



we can three plots of traffic type with bouncerates,exits rates and product and their values of creating a conversion(revenue)

1. In the first plot (traffictype vs bounce rates) the traffic type is way smoother with low bounce rates which generaterevenue. higher bounce rate and higher traffic doesn't make good conversions of customers.
2. In the second plot(Traffictype vs exit rates) thetraffic is with sight increase craeted a good conversion rate, whereas higher exitrates and higher traffic created less conversions.
3. In the third plot(traffictype vs product) customer spending an average time with low to medium traffic made more conversion rate.

[74]:

Here, we worked with shopping dataset of customers which resembles the customer behaviour with online ecommerce website.

We did the exploratory data analysis and visulization for better understand data and drawed some important insights from our analysis

[74]: