Store analysis

May 17, 2025

[1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import math as m
```

[2]:
```
Data = pd.read_csv(r"D:\Project\Quuantium\Task 2\QVI_data.csv")
```

[3]:
```
Data
```

[3]:

	LYLTY_CARD_NBR	DATE	STORE_NBR	TXN_ID	PROD_NBR	PROD_NAME	PROD_QTY	TOT_SALES	PACK_SIZE	BRAND	LIFESTAGE	PREMIUM_CUSTOMER
0	1000	2018-10-17	1	1	5	Natural Chip Compny SeaSalt175g	2	6.0	175	NATURAL	YOUNG SINGLES/COUPLES	Premium
1	1002	2018-09-16	1	2	58	Red Rock Deli Chikn&Garlic Aioli 150g	1	2.7				
2	1003	2019-03-07	1	3	52	Grain Waves Sour Cream&Chives 210G	1	3.6				
3	1003	2019-03-08	1	4	106	Natural ChipCo Hony Soy Chckn175g	1	3.0				
4	1004	2018-11-02	1	5	96	WW Original Stacked Chips 160g	1	1.9				
…	…	…	…	…	…	…	…	…				
264829	2370701	2018-12-08	88	240378	24	Grain Waves Sweet Chilli 210g	2	7.2				
264830	2370751	2018-10-01	88	240394	60	Kettle Tortilla ChpsFeta&Garlic 150g	2	9.2				
264831	2370961	2018-10-24	88	240480	70	Tyrrells Crisps Lightly Salted 165g	2	8.4				
264832	2370961	2018-10-27	88	240481	65	Old El Paso Salsa Dip Chnky Tom Ht300g	2	10.2				
264833	2373711	2018-12-14	88	241815	16	Smiths Crinkle Chips Salt & Vinegar 330g	2	11.4				

```
1              150          RRD  YOUNG SINGLES/COUPLES       Mainstream
2              210     GRNWVES          YOUNG FAMILIES           Budget
3              175     NATURAL          YOUNG FAMILIES           Budget
4              160  WOOLWORTHS  OLDER SINGLES/COUPLES       Mainstream
…              …          …                    …                …
264829         210     GRNWVES          YOUNG FAMILIES       Mainstream
264830         150      KETTLE          YOUNG FAMILIES          Premium
264831         165    TYRRELLS          OLDER FAMILIES           Budget
264832         300         OLD          OLDER FAMILIES           Budget
264833         330      SMITHS  YOUNG SINGLES/COUPLES       Mainstream

[264834 rows x 12 columns]
```

[4]:
```
Data['DATE'] = pd.to_datetime(Data['DATE'])
start_date = '2019-02-01'
end_date = '2019-04-30'

Data3 = Data[(Data['DATE'] >= start_date) & (Data['DATE'] <= end_date)]

Data4  =Data[(Data['DATE'] < start_date)]
```

[5]:
```
def str_input():
    return input("Enter value:  ").split(',')

str_input = list(map(int,str_input()))
Data2 = Data3[Data3['STORE_NBR'].isin(str_input)][['STORE_NBR','TOT_SALES']].
 ↪groupby('STORE_NBR')['TOT_SALES'].sum().sort_values(ascending = False)
print("Total Sales for selected stores:", Data2)
```

```
Enter value:  77,88,86

Total Sales for selected stores: STORE_NBR
88    4286.8
86    2788.2
77     777.0
Name: TOT_SALES, dtype: float64
```

[7]:
```
def str_input():
    return input("Enter value:  ").split(',')

str_input = list(map(int,str_input()))
Data6 = Data[Data['STORE_NBR'].isin(str_input)][['STORE_NBR','TOT_SALES']].
 ↪groupby('STORE_NBR')['TOT_SALES'].sum().sort_values(ascending = False)
print("Total Sales for selected stores:", Data6)
```

```
Enter value:  77,86,88

Total Sales for selected stores: STORE_NBR
88    16333.25
```

```
86    10635.35
77     3040.00
Name: TOT_SALES, dtype: float64
```

[120]:
```
def str_input():
    return input("Enter value:  ").split(',')

str_input = list(map(int,str_input()))

Data7 = Data3[Data3['STORE_NBR'].isin(str_input)][['STORE_NBR','PROD_QTY']].
 ↪groupby('STORE_NBR')['PROD_QTY'].sum().sort_values(ascending = False)
print("Total Sales for selected stores:", Data7)
```

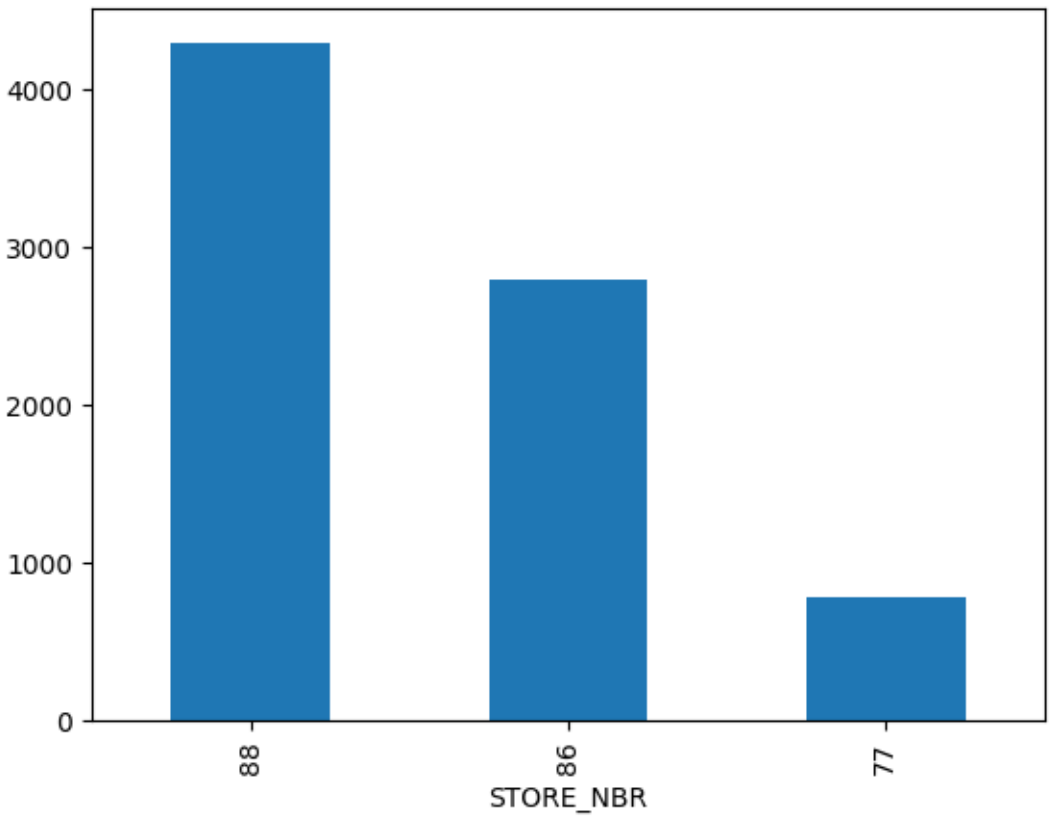
```
Enter value:  77,86,88

Total Sales for selected stores: STORE_NBR
88    972
86    815
77    234
Name: PROD_QTY, dtype: int64
```

[190]:
```
Data8 = Data[Data['STORE_NBR'].isin(str_input)][['STORE_NBR','PROD_QTY']].
 ↪groupby('STORE_NBR')['PROD_QTY'].sum().sort_values(ascending = False)
print("Total Sales for selected stores:", Data8)
```

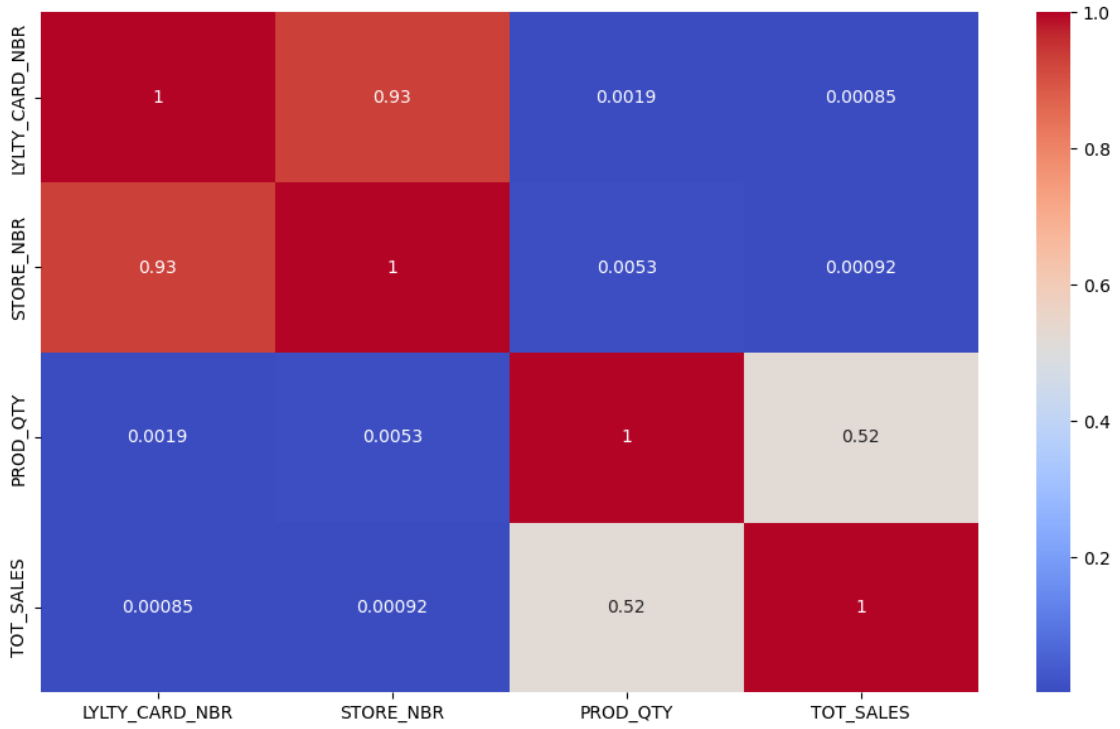
```
Total Sales for selected stores: STORE_NBR
88    3718
86    3066
77     872
Name: PROD_QTY, dtype: int64
```

[8]:
```
Data2.plot(kind = 'bar')
```

[8]:
```
<Axes: xlabel='STORE_NBR'>
```



[9]:
```
Datas = Data4[['LYLTY_CARD_NBR','STORE_NBR','PROD_QTY','TOT_SALES']]
Datass = Datas.corr()

plt.figure(figsize = (12,7))
sns.heatmap(data = Datass ,annot=True, cmap='coolwarm')
plt.show()
```



[ ]:

[10]:
```
def str_input():
    return input("Enter value:  ").split(',')

str_input = list(map(int,str_input()))
Data10 = Data3[Data3['STORE_NBR'].isin(str_input)][['STORE_NBR','TOT_SALES']].
 ↪groupby('STORE_NBR')['TOT_SALES'].sum().sort_values(ascending = False)
print("Total Sales for selected stores:", Data10)
```

```
Enter value:  77

Total Sales for selected stores: STORE_NBR
77    777.0
Name: TOT_SALES, dtype: float64
```

[11]:
```
Data11 = Data.groupby('STORE_NBR')['TOT_SALES'].sum()
Data11.iloc[75:88].sort_values(ascending = False)
```

[11]:
```
STORE_NBR
88    16333.25
81    14361.95
79    11831.20
80    11756.90
```

```
86    10635.35
83     9924.90
78     9381.25
84     5396.30
82     4103.50
87     3991.60
77     3040.00
85       13.90
76        6.00
Name: TOT_SALES, dtype: float64
```

[12]:
```
Data12 = Data.groupby('STORE_NBR')['TOT_SALES'].sum().sort_values(ascending =␣
 ↪True)
Data12.iloc[57:75]
```

[12]:
```
STORE_NBR
41     2570.20
268    2601.05
195    2608.25
163    2635.70
6      2684.90
53     2715.05
214    2720.40
176    2752.90
233    2826.90
255    2835.30
185    2868.60
187    2909.70
205    2966.80
220    3008.20
50     3009.80
46     3023.45
141    3025.40
77     3040.00
Name: TOT_SALES, dtype: float64
```

[13]:
```
Data['DATE'] = pd.to_datetime(Data['DATE'])
Data['Month_Year'] = Data['DATE'].dt.strftime("%m/%Y")
```

[14]:
```
filtered_data = Data[Data['STORE_NBR'].isin(Sorted)]
Data13 = filtered_data.groupby(['STORE_NBR','Month_Year'])['TOT_SALES'].sum()

Data13
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[14], line 1
```

```
----> 1 filtered_data = Data[Data['STORE_NBR'].isin(Sorted)]
      2 Data13 = filtered_data.groupby(['STORE_NBR','Month_Year'])['TOT_SALES']
 ↪sum()
      4 Data13

NameError: name 'Sorted' is not defined
```

[ ]:
```
Sorted = [41,268,195,163,6,53,214,176,233,255,185,187,205,220,50,46,141,77]
Data15 = pd.DataFrame({"values": Data13[Sorted]})
Data15
```

[ ]:
```
piv = Data15.pivot_table(index = "Month_Year", columns = "STORE_NBR", values =␣
 ↪"values")
piv
```

[ ]:
```
piv.corr(method = "pearson")
```

[ ]:
```
plt.figure(figsize = (12,7))

Known = piv[[41,77]]
Known.plot()
plt.show()
```

[15]:
```
Data
```

[15]:

	LYLTY_CARD_NBR	DATE	STORE_NBR	TXN_ID	PROD_NBR	PROD_NAME	PROD_QTY	TOT_SALES
0	1000	2018-10-17	1	1	5	Natural Chip Compny SeaSalt175g	2	6.0
1	1002	2018-09-16	1	2	58	Red Rock Deli Chikn&Garlic Aioli 150g	1	2.7
2	1003	2019-03-07	1	3	52	Grain Waves Sour Cream&Chives 210G	1	3.6
3	1003	2019-03-08	1	4	106	Natural ChipCo Hony Soy Chckn175g	1	3.0
4	1004	2018-11-02	1	5	96	WW Original Stacked Chips 160g	1	1.9
…	…	…	…	…	…	…	…	…
264829	2370701	2018-12-08	88	240378	24	Grain Waves Sweet Chilli 210g	2	7.2
264830	2370751	2018-10-01	88	240394	60	Kettle Tortilla ChpsFeta&Garlic 150g	2	9.2
264831	2370961	2018-10-24	88	240480	70			
264832	2370961	2018-10-27	88	240481	65			
264833	2373711	2018-12-14	88	241815	16			

```
264831   Tyrrells Crisps     Lightly Salted 165g         2        8.4
264832  Old El Paso Salsa   Dip Chnky Tom Ht300g         2       10.2
264833  Smiths Crinkle Chips Salt & Vinegar 330g         2       11.4

        PACK_SIZE       BRAND              LIFESTAGE PREMIUM_CUSTOMER  \
0             175     NATURAL  YOUNG SINGLES/COUPLES          Premium
1             150         RRD  YOUNG SINGLES/COUPLES       Mainstream
2             210     GRNWVES         YOUNG FAMILIES           Budget
3             175     NATURAL         YOUNG FAMILIES           Budget
4             160  WOOLWORTHS  OLDER SINGLES/COUPLES       Mainstream
…             …         …                   …               …
264829        210     GRNWVES         YOUNG FAMILIES       Mainstream
264830        150      KETTLE         YOUNG FAMILIES          Premium
264831        165    TYRRELLS         OLDER FAMILIES           Budget
264832        300         OLD         OLDER FAMILIES           Budget
264833        330      SMITHS  YOUNG SINGLES/COUPLES       Mainstream

       Month_Year
0         10/2018
1         09/2018
2         03/2019
3         03/2019
4         11/2018
…           …
264829    12/2018
264830    10/2018
264831    10/2018
264832    10/2018
264833    12/2018

[264834 rows x 13 columns]
```

[16]:
```
Data16 = Data.groupby(['STORE_NBR','Month_Year'])['TOT_SALES'].sum().
 ↪sort_values(ascending = True)
Data16.iloc[178:201]
```

[16]:
```
STORE_NBR  Month_Year
117        12/2018       27.9
263        08/2018       28.0
158        07/2018       28.0
52         07/2018       28.3
258        04/2019       28.4
198        05/2019       28.5
159        02/2019       28.6
           06/2019       28.6
161        04/2019       28.8
258        12/2018       29.0
```

```
204        01/2019       29.0
140        10/2018       29.5
146        05/2019       29.6
132        01/2019       29.6
192        04/2019       29.7
244        04/2019       29.8
158        08/2018       29.8
177        06/2019       30.2
140        12/2018       30.3
158        10/2018       30.3
42         05/2019       30.3
99         03/2019       30.3
192        09/2018       30.4
Name: TOT_SALES, dtype: float64
```

[17]:
```
Sort = [109, 191, 196, 229, 97, 102, 105, 232, 57, 172, 113, 225, 62, 236, 227,␣
 ↪155, 86, 247, 13, 164, 106, 55, 138]
Data16 = pd.DataFrame({"values": Data16[Sort]})
```

[18]:
```
piv2 = Data16.pivot_table(index = "Month_Year", columns = "STORE_NBR", values =␣
 ↪"values")
piv2.corr(method = "pearson")
```

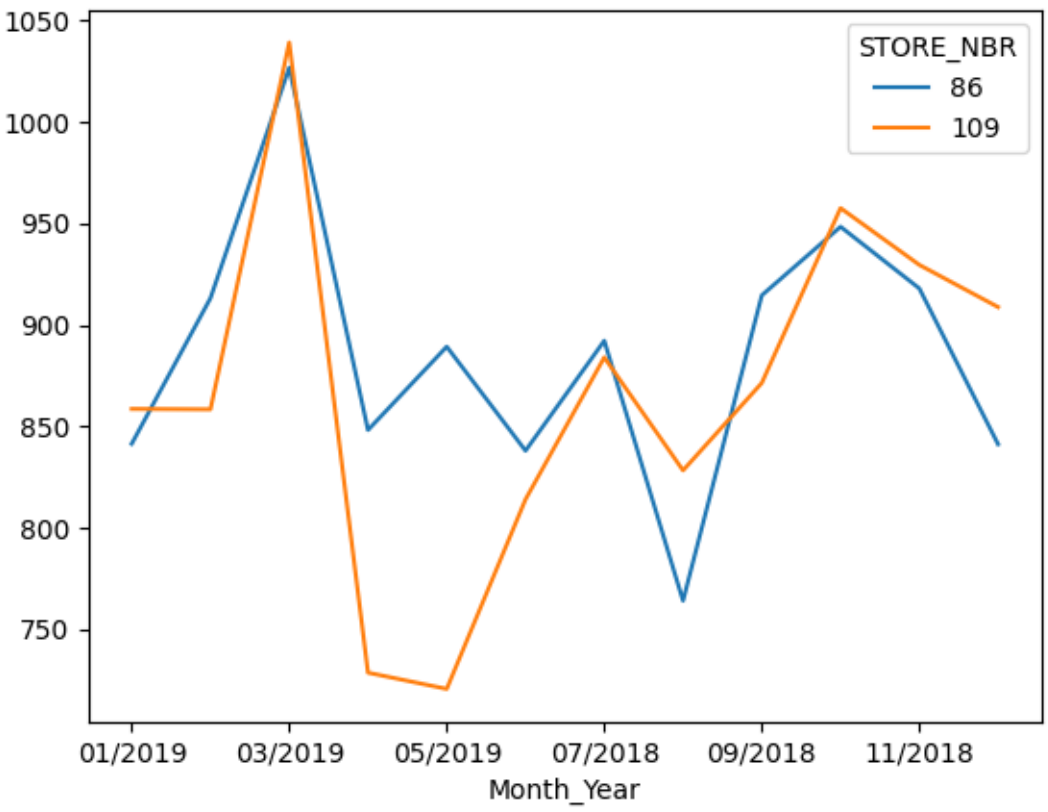
[18]:
```
STORE_NBR       13        55        57        62        86        97   \
STORE_NBR
13         1.000000 -0.125341 -0.291218  0.365314  0.457947 -0.373037
55        -0.125341  1.000000 -0.039301  0.181823  0.043906  0.495256
57        -0.291218 -0.039301  1.000000 -0.428165 -0.402687  0.221201
62         0.365314  0.181823 -0.428165  1.000000  0.276452 -0.184301
86         0.457947  0.043906 -0.402687  0.276452  1.000000 -0.015617
97        -0.373037  0.495256  0.221201 -0.184301 -0.015617  1.000000
102       -0.377415  0.418809 -0.139586 -0.206387 -0.226422  0.578719
105       -0.059766  0.124132  0.301428  0.113294 -0.202451  0.334039
106        0.049336  0.181864 -0.658612  0.634354  0.510548  0.203434
109        0.324289  0.326968 -0.124668  0.426023  0.643075  0.241536
113       -0.161963  0.306164 -0.087082  0.287274  0.043835  0.548974
138        0.284311  0.500047 -0.001387  0.172155  0.250447  0.286776
155       -0.228967  0.174382 -0.232252  0.339800  0.326149  0.275949
164        0.357477  0.060884  0.060840 -0.006044 -0.117970  0.140764
172       -0.091999  0.250338  0.665384 -0.100249 -0.156398  0.128774
191        0.733656  0.018181  0.081015  0.227897  0.043345 -0.359215
196        0.166098  0.101949 -0.113210  0.049385  0.081832  0.240357
225        0.043419  0.338013 -0.005863  0.005783 -0.109479  0.224941
227        0.289917  0.354941  0.106827 -0.028706  0.393785  0.403000
229        0.508201  0.234072 -0.335684  0.426077  0.596886 -0.120038
232       -0.084443 -0.320462 -0.100878  0.461276  0.327006  0.141757
236       -0.597718 -0.206578  0.237461 -0.334550 -0.164982  0.162069
```

```
247         0.167139  0.096625  0.237256 -0.295701  0.250601 -0.106598

STORE_NBR       102       105       106       109  …       164       172  \
STORE_NBR                                          …
13         -0.377415 -0.059766  0.049336  0.324289  …  0.357477 -0.091999
55          0.418809  0.124132  0.181864  0.326968  …  0.060884  0.250338
57         -0.139586  0.301428 -0.658612 -0.124668  …  0.060840  0.665384
62         -0.206387  0.113294  0.634354  0.426023  … -0.006044 -0.100249
86         -0.226422 -0.202451  0.510548  0.643075  … -0.117970 -0.156398
97          0.578719  0.334039  0.203434  0.241536  …  0.140764  0.128774
102         1.000000 -0.303843  0.088393  0.057036  … -0.324841  0.000426
105        -0.303843  1.000000 -0.084228  0.117184  …  0.754963 -0.099642
106         0.088393 -0.084228  1.000000  0.363415  … -0.132514 -0.452421
109         0.057036  0.117184  0.363415  1.000000  …  0.065696 -0.115734
113         0.388871  0.519296  0.331634  0.559222  …  0.190053 -0.359782
138         0.317674 -0.120865 -0.089779  0.307574  … -0.103974  0.600687
155         0.171003 -0.345718  0.684674 -0.009058  … -0.494106  0.214429
164        -0.324841  0.754963 -0.132514  0.065696  …  1.000000 -0.275936
172         0.000426 -0.099642 -0.452421 -0.115734  … -0.275936  1.000000
191        -0.454167  0.374381 -0.327944  0.179012  …  0.537125 -0.004155
196        -0.283326  0.730895  0.110802  0.170177  …  0.894100 -0.363440
225        -0.023039  0.169544  0.053068 -0.481235  …  0.134784  0.304429
227        -0.009479  0.159843  0.054562  0.652795  …  0.365061  0.042652
229        -0.406497  0.407354  0.233852  0.437263  …  0.417885 -0.147605
232        -0.251850  0.176014  0.599607  0.287077  … -0.019865 -0.136244
236        -0.245020  0.520565 -0.022502 -0.130827  …  0.294461 -0.329880
247        -0.460621 -0.131195 -0.155990 -0.296431  …  0.033993  0.403578

STORE_NBR       191       196       225       227       229       232  \
STORE_NBR
13          0.733656  0.166098  0.043419  0.289917  0.508201 -0.084443
55          0.018181  0.101949  0.338013  0.354941  0.234072 -0.320462
57          0.081015 -0.113210 -0.005863  0.106827 -0.335684 -0.100878
62          0.227897  0.049385  0.005783 -0.028706  0.426077  0.461276
86          0.043345  0.081832 -0.109479  0.393785  0.596886  0.327006
97         -0.359215  0.240357  0.224941  0.403000 -0.120038  0.141757
102        -0.454167 -0.283326 -0.023039 -0.009479 -0.406497 -0.251850
105         0.374381  0.730895  0.169544  0.159843  0.407354  0.176014
106        -0.327944  0.110802  0.053068  0.054562  0.233852  0.599607
109         0.179012  0.170177 -0.481235  0.652795  0.437263  0.287077
113         0.062261  0.244345 -0.186722  0.139154  0.222562  0.181658
138         0.115548 -0.080774  0.269426  0.311361  0.298551 -0.160849
155        -0.637781 -0.243707  0.313403 -0.134664 -0.076662  0.547102
164         0.537125  0.894100  0.134784  0.365061  0.417885 -0.019865
172        -0.004155 -0.363440  0.304429  0.042652 -0.147605 -0.136244
191         1.000000  0.220600  0.141753  0.111003  0.533908 -0.402733
196         0.220600  1.000000  0.049180  0.402251  0.511064  0.247312
```

```
225        0.141753  0.049180  1.000000 -0.265452  0.207075 -0.264708
227        0.111003  0.402251 -0.265452  1.000000  0.141159  0.083682
229        0.533908  0.511064  0.207075  0.141159  1.000000  0.071268
232       -0.402733  0.247312 -0.264708  0.083682  0.071268  1.000000
236       -0.324656  0.489581 -0.219614 -0.037733 -0.033110  0.327655
247        0.105910 -0.013787  0.552432  0.131065  0.136757 -0.216490

STORE_NBR       236       247
STORE_NBR
13        -0.597718  0.167139
55        -0.206578  0.096625
57         0.237461  0.237256
62        -0.334550 -0.295701
86        -0.164982  0.250601
97         0.162069 -0.106598
102       -0.245020 -0.460621
105        0.520565 -0.131195
106       -0.022502 -0.155990
109       -0.130827 -0.296431
113        0.190227 -0.679650
138       -0.641239  0.073685
155       -0.127099  0.201025
164        0.294461  0.033993
172       -0.329880  0.403578
191       -0.324656  0.105910
196        0.489581 -0.013787
225       -0.219614  0.552432
227       -0.037733  0.131065
229       -0.033110  0.136757
232        0.327655 -0.216490
236        1.000000 -0.045046
247       -0.045046  1.000000

[23 rows x 23 columns]
```

[19]:
```
plt.figure(figsize = (12,7))

Known1 = piv2[[86,109]]
Known1.plot()
plt.show()
```

```
<Figure size 1200x700 with 0 Axes>
```



[ ]:

[20]:
```
Data20 = Data.groupby('STORE_NBR')['TOT_SALES'].sum().sort_values(ascending =
↪True)#,'Month_Year'
Data20.iloc[250:]
```

[20]:
```
STORE_NBR
125    13352.85
123    13468.40
210    13689.25
238    13708.40
261    13859.75
95     13915.50
217    13993.60
181    14108.45
72     14204.40
130    14289.65
201    14298.70
81     14361.95
26     14469.30
203    14551.60
```

```
4      14647.65
199    14797.00
58     15251.45
237    15539.50
40     15559.50
165    15973.75
88     16333.25
226    17605.45
Name: TOT_SALES, dtype: float64
```

[22]:
```
Data21 = Data.groupby(['STORE_NBR','Month_Year'])['TOT_SALES'].sum().
 ↪sort_values(ascending = True)

Sorts =␣
 ↪[125,123,210,238,261,95,217,181,72,130,201,81,26,203,4,199,58,237,40,165,88,226]
Data22 = pd.DataFrame({"values" :Data21[Sorts]})
```

[25]:
```
piv3 = Data22.pivot_table(index = "Month_Year", columns = "STORE_NBR", values =␣
 ↪"values")
piv3.corr(method = "pearson")
```

[25]:
```
STORE_NBR       4         26        40        58        72        81   \
STORE_NBR
4          1.000000  0.145229 -0.239541  0.462517  0.642654 -0.091353
26         0.145229  1.000000  0.024883  0.301079  0.049001  0.703522
40        -0.239541  0.024883  1.000000  0.060504  0.182486  0.565346
58         0.462517  0.301079  0.060504  1.000000  0.311481  0.147287
72         0.642654  0.049001  0.182486  0.311481  1.000000  0.152143
81        -0.091353  0.703522  0.565346  0.147287  0.152143  1.000000
88        -0.319831  0.439615 -0.141734  0.224899 -0.348883  0.197483
95        -0.038154 -0.133565 -0.211049 -0.356401  0.017109 -0.291601
123       -0.056994 -0.299583 -0.431312  0.094355  0.083195 -0.513938
125        0.312848  0.609910 -0.468838  0.113932  0.033888  0.187666
130        0.010957 -0.143166  0.416386 -0.270807  0.414191 -0.093296
165        0.249888  0.167009  0.372374  0.121878  0.002086  0.329747
181        0.446083  0.463871  0.150957  0.579452  0.593012  0.143459
199        0.213738  0.208887  0.324596  0.300950  0.532261  0.128549
201        0.019206  0.524732 -0.086166  0.072914 -0.070383  0.110531
203        0.304076  0.282060 -0.138148  0.057085  0.406311  0.039563
210        0.569730  0.766554 -0.183315  0.373580  0.172489  0.321661
217        0.334087  0.382774 -0.014094  0.326146  0.092255  0.311429
226        0.035908  0.726257  0.049454  0.127159 -0.223535  0.303700
237        0.005885 -0.250022 -0.345708  0.311110  0.175916 -0.406137
238        0.245825 -0.259899 -0.436141 -0.216723 -0.231521 -0.209974
261        0.354522  0.225313  0.263819 -0.003996  0.645316  0.251837

STORE_NBR       88        95        123       125   …       181       199   \
```

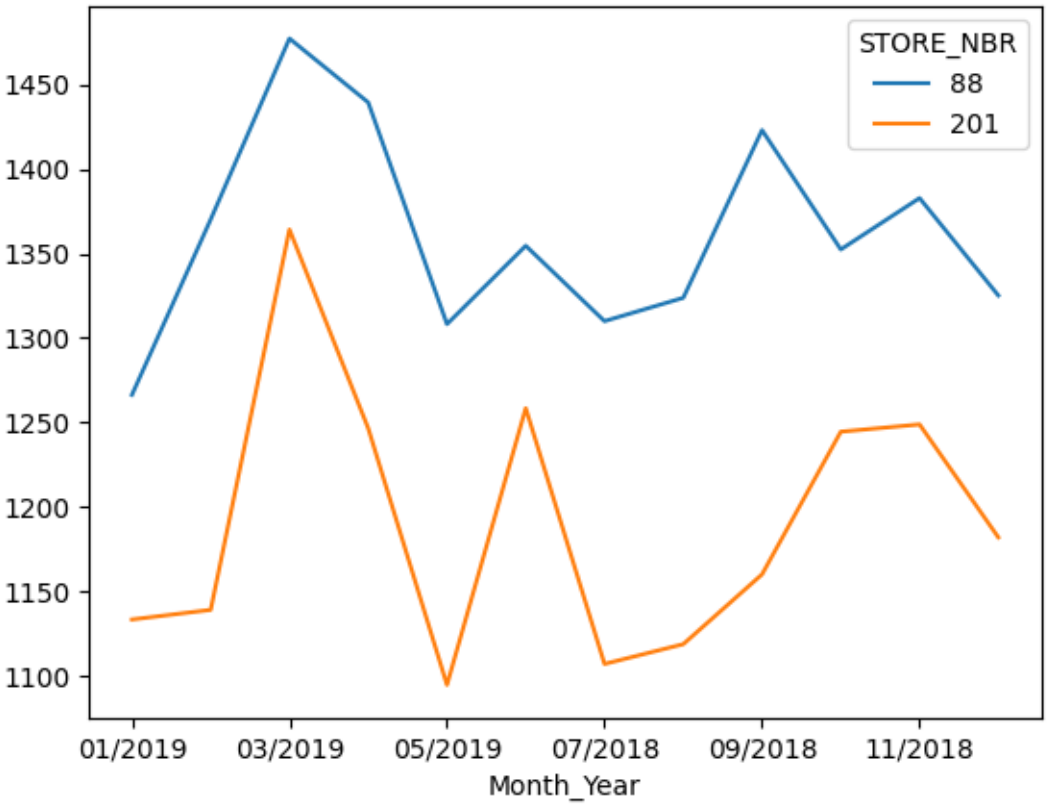
STORE_NBR					…		
4	-0.319831	-0.038154	-0.056994	0.312848	…	0.446083	0.213738
26	0.439615	-0.133565	-0.299583	0.609910	…	0.463871	0.208887
40	-0.141734	-0.211049	-0.431312	-0.468838	…	0.150957	0.324596
58	0.224899	-0.356401	0.094355	0.113932	…	0.579452	0.300950
72	-0.348883	0.017109	0.083195	0.033888	…	0.593012	0.532261
81	0.197483	-0.291601	-0.513938	0.187666	…	0.143459	0.128549
88	1.000000	0.364865	0.276588	0.360838	…	0.026537	-0.071859
95	0.364865	1.000000	0.377951	0.343644	…	-0.127888	0.087424
123	0.276588	0.377951	1.000000	0.118875	…	-0.011415	0.085259
125	0.360838	0.343644	0.118875	1.000000	…	0.152041	0.088827
130	-0.366231	0.335076	-0.082874	-0.218384	…	0.450007	0.481370
165	-0.457783	-0.503583	-0.862925	-0.289279	…	0.103769	-0.119110
181	0.026537	-0.127888	-0.011415	0.152041	…	1.000000	0.652396
199	-0.071859	0.087424	0.085259	0.088827	…	0.652396	1.000000
201	0.737583	0.477539	0.086478	0.413662	…	0.370930	0.232501
203	-0.273307	-0.172814	0.207557	0.381610	…	0.501786	0.168292
210	0.223905	-0.046953	-0.427565	0.516898	…	0.536018	0.214427
217	-0.092024	-0.215666	-0.466050	0.281443	…	0.241396	-0.329460
226	0.235797	-0.016937	-0.389581	0.508336	…	0.478759	0.237973
237	-0.137177	-0.317383	0.669474	-0.166064	…	0.221592	0.010110
238	-0.404274	-0.115867	-0.179323	0.172456	…	-0.582435	-0.699914
261	-0.254241	0.223455	-0.154009	0.373689	…	0.477049	0.743443

STORE_NBR	201	203	210	217	226	237	\
STORE_NBR							
4	0.019206	0.304076	0.569730	0.334087	0.035908	0.005885	
26	0.524732	0.282060	0.766554	0.382774	0.726257	-0.250022	
40	-0.086166	-0.138148	-0.183315	-0.014094	0.049454	-0.345708	
58	0.072914	0.057085	0.373580	0.326146	0.127159	0.311110	
72	-0.070383	0.406311	0.172489	0.092255	-0.223535	0.175916	
81	0.110531	0.039563	0.321661	0.311429	0.303700	-0.406137	
88	0.737583	-0.273307	0.223905	-0.092024	0.235797	-0.137177	
95	0.477539	-0.172814	-0.046953	-0.215666	-0.016937	-0.317383	
123	0.086478	0.207557	-0.427565	-0.466050	-0.389581	0.669474	
125	0.413662	0.381610	0.516898	0.281443	0.508336	-0.166064	
130	0.068873	0.278049	-0.081858	-0.004984	0.199733	-0.091144	
165	-0.311931	-0.176479	0.390984	0.591086	0.276446	-0.384525	
181	0.370930	0.501786	0.536018	0.241396	0.478759	0.221592	
199	0.232501	0.168292	0.214427	-0.329460	0.237973	0.010110	
201	1.000000	0.000869	0.543068	-0.158368	0.512059	-0.360471	
203	0.000869	1.000000	0.106441	0.220628	0.314826	0.438624	
210	0.543068	0.106441	1.000000	0.429079	0.655314	-0.374835	
217	-0.158368	0.220628	0.429079	1.000000	0.389677	-0.092522	
226	0.512059	0.314826	0.655314	0.389677	1.000000	-0.326469	
237	-0.360471	0.438624	-0.374835	-0.092522	-0.326469	1.000000	
238	-0.554762	-0.003216	-0.081607	0.436287	-0.206571	-0.044638	

```
261        0.160783  0.389786  0.249429 -0.049961  0.261636 -0.270792
```

```
STORE_NBR       238       261
STORE_NBR
4           0.245825  0.354522
26         -0.259899  0.225313
40         -0.436141  0.263819
58         -0.216723 -0.003996
72         -0.231521  0.645316
81         -0.209974  0.251837
88         -0.404274 -0.254241
95         -0.115867  0.223455
123        -0.179323 -0.154009
125         0.172456  0.373689
130        -0.399274  0.529394
165         0.280339 -0.055367
181        -0.582435  0.477049
199        -0.699914  0.743443
201        -0.554762  0.160783
203        -0.003216  0.389786
210        -0.081607  0.249429
217         0.436287 -0.049961
226        -0.206571  0.261636
237        -0.044638 -0.270792
238         1.000000 -0.299184
261        -0.299184  1.000000

[22 rows x 22 columns]
```

[27]:
```
Known2 = piv3[[88,201]]
Known2.plot()
plt.show()
```



[ ]:

[29]:
```
trail_77 = Data[Data['STORE_NBR'] == 77]
control_41 = Data[Data['STORE_NBR'] == 41]

trail_86 = Data[Data['STORE_NBR'] == 86]
control_109 = Data[Data['STORE_NBR'] == 109]

trail_88 = Data[Data['STORE_NBR'] == 88]
control_201 = Data[Data['STORE_NBR'] == 201]
```

[33]:
```
trail_77[['TOT_SALES','PROD_QTY']].sum()
```

[33]:
```
TOT_SALES    3040.0
PROD_QTY      872.0
dtype: float64
```

[35]:
```
control_41[['TOT_SALES','PROD_QTY']].sum()
```

[35]:
```
TOT_SALES    2570.2
PROD_QTY      723.0
```

```
dtype: float64
```

[36]:
```
trail_77[['LYLTY_CARD_NBR']].value_counts()
```

[36]:
```
LYLTY_CARD_NBR
77476             5
77109             4
77205             4
77066             4
77093             4
                 ..
77023             1
77024             1
77025             1
77187             1
77003             1
Name: count, Length: 356, dtype: int64
```

[37]:
```
control_41[['LYLTY_CARD_NBR']].value_counts()
```

[37]:
```
LYLTY_CARD_NBR
41497             4
41453             4
41466             4
41367             4
41359             4
                 ..
41471             1
41499             1
41002             1
41001             1
41505             1
Name: count, Length: 344, dtype: int64
```

[39]:
```
control_41[['LYLTY_CARD_NBR']].count()
```

[39]:
```
LYLTY_CARD_NBR    567
dtype: int64
```

[40]:
```
trail_77[['LYLTY_CARD_NBR']].count()
```

[40]:
```
LYLTY_CARD_NBR    563
dtype: int64
```

[41]:
```
repeat_cust = control_41[['LYLTY_CARD_NBR']].value_counts()
repeat_cust.head(15)
```

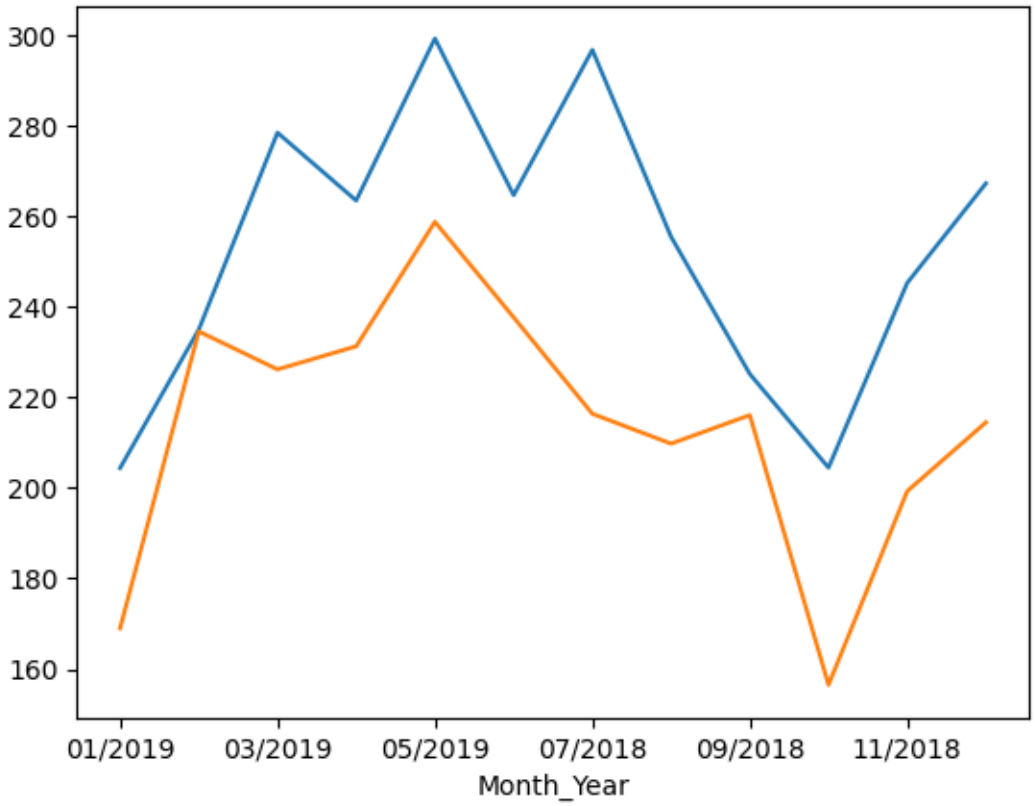
[41]:
```
LYLTY_CARD_NBR
41497             4
41453             4
41466             4
41367             4
41359             4
41368             4
41418             4
41423             4
41432             4
41429             4
41430             4
41474             4
41247             4
41254             4
41182             4
Name: count, dtype: int64
```

[42]:
```
repeated_cust = trail_77[['LYLTY_CARD_NBR']].value_counts()
repeat_cust.head(15)
```

[42]:
```
LYLTY_CARD_NBR
41497             4
41453             4
41466             4
41367             4
41359             4
41368             4
41418             4
41423             4
41432             4
41429             4
41430             4
41474             4
41247             4
41254             4
41182             4
Name: count, dtype: int64
```

[43]:
```
group77 = trail_77.groupby('Month_Year')
group41 = control_41.groupby('Month_Year')
```

[45]:
```
group77['TOT_SALES'].sum().plot(label = " trail_77")
group41['TOT_SALES'].sum().plot(label = " control_77")
plt.show()
```



[46]:
```
trail_86[['TOT_SALES','PROD_QTY']].sum()
```

[46]:
```
TOT_SALES    10635.35
PROD_QTY      3066.00
dtype: float64
```

[47]:
```
control_109[['TOT_SALES','PROD_QTY']].sum()
```

[47]:
```
TOT_SALES    10399.1
PROD_QTY      2977.0
dtype: float64
```

[59]:
```
trail_86[['LYLTY_CARD_NBR']].value_counts()
```

[59]:
```
LYLTY_CARD_NBR
86133             13
86112             13
86151             12
86075             12
86008             12
                  ..
```

```
155000              1
155003              1
155004              1
155005              1
155510              1
Name: count, Length: 273, dtype: int64
```

[60]:
```
control_109[['LYLTY_CARD_NBR']].value_counts()
```

[60]:
```
LYLTY_CARD_NBR
109036            16
109080            14
109086            13
109078            12
109212            12
                  ..
109121             1
109017             1
109200             1
109214             1
109222             1
Name: count, Length: 261, dtype: int64
```

[61]:
```
trail_86[['LYLTY_CARD_NBR']].count()
```

[61]:
```
LYLTY_CARD_NBR    1538
dtype: int64
```

[62]:
```
control_109[['LYLTY_CARD_NBR']].count()
```

[62]:
```
LYLTY_CARD_NBR    1505
dtype: int64
```

[52]:
```
repeat_cust = trail_86[['LYLTY_CARD_NBR']].value_counts()
repeat_cust.head(15)
```

[52]:
```
LYLTY_CARD_NBR
86133             13
86112             13
86151             12
86075             12
86008             12
86129             11
86116             11
86223             11
86250             11
86019             11
86027             11
```

```
86177              10
86193              10
86230              10
86238              10
Name: count, dtype: int64
```

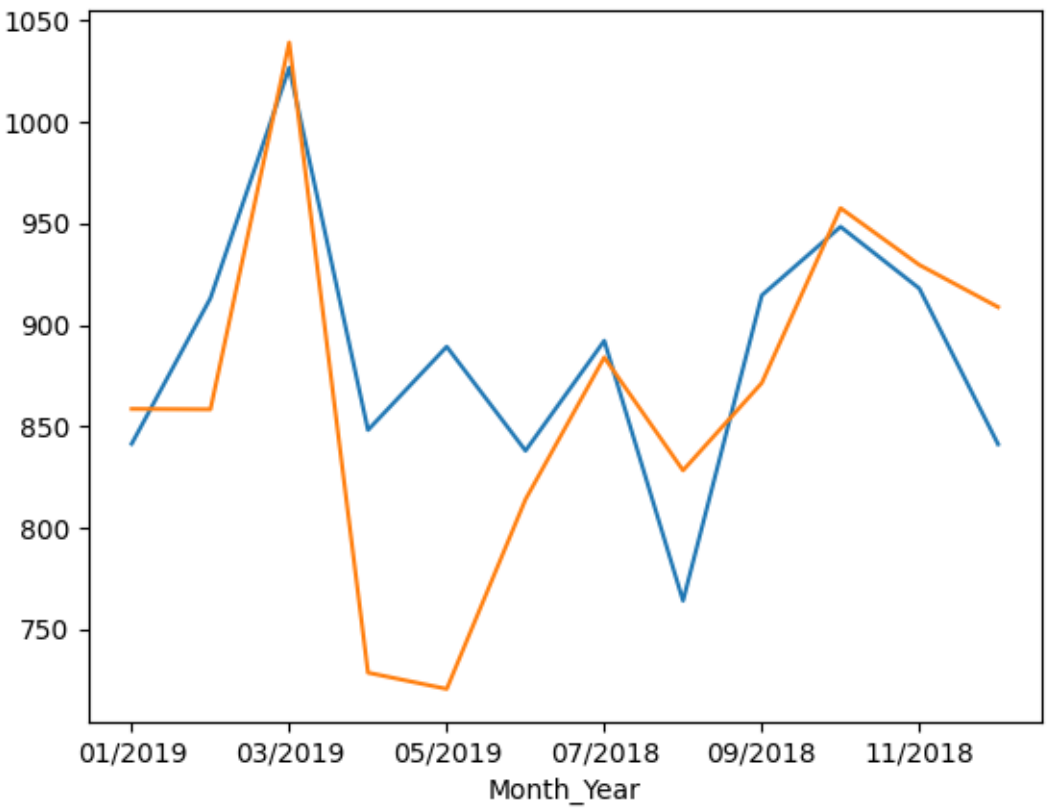
[53]:
```
repeat_cust = control_109[['LYLTY_CARD_NBR']].value_counts()
repeat_cust.head(15)
```

[53]:
```
LYLTY_CARD_NBR
109036             16
109080             14
109086             13
109078             12
109212             12
109094             12
109255             12
109067             11
109015             11
109100             11
109207             11
109246             10
109142             10
109133             10
109204             10
Name: count, dtype: int64
```

[55]:
```
group86 = trail_86.groupby('Month_Year')
group109 = control_109.groupby('Month_Year')
```

[56]:
```
group86['TOT_SALES'].sum().plot(label = "trail_86" )
group109['TOT_SALES'].sum().plot(label = "control_109" )
```

[56]:
```
<Axes: xlabel='Month_Year'>
```



[57]:
```
trail_88[['TOT_SALES','PROD_QTY']].sum()
```

[57]:
```
TOT_SALES    16333.25
PROD_QTY      3718.00
dtype: float64
```

[58]:
```
control_201[['TOT_SALES','PROD_QTY']].sum()
```

[58]:
```
TOT_SALES    14298.7
PROD_QTY      3262.0
dtype: float64
```

[64]:
```
trail_88[['LYLTY_CARD_NBR']].value_counts()
```

[64]:
```
LYLTY_CARD_NBR
88105             13
88247             11
88358             11
88351             10
88348             10
                  ..
```

```
88355              1
88372              1
2370701            1
2370751            1
2373711            1
Name: count, Length: 388, dtype: int64
```

[65]:
```
control_201[['LYLTY_CARD_NBR']].value_counts()
```

[65]:
```
LYLTY_CARD_NBR
201294            13
201120            11
201186            11
201206            10
201018            10
                  ..
201057             1
201037             1
201043             1
201356             1
201005             1
Name: count, Length: 376, dtype: int64
```

[66]:
```
trail_88[['LYLTY_CARD_NBR']].count()
```

[66]:
```
LYLTY_CARD_NBR    1873
dtype: int64
```

[67]:
```
control_201[['LYLTY_CARD_NBR']].count()
```

[67]:
```
LYLTY_CARD_NBR    1654
dtype: int64
```

[68]:
```
repeated_cust = trail_88[['LYLTY_CARD_NBR']].value_counts()
repeat_cust.head(15)
```

[68]:
```
LYLTY_CARD_NBR
109036            16
109080            14
109086            13
109078            12
109212            12
109094            12
109255            12
109067            11
109015            11
109100            11
109207            11
```

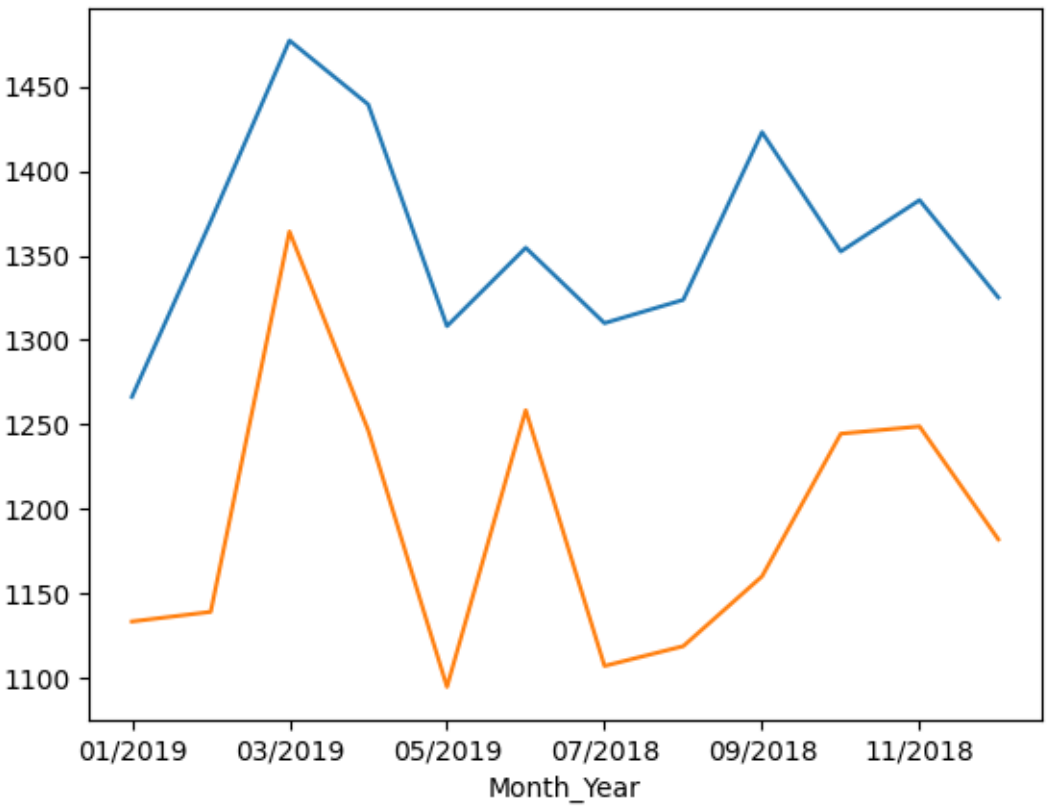
```
109246            10
109142            10
109133            10
109204            10
Name: count, dtype: int64
```

[69]:
```
repeated_cust = control_201[['LYLTY_CARD_NBR']].value_counts()
repeat_cust.head(15)
```

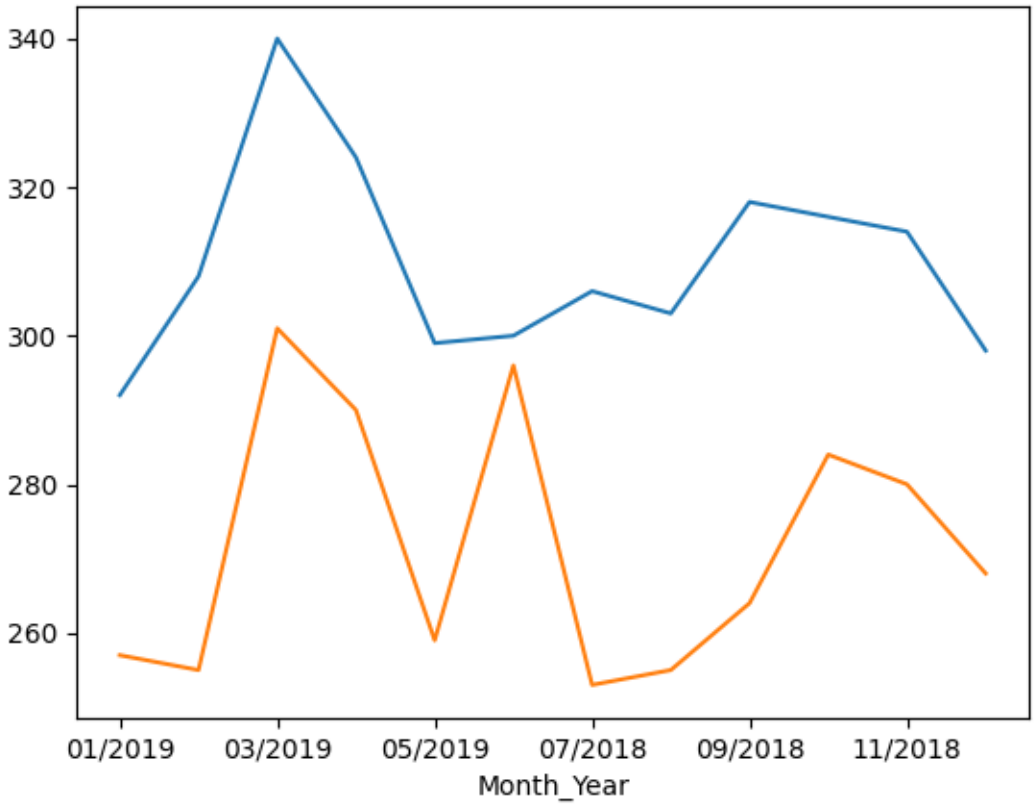
[69]:
```
LYLTY_CARD_NBR
109036            16
109080            14
109086            13
109078            12
109212            12
109094            12
109255            12
109067            11
109015            11
109100            11
109207            11
109246            10
109142            10
109133            10
109204            10
Name: count, dtype: int64
```

[71]:
```
group88 = trail_88.groupby('Month_Year')
group201 = control_201.groupby('Month_Year')
```

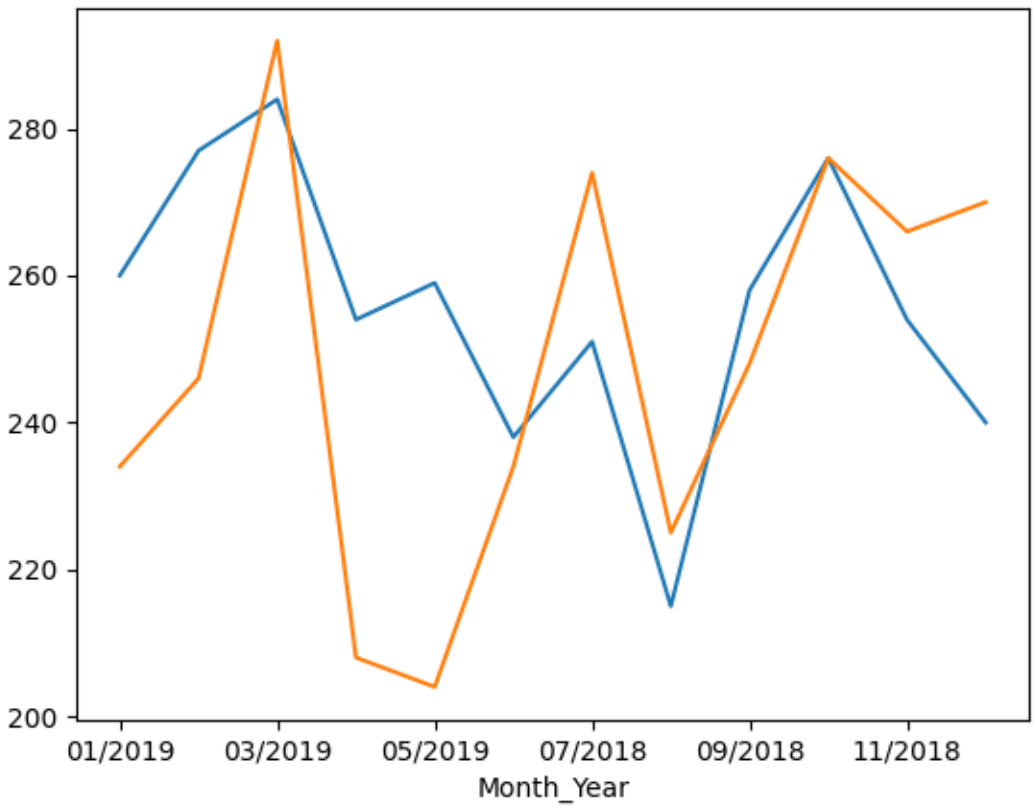
[73]:
```
group88['TOT_SALES'].sum().plot(label= "trail_88")
group201['TOT_SALES'].sum().plot(label= "control_201")
plt.show()
```



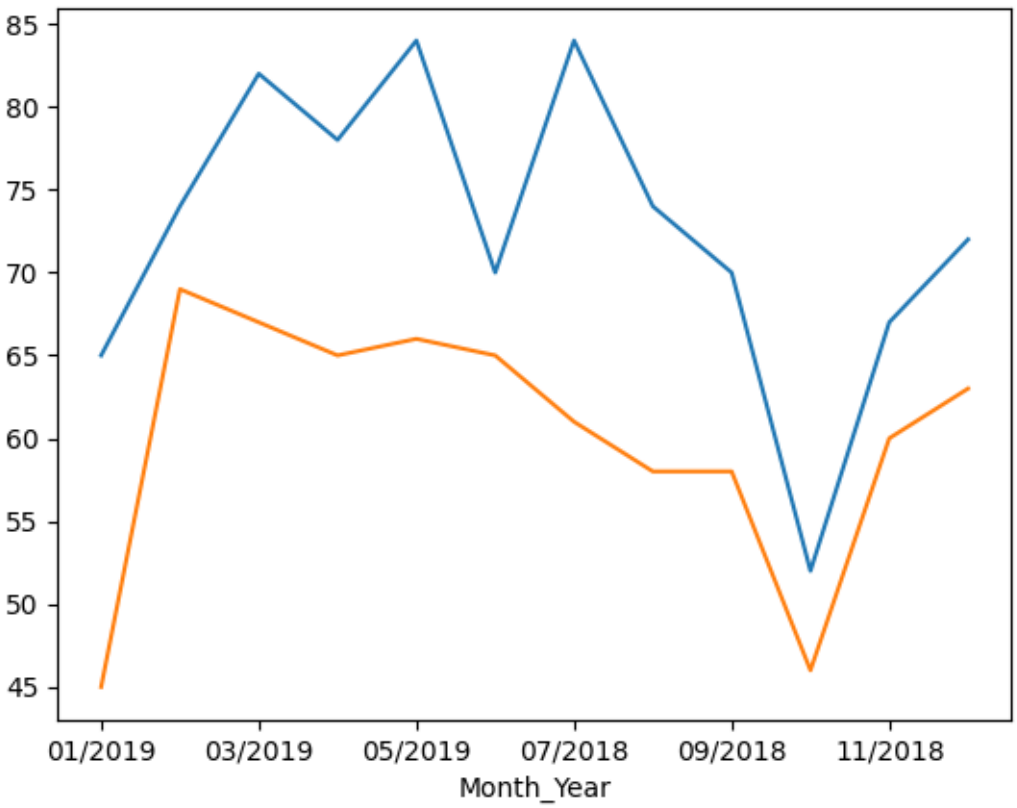
[74]:
```
group88['PROD_QTY'].sum().plot(label= "trail_88")
group201['PROD_QTY'].sum().plot(label= "control_201")
plt.show()
```



[75]:
```
group86['PROD_QTY'].sum().plot(label= "trail_86")
group109['PROD_QTY'].sum().plot(label= "control_109")
plt.show()
```



[76]:
```
group77['PROD_QTY'].sum().plot(label= "trail_77")
group41['PROD_QTY'].sum().plot(label= "control_41")
plt.show()
```



[78]:
```
group41['LYLTY_CARD_NBR'].value_counts().mean()
```

[78]: np.float64(1.05)

[79]:
```
group77['LYLTY_CARD_NBR'].value_counts().mean()
```

[79]: np.float64(1.048417132216015)

[80]:
```
group86['LYLTY_CARD_NBR'].value_counts().mean()
```

[80]: np.float64(1.2544861337683524)

[81]:
```
group109['LYLTY_CARD_NBR'].value_counts().mean()
```

[81]: np.float64(1.2918454935622317)

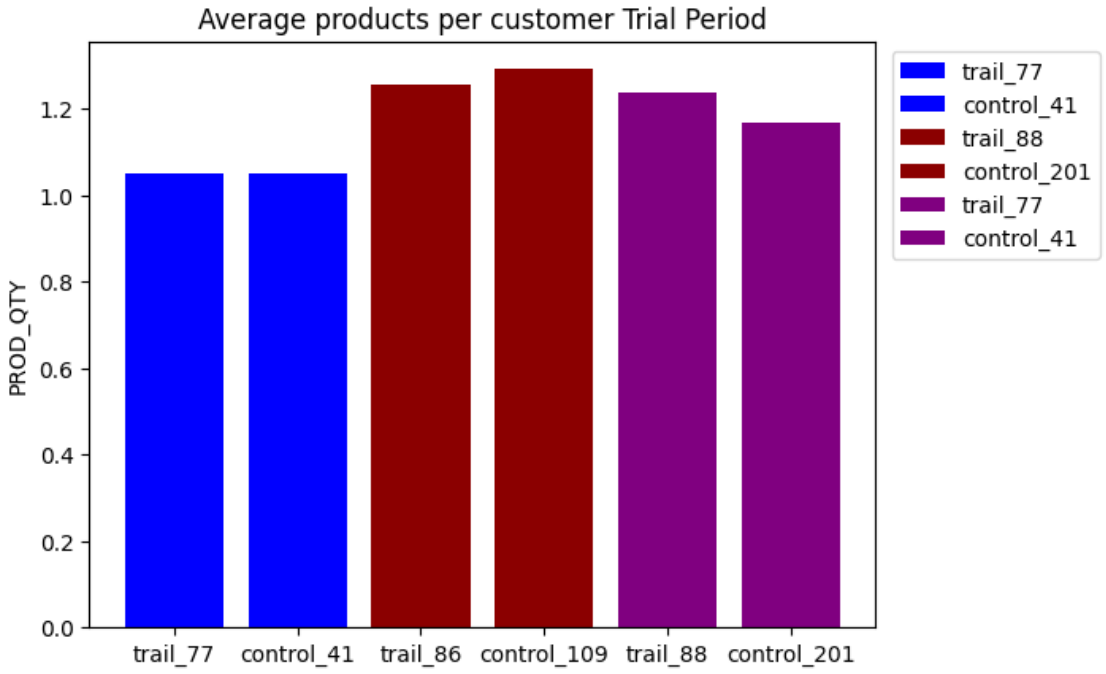
[82]:
```
group88['LYLTY_CARD_NBR'].value_counts().mean()
```

[82]: np.float64(1.2363036303630364)

[83]:
```
group201['LYLTY_CARD_NBR'].value_counts().mean()
```

[83]:
```
np.float64(1.1689045936395759)
```

[88]:
```
group1 = ["trail_77","control_41"]
group2 = ["trail_86","control_109"]
group3 = ["trail_88","control_201"]

values_grp_1 = [1.048417132216015, 1.05]
values_grp_2 = [1.2544861337683524, 1.2918454935622317]
values_grp_3 = [1.2363036303630364, 1.1689045936395759]
plt.bar(group1, values_grp_1, label = group1, color = "blue")
plt.bar(group2, values_grp_2, label = group3, color = "darkred")
plt.bar(group3, values_grp_3, label = group1, color = "purple")
plt.ylabel("PROD_QTY")
plt.legend(loc= "upper right", bbox_to_anchor= (1.3, 1))
plt.title("Average products per customer Trial Period")
plt.show()
```



Based on the data, the average number of transactions was slightly higher in one of the three trial stores.

This suggests that the new store layout may be positively impacting performance. Indicators such as total sales, number of products sold, repeat customer rate, and average transactions per customer all show that the trial stores are outperforming the control stores.

Recommendation: Expand the trial to include additional stores and conduct a follow-up analysis after three months. This will help determine whether the increase in sales is consistent and

sustainable over time.

[ ]: