```
#imports
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline


from plotly.offline import init_notebook_mode, iplot
init_notebook_mode(connected=True)
import plotly.offline as offline
offline.init_notebook_mode()
import cufflinks as cf
cf.go_offline()
```

```
#reading data
data=pd.read_csv("/content/drive/MyDrive/Colab Notebooks/QVI_data.csv");
data.head(2)
```

	LYLTY_CARD_NBR	DATE	STORE_NBR	TXN_ID	PROD_NBR	PROD_NAME	PROD_QTY	TOT_SALES	PACK_SIZE	BRAND	LIFESTAGE	PREI
0	1000	2018-10-17	1	1	5	Natural Chip Compny SeaSalt175g	2	6.0	175	NATURAL	YOUNG SINGLES/COUPLES	

```
data['DATE']=pd.to_datetime(data['DATE'])
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 264834 entries, 0 to 264833
Data columns (total 12 columns):
 #   Column            Non-Null Count   Dtype
---  ------            --------------   -----
 0   LYLTY_CARD_NBR    264834 non-null  int64
 1   DATE              264834 non-null  datetime64[ns]
 2   STORE_NBR         264834 non-null  int64
 3   TXN_ID            264834 non-null  int64
 4   PROD_NBR          264834 non-null  int64
 5   PROD_NAME         264834 non-null  object
 6   PROD_QTY          264834 non-null  int64
 7   TOT_SALES         264834 non-null  float64
 8   PACK_SIZE         264834 non-null  int64
 9   BRAND             264834 non-null  object
 10  LIFESTAGE         264834 non-null  object
 11  PREMIUM_CUSTOMER  264834 non-null  object
dtypes: datetime64[ns](1), float64(1), int64(6), object(4)
memory usage: 24.2+ MB
```

```
data['YEARMONTH']=[s.year*100+s.month for s in data['DATE']]
```

```
data
```

LYLTY_CARD_NBR	DATE	STORE_NBR	TXN_ID	PROD_NBR	PROD_NAME	PROD_QTY	TOT_SALES	PACK_SIZE	BRAND	L

METRICS UNDER CONSIDERATION:

- Monthly overall sales revenue
- Monthly number of customers
- Monthly number of transactions per customer

```
metrics=data.groupby(['STORE_NBR','YEARMONTH']).agg({'TOT_SALES':'sum','LYLTY_CARD_NBR':'nunique','TXN_ID':'nunique','PI
metrics['PRICE_PER_UNIT']=metrics['TOT_SALES']/metrics['PROD_QTY']
metrics['CHIP_PER_TXN']=metrics['PROD_QTY']/metrics['TXN_ID']
metrics=metrics.rename(columns={'LYLTY_CARD_NBR':'CUSTOMERS'})
metrics['TXN_PER_CUST']=metrics['TXN_ID']/metrics['CUSTOMERS']
metrics.drop(['TXN_ID'],axis=1,inplace=True)
```

WW Original

```
full=metrics.copy()
```

```
#taking data before 2019-02 into consideration
trial=[]
for i in metrics.index:
    if(i[1]>=201902):
        if(i[1]<=201904):
            trial.append(metrics.loc[i])
        metrics.drop(i,inplace=True)
trial=pd.DataFrame(trial)
```

```
#taking data after 2019-02 into trial dataframe
trial.index.name=('IDX')
k=0
trial['STORE_NBR']=0
trial['MONTHYEAR']=0
for (i,j) in trial.reset_index()['IDX']:
    trial['STORE_NBR'].iloc[k]=i
    trial['MONTHYEAR'][k]=j
    k=k+1
trial=trial.set_index(['STORE_NBR','MONTHYEAR'])
```

```
metrics
```

STORE_NBR	YEARMONTH	TOT_SALES	CUSTOMERS	PROD_QTY	PRICE_PER_UNIT	CHIP_PER_TXN	T
1	**201807**	206.9	49	62	3.337097	1.192308	
	201808	176.1	42	54	3.261111	1.255814	
	201809	278.8	59	75	3.717333	1.209677	
	201810	188.1	44	58	3.243103	1.288889	
	201811	192.6	46	57	3.378947	1.212766	
...	**...**	...	...	...	...	...	
272	**201809**	304.7	32	71	4.291549	1.972222	
	201810	430.6	44	99	4.349495	1.980000	
	201811	376.2	41	87	4.324138	1.933333	
	201812	403.9	47	89	4.538202	1.893617	
	201901	423.0	46	96	4.406250	1.920000	

1848 rows × 6 columns

Funtions to find correlation and magnitude of any store with another store

```
def calcCorr(store):
    '''
    input=store number which is to be compared
    output=dataframe with corelation coefficient values
    '''
    a=[]
    metrix=metrics[['TOT_SALES','CUSTOMERS']]#add metrics as required e.g. ,'TXN_PER_CUST'
    for i in metrix.index:
```

```
    for i in metrix.index:
        a.append(metrix.loc[store].corrwith(metrix.loc[i[0]]))
    df= pd.DataFrame(a)
    df.index=metrix.index
    df=df.drop_duplicates()
    df.index=[s[0] for s in df.index]
    df.index.name="STORE_NBR"
    return df


def standardizer(df):
    '''
    input=dataframe with metrics
    output=dataframe with mean of the metrics in a new column
    '''
    df=df.abs()
    df['MAGNITUDE']=df.mean(axis=1)
    return df
```

Store 77

Finding stores corelated to store 77

```
corr77=calcCorr(77)
corr77.head(3)
```

	TOT_SALES	CUSTOMERS
STORE_NBR		
1	0.075218	0.322168
2	-0.263079	-0.572051
3	0.806644	0.834207

```
corr77=standardizer(corr77)
corr77
```

	TOT_SALES	CUSTOMERS	MAGNITUDE
STORE_NBR			
1	0.075218	0.322168	0.198693
2	0.263079	0.572051	0.417565
3	0.806644	0.834207	0.820426
4	0.263300	0.295639	0.279469
5	0.110652	0.370659	0.240655
...	...	...	...
268	0.344757	0.369517	0.357137
269	0.315730	0.474293	0.395011
270	0.315430	0.131259	0.223345
271	0.355487	0.019629	0.187558
272	0.117622	0.223217	0.170420

266 rows × 3 columns

```
corr77=corr77.sort_values(['MAGNITUDE'],ascending=False).dropna()
corr77
```

STORE_NBR	TOT_SALES	CUSTOMERS	MAGNITUDE
77	1.000000	1.000000	1.000000
233	0.903774	0.990358	0.947066
119	0.867664	0.983267	0.925466
71	0.914106	0.754817	0.834461
3	0.806644	0.834207	0.820426

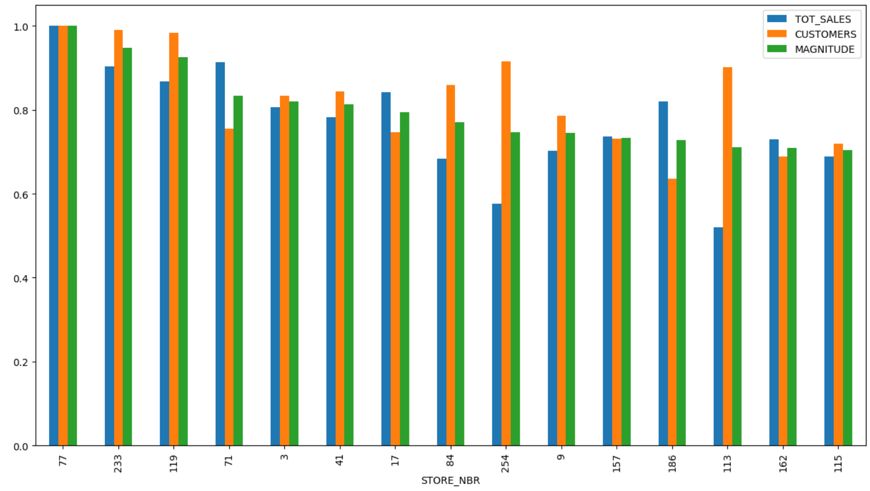
shows that stores 233,119,71 are the most correlated to store 77

Selecting 233 as control store as it has max correlation

Visualizing ...

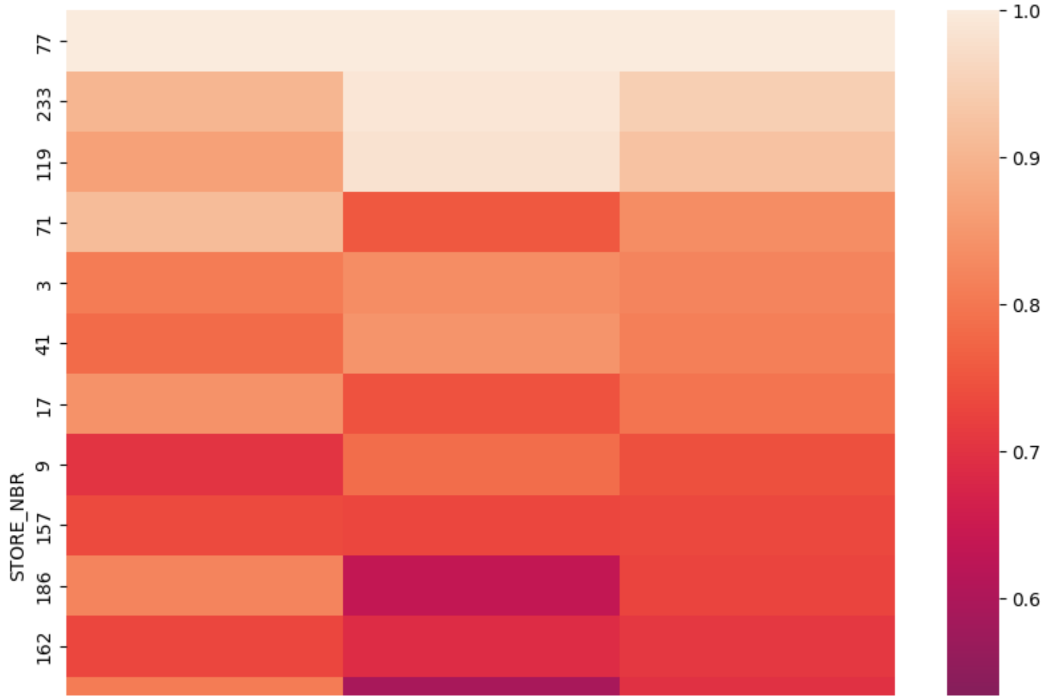
```
#Taking 0.7 as threshold corelation
corr77[(corr77.MAGNITUDE.abs()>0.7)].plot(kind='bar',figsize=(15,8))
```

<Axes: xlabel='STORE_NBR'>




```
plt.figure(figsize=(10,10))
sns.heatmap(corr77[corr77.TOT_SALES.abs()>0.7])
```

```
<Axes: ylabel='STORE_NBR'>
```



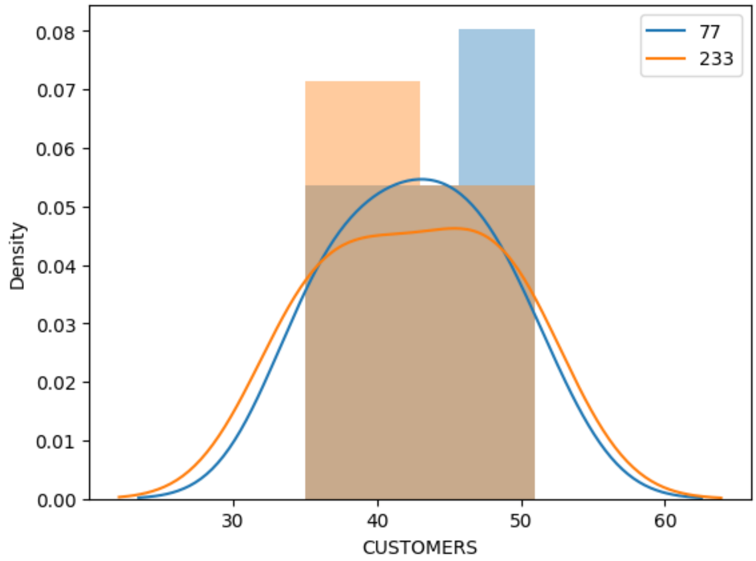
Taking the store 233 into consideration plotting different measure against those of store 77

```
sns.distplot(metrics.loc[77]['TOT_SALES'])
sns.distplot(metrics.loc[233]['TOT_SALES'])
plt.legend(labels=['77','233'])
```

```
<ipython-input-21-8d02d6235829>:1: UserWarning:


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

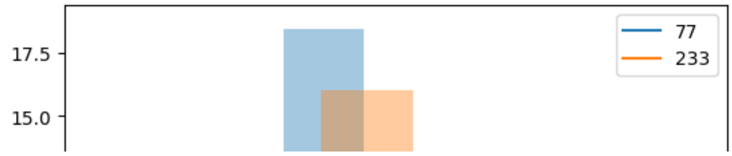
```
sns.distplot(metrics.loc[77]['CUSTOMERS'])
sns.distplot(metrics.loc[233]['CUSTOMERS'])
plt.legend(labels=['77','233'])
```

```
<ipython-input-22-de7c783a6076>:1: 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


<ipython-input-22-de7c783a6076>:2: 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


<matplotlib.legend.Legend at 0x7f56a8f58eb0>
```




```
sns.distplot(metrics.loc[77]['TXN_PER_CUST'])
sns.distplot(metrics.loc[233]['TXN_PER_CUST'])
plt.legend(labels=['77','233'])
```

```
<ipython-input-23-29a30bd1800b>:1: 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


<ipython-input-23-29a30bd1800b>:2: 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


<matplotlib.legend.Legend at 0x7f56a943d570>
```



Since distributions of store 233 are similar to that of store 77, selecting store 233 as control store with max similarities to store 77

Calculating difference between scaled control sales and trial sales

Let null hypothesis be that both stores 77 ans 233 have no difference

```
from scipy.stats import ks_2samp,ttest_ind,t
# difference between control and trial sales
a=[]
for x in metrics.columns:
    a.append(ks_2samp(metrics.loc[77][x], metrics.loc[233][x]))
a=pd.DataFrame(a,index=metrics.columns)
```

```
<ipython-input-24-03f79ccd5bd9>:5: RuntimeWarning:

ks_2samp: Exact calculation unsuccessful. Switching to method=asymp.
```

```
a
```

	statistic	pvalue
TOT_SALES	0.285714	0.962704
CUSTOMERS	0.142857	0.999961
PROD_QTY	0.285714	0.962704
PRICE_PER_UNIT	0.285714	0.962704
CHIP_PER_TXN	0.285714	0.962704
TXN_PER_CUST	0.428571	0.575175

For pre trial period, since all of the p-values are high (say more than 0.05), we can't reject the null hypothesis

Assessment of trial

The trial period goes from the start of February 2019 to April 2019. We now want to see if there has been an uplift in overall chip sales.

Sampling march and april from the 3 months

```
b=[]
for x in trial.columns:
    b.append(ttest_ind(trial.loc[77][x].tail(2), trial.loc[233][x].tail(2)))
b=pd.DataFrame(b,index=metrics.columns)
```

```
b
```

```
[Ttest_indResult(statistic=4.267335718552558, pvalue=0.05076881409465864),
 Ttest_indResult(statistic=2.5861309700971087, pvalue=0.12261789030076836),
 Ttest_indResult(statistic=4.043680421515942, pvalue=0.05606345536654277),
 Ttest_indResult(statistic=-0.6341732526845992, pvalue=0.5908283504129962),
 Ttest_indResult(statistic=1.7851264851986517, pvalue=0.21616543644909403),
 Ttest_indResult(statistic=0.33243393574435853, pvalue=0.7711708194767704),
 Ttest_indResult(statistic=nan, pvalue=nan),
 Ttest_indResult(statistic=nan, pvalue=nan)]
```

```
#critical value
t.ppf(0.95,df=7)
```

```
1.894578605061305
```

Since all of the p-values are high (say more than 0.05), we reject the null hypothesis i.e. there means are significantly different. We can observe that the t-value is much larger than the 95th percentile value of the t-distribution for March and April - i.e. the increase in sales in the trial store in March and April is statistically greater than in the control store.

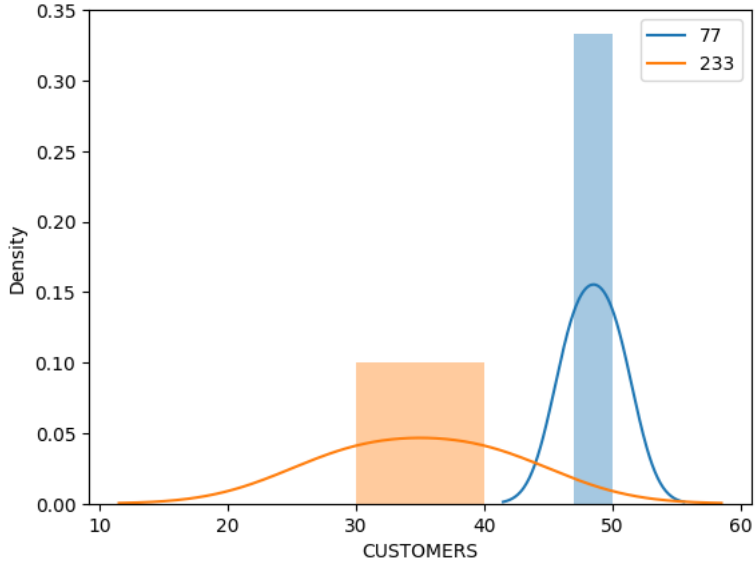
Vizualizing means

```
sns.distplot(trial.loc[77]['TOT_SALES'].tail(2))
sns.distplot(trial.loc[233]['TOT_SALES'].tail(2))
plt.legend(labels=['77','233'])
```

```
<ipython-input-29-536285c528f0>:1: UserWarning:


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

```
sns.distplot(trial.loc[77]['CUSTOMERS'].tail(2))
sns.distplot(trial.loc[233]['CUSTOMERS'].tail(2))
plt.legend(labels=['77','233'])
```

```
<ipython-input-30-2a7f1c6d83d9>:1: 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


<ipython-input-30-2a7f1c6d83d9>:2: 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


<matplotlib.legend.Legend at 0x7f56a615e290>
```



It can be visualized that the is a significant difference in the means, so trial store behavior(77) is different from control store (233).

The results show that the trial in store 77 is significantly different to its control store in the trial period as the trial store performance lies outside the 5% to 95% confidence interval of the control store in two of the three trial months.

Store 86

Repeating same process for trial store 86

```
corr86=calcCorr(86)
corr86.head(3)
```

	TOT_SALES	CUSTOMERS
STORE_NBR		
1	0.445632	0.485831

```
corr86=standardizer(corr86)
corr86
```

	TOT_SALES	CUSTOMERS	MAGNITUDE
STORE_NBR			
1	0.445632	0.485831	0.465731
2	0.403835	0.086161	0.244998
3	0.261284	0.353786	0.307535
4	0.039035	0.169608	0.104322
5	0.235159	0.253229	0.244194
...	...	...	...
268	0.452182	0.034273	0.243228
269	0.697055	0.098587	0.397821
270	0.730679	0.767267	0.748973
271	0.527637	0.267393	0.397515
272	0.004926	0.353815	0.179371

266 rows × 3 columns

```
corr86=corr86.sort_values(['MAGNITUDE'],ascending=False).dropna()
corr86
```

	TOT_SALES	CUSTOMERS	MAGNITUDE
STORE_NBR			
86	1.000000	1.000000	1.000000
155	0.877882	0.942876	0.910379
23	0.784698	0.943559	0.864128
120	0.872693	0.815097	0.843895
114	0.734415	0.855339	0.794877
...	...	...	...
91	0.019027	0.041271	0.030149
17	0.029793	0.030039	0.029916
131	0.028487	0.031142	0.029815
219	0.046653	0.004999	0.025826
234	0.010509	0.040306	0.025407

263 rows × 3 columns

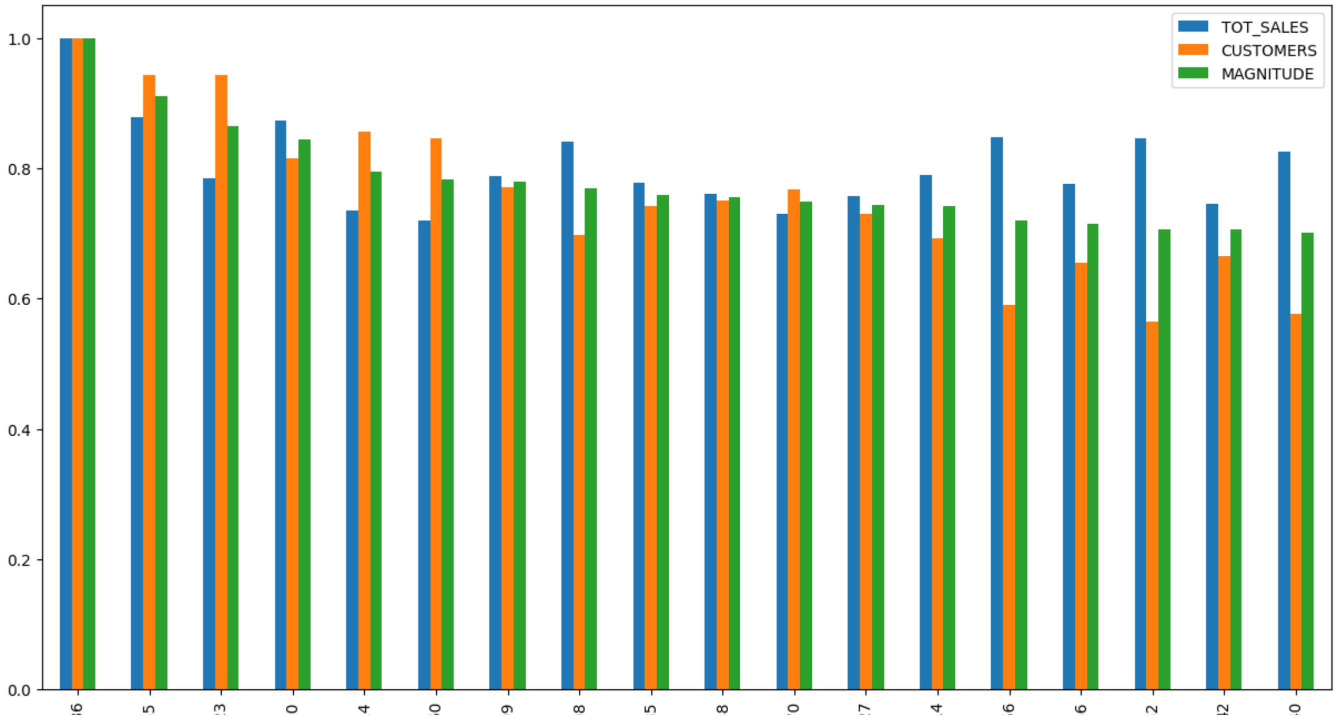
**shows that stores 155,23,120 are the most correlated to store 86

Selecting 155 as control store as it has max correlation

Visualising Means

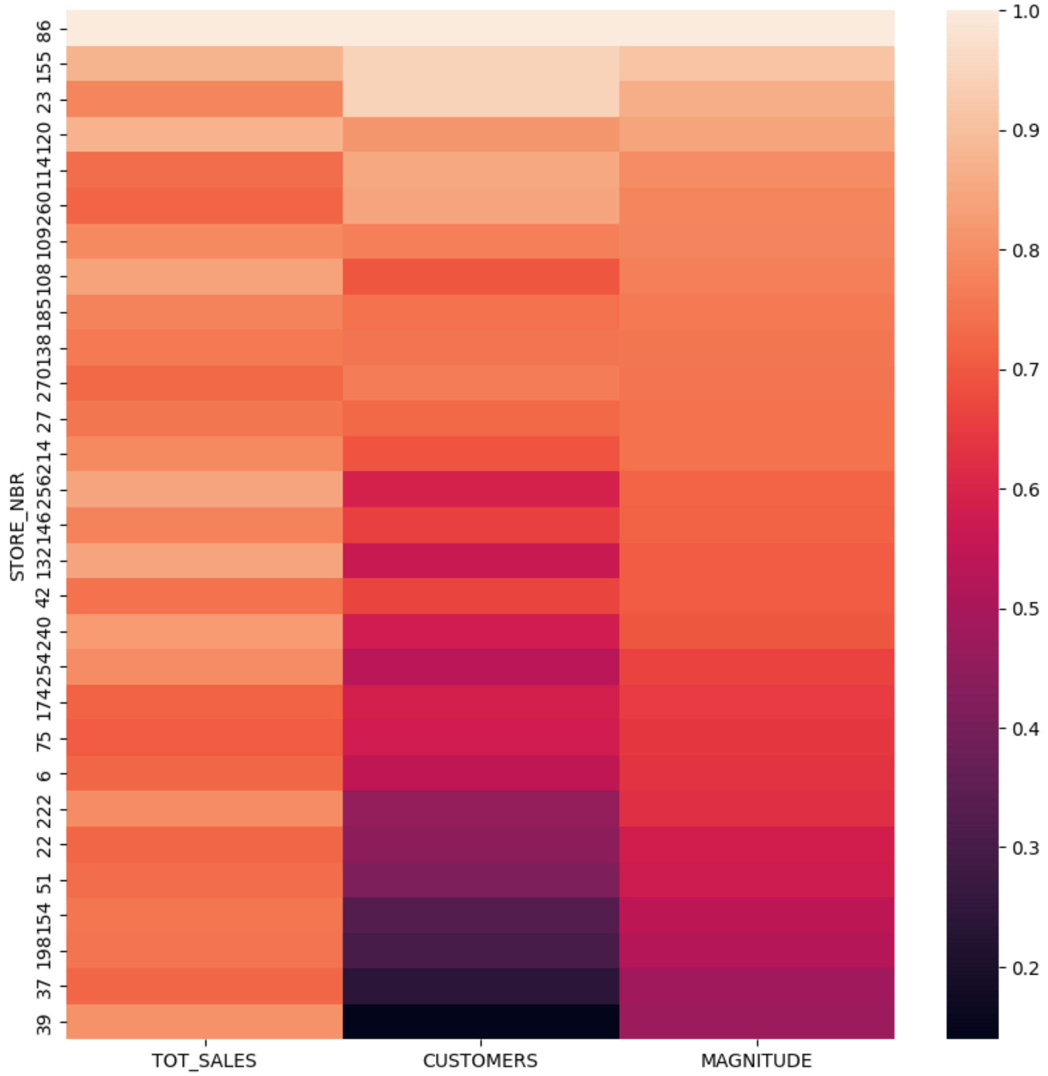
```
#Taking 0.7 as threshold corelation
corr86[(corr86.MAGNITUDE.abs()>0.7)].plot(kind='bar',figsize=(15,8))
```

```
<Axes: xlabel='STORE_NBR'>
```




```
plt.figure(figsize=(10,10))
sns.heatmap(corr86[corr86.TOT_SALES.abs()>0.7])
```

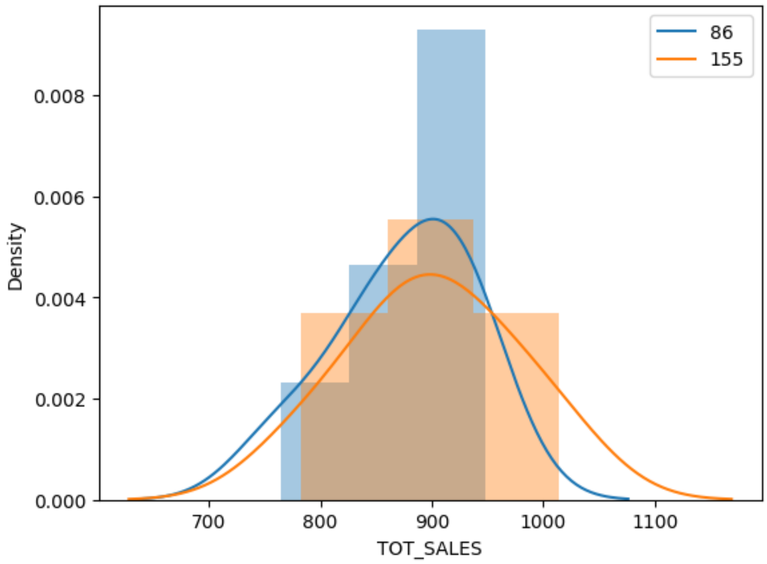
```
<Axes: ylabel='STORE_NBR'>
```



Taking the store 155 into consideration plotting different measure against those of store 86

```
sns.distplot(metrics.loc[86]['TOT_SALES'])
sns.distplot(metrics.loc[155]['TOT_SALES'])
plt.legend(labels=['86','155'])
```

```
<ipython-input-36-f50418f92e3b>:1: 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


<ipython-input-36-f50418f92e3b>:2: 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

<matplotlib.legend.Legend at 0x7f56a5f55120>
```

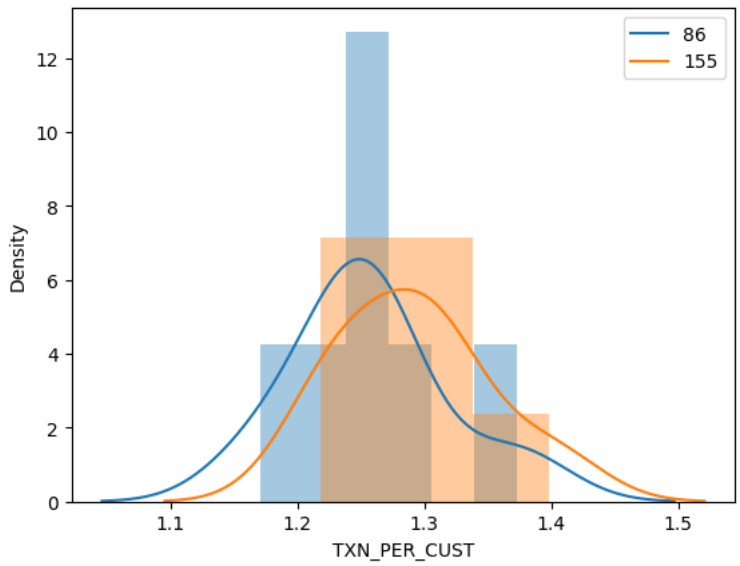



```
sns.distplot(metrics.loc[86]['CUSTOMERS'])
sns.distplot(metrics.loc[155]['CUSTOMERS'])
plt.legend(labels=['86','155'])
```

```
<ipython-input-37-df86b32c0a0e>:1: 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


<ipython-input-37-df86b32c0a0e>:2: 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

<matplotlib.legend.Legend at 0x7f57003dfcd0>
```

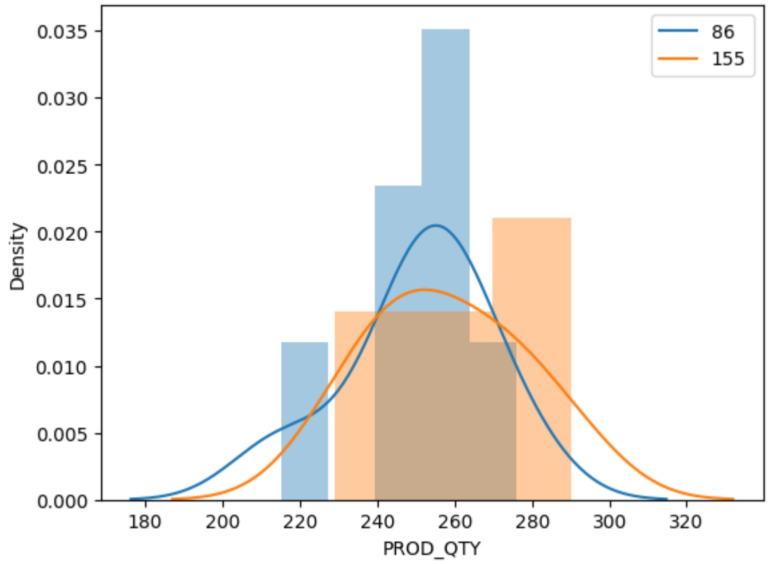

```
sns.distplot(metrics.loc[86]['TXN_PER_CUST'])
sns.distplot(metrics.loc[155]['TXN_PER_CUST'])
plt.legend(labels=['86','155'])
```

```
<ipython-input-38-a5243cb1b919>:1: 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


<ipython-input-38-a5243cb1b919>:2: 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

<matplotlib.legend.Legend at 0x7f56a5e1df30>
```



```
sns.distplot(metrics.loc[86]['PROD_QTY'])
sns.distplot(metrics.loc[155]['PROD_QTY'])
plt.legend(labels=['86','155'])
```

```
<ipython-input-39-43bc55766ab0>:1: 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


<ipython-input-39-43bc55766ab0>:2: 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


<matplotlib.legend.Legend at 0x7f56a5d53ee0>
```



Since distributions of store 155 are similar to that of store 86, selecting store 155 as control store with max similarities to store 86

Calculating difference between scaled control sales and trial sales

Let null hypothesis be that both stores 77 ans 233 have no difference

```
from scipy.stats import ks_2samp,ttest_ind,ttest_rel,t
# difference between control and trial sales
a=[]
for x in metrics.columns:
    a.append(ks_2samp(metrics.loc[86][x], metrics.loc[155][x]))
a=pd.DataFrame(a,index=metrics.columns)
```

```
a
```

	statistic	pvalue
TOT_SALES	0.285714	0.962704
CUSTOMERS	0.285714	0.962704
PROD_QTY	0.285714	0.962704
TXN_PER_CUST	0.428571	0.575175

For pre trial period, since p-values for TOT_SALES, CUSTOMERS and PROD_QTY are high (say more than 0.95), we can't reject the null hypothesis

Assessment of trial

The trial period goes from the start of February 2019 to April 2019. We now want to see if there has been an uplift in overall chip sales.

```
b=[]
for x in trial.columns:
    b.append(ttest_ind(trial.loc[86][x].tail(2), trial.loc[155][x].tail(2)))
b=pd.DataFrame(b,index=metrics.columns)
```

```
b
```

```
[Ttest_indResult(statistic=1.234511973459806, pvalue=0.3423782995617056),
 Ttest_indResult(statistic=2.414953415699773, pvalue=0.1370762576641993),
 Ttest_indResult(statistic=1.8625322946656702, pvalue=0.2035678264159959),
 Ttest_indResult(statistic=0.3662137805590835, pvalue=0.749316341306336),
 Ttest_indResult(statistic=-0.2859375734541281, pvalue=0.8018218058538022),
 Ttest_indResult(statistic=-1.074766842347899, pvalue=0.39492938851613935),
 Ttest_indResult(statistic=nan, pvalue=nan),
 Ttest_indResult(statistic=nan, pvalue=nan)]
```

```
#critical value
t.ppf(0.95,df=7)
```

```
1.894578605061305
```

Since all of the p-values are high (say more than 0.05), we reject the null hypothesis i.e. there means are significantly different.

We can observe that the t-value is much larger than the 95th percentile value of the t-distribution for March and April - i.e. the increase in sales in the trial store in March and April is statistically greater than in the control store.

The results show that the trial in store 88 is significantly different to its control store in the trial period as the trial store performance lies outside of the 5% to 95% confidence interval of the control store in two of the three trial months.

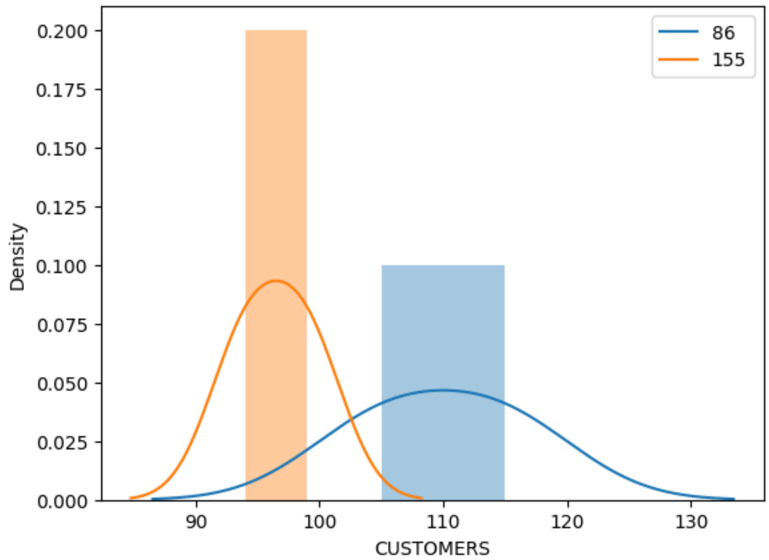
Vizualizing means

```
sns.distplot(trial.loc[86]['TOT_SALES'].tail(2))
sns.distplot(trial.loc[155]['TOT_SALES'].tail(2))
plt.legend(labels=['86','155'])
```

```
<ipython-input-45-993ae71e0d56>:1: 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


<ipython-input-45-993ae71e0d56>:2: 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(trial.loc[86]['CUSTOMERS'].tail(2))
sns.distplot(trial.loc[155]['CUSTOMERS'].tail(2))
plt.legend(labels=['86','155'])
```

```
<ipython-input-46-0505f93cce56>:1: 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


<ipython-input-46-0505f93cce56>:2: 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

<matplotlib.legend.Legend at 0x7f56a5c16a10>
```



It can be visualized that the is a significant difference in the means, so trial store behavior(86) is different from control store (155).

It looks like the number of customers is significantly higher in all of the three months. This seems to suggest that the trial had a significant impact on increasingthe number of customers in trial store 86 but as we saw, sales were not significantly higher. We should check with the Category Manager if there were special deals in the trial store that were may have resulted in lower prices, impacting the results.

Store 88

Finding stores corelated to store 88

```
corr88=calcCorr(88)
corr88.head(3)
```

STORE_NBR	TOT_SALES	CUSTOMERS
1	0.813636	0.305334
2	-0.067927	-0.452379
3	-0.507847	0.522884

```
corr88=standardizer(corr88)
corr88
```

STORE_NBR	TOT_SALES	CUSTOMERS	MAGNITUDE
1	0.813636	0.305334	0.559485
2	0.067927	0.452379	0.260153
3	0.507847	0.522884	0.515365
4	0.745566	0.361503	0.553534
5	0.190330	0.025320	0.107825
...	...	...	...
268	0.021429	0.672672	0.347050
269	0.172578	0.274781	0.223679
270	0.723272	0.103032	0.413152
271	0.103037	0.018831	0.060934
272	0.772772	0.026909	0.399841

265 rows × 3 columns

```
corr88=corr88.sort_values(['MAGNITUDE'],ascending=False).dropna()
corr88.head(15)
```

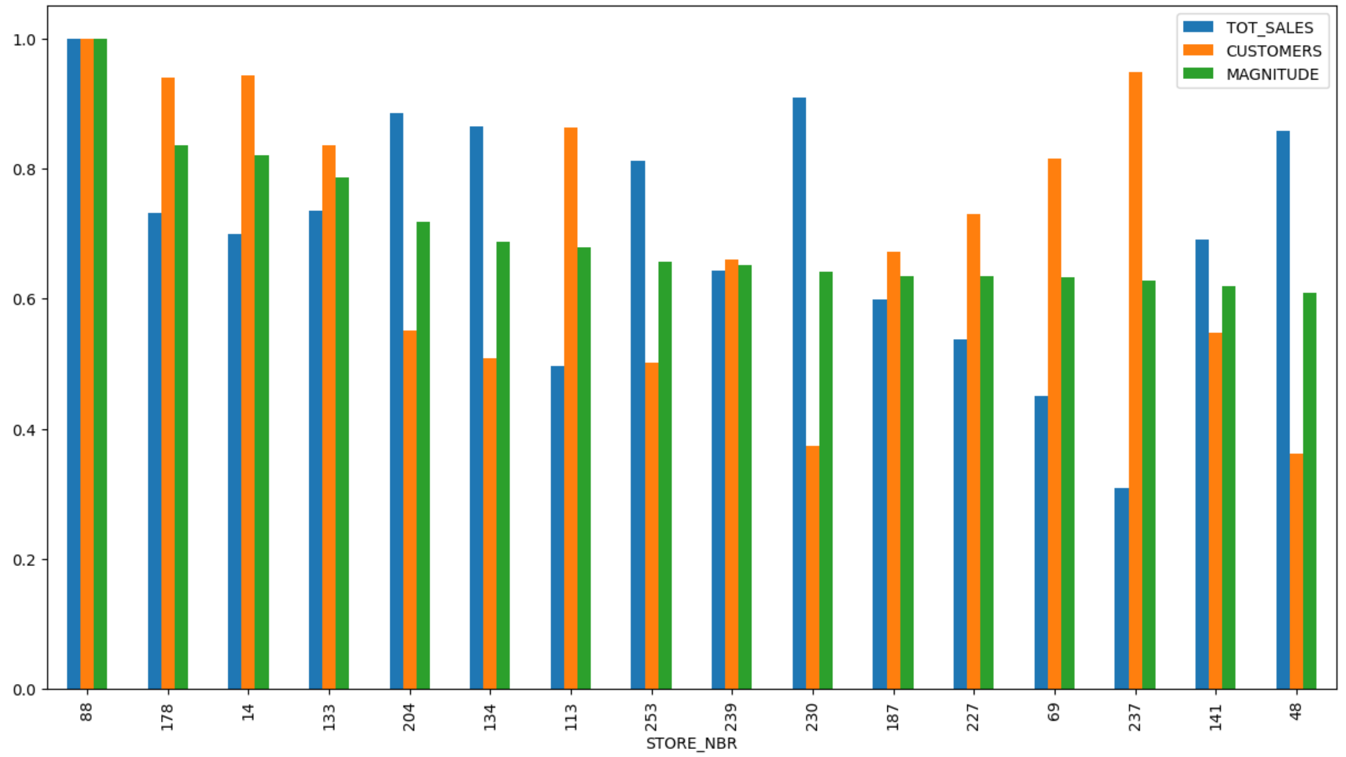
TOT_SALES CUSTOMERS MAGNITUDE

**shows that stores 178,14,133 are the most correlated to store 88

Visualizing ...

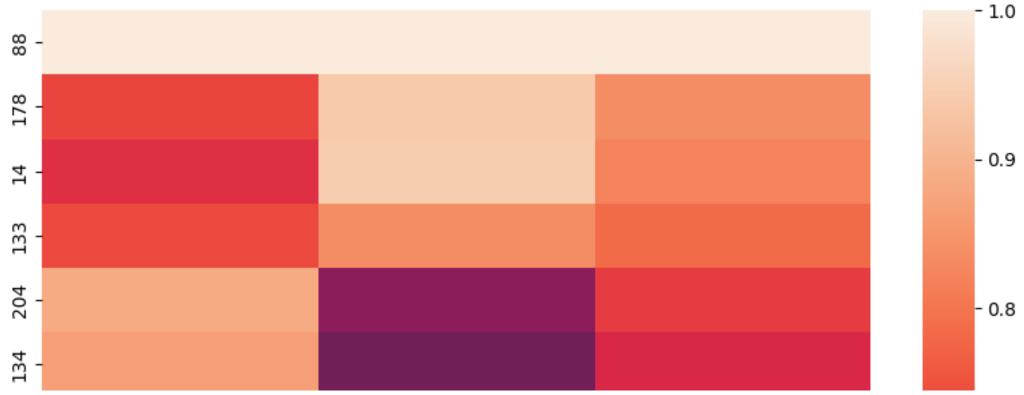
```
#Taking 0.6 as threshold corelation
corr88[(corr88.MAGNITUDE.abs()>0.6)].plot(kind='bar',figsize=(15,8))
```

```
<Axes: xlabel='STORE_NBR'>
```




```
plt.figure(figsize=(10,10))
sns.heatmap(corr88[corr88.MAGNITUDE.abs()>0.6])
```

```
<Axes: ylabel='STORE_NBR'>
```




```
plt.figure(figsize=(15,10))
for x in corr88[corr88.MAGNITUDE.abs()>0.6].index:
    sns.distplot(metrics.loc[88]['TOT_SALES'])
    sns.distplot(metrics.loc[x]['TOT_SALES'],label=x,hist=False)
plt.legend()
```

```
<ipython-input-52-2a32d86093c3>: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


<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

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


<ipython-input-52-2a32d86093c3>: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


<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

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


<ipython-input-52-2a32d86093c3>: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


<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

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


<ipython-input-52-2a32d86093c3>: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
```

```
<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

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


<ipython-input-52-2a32d86093c3>: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


<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

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


<ipython-input-52-2a32d86093c3>: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


<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

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


<ipython-input-52-2a32d86093c3>: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


<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

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

```
<ipython-input-52-2a32d86093c3>: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


<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

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


<ipython-input-52-2a32d86093c3>: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


<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

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


<ipython-input-52-2a32d86093c3>: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


<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

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


<ipython-input-52-2a32d86093c3>: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
```

https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

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

<ipython-input-52-2a32d86093c3>: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

<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

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

<ipython-input-52-2a32d86093c3>: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

<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

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

<ipython-input-52-2a32d86093c3>: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

<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

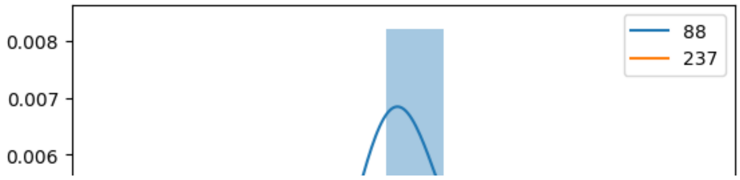
For a guide to updating your code to use the new functions, please see

```
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751


<ipython-input-52-2a32d86093c3>: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


<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

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


<ipython-input-52-2a32d86093c3>: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


<ipython-input-52-2a32d86093c3>:4: 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 `kdeplot` (an axes-level function for kernel density plots).

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


<matplotlib.legend.Legend at 0x7f56a56b1a50>
```

Therefore Taking the store 237 into consideration plotting different measure against those of store 88

```
sns.distplot(metrics.loc[88]['TOT_SALES'])
sns.distplot(metrics.loc[237]['TOT_SALES'])
plt.legend(labels=['88','237'])
```

```
<ipython-input-53-7994e8644ade>:1: 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


<ipython-input-53-7994e8644ade>:2: 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


<matplotlib.legend.Legend at 0x7f56a579c2b0>
```




```
sns.distplot(metrics.loc[88]['CUSTOMERS'])
sns.distplot(metrics.loc[237]['CUSTOMERS'])
plt.legend(labels=['88','237'])
```