

# bcg-feature-eng-task-4

September 9, 2023

## 1 BCG-FORAGE VIRTUAL INTENSHIP PROGRAM

## 2 TASK 4 : FEATURE ENGINEERING & PREDICTIVE MODELING

## 3 IMPORTING REQUIRED LIBRARIES

```
[1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
pd.set_option('display.max_columns',100)
from sklearn.preprocessing import FunctionTransformer
import scipy

print('done')
```

```
/opt/conda/lib/python3.10/site-packages/scipy/_init_.py:146: UserWarning: A
NumPy version >=1.16.5 and <1.23.0 is required for this version of SciPy
(detected version 1.23.5
    warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion}"
```

done

**PRICE ANALYSIS TERMINOLOGIES** \* Peak-> relatively highest demand \* Off peak->relatively low demand \* Mid peak-> lies between peak and off peak

```
[2]: df=pd.read_excel('/kaggle/input/bcg-fe/bcg_bfe.xlsx')
df.head()
```

```
[2]:   Unnamed: 0          id  \
0           0  24011ae4ebbe3035111d65fa7c15bc57
1           1  24011ae4ebbe3035111d65fa7c15bc57
2           2  24011ae4ebbe3035111d65fa7c15bc57
3           3  24011ae4ebbe3035111d65fa7c15bc57
4           4  24011ae4ebbe3035111d65fa7c15bc57

channel_sales  cons_12m  cons_gas_12m  cons_last_month  \
```

0	foosdfpkusacimwkcso <b>s</b> bicdxkicaua	0	54946	0					
1	foosdfpkusacimwkcso <b>s</b> bicdxkicaua	0	54946	0					
2	foosdfpkusacimwkcso <b>s</b> bicdxkicaua	0	54946	0					
3	foosdfpkusacimwkcso <b>s</b> bicdxkicaua	0	54946	0					
4	foosdfpkusacimwkcso <b>s</b> bicdxkicaua	0	54946	0					
	date_activ	date_end	date_modif_prod	date_renewal	forecast_cons_12m	\			
0	2013-06-15	2016-06-15	2015-11-01	2015-06-23	0.0				
1	2013-06-15	2016-06-15	2015-11-01	2015-06-23	0.0				
2	2013-06-15	2016-06-15	2015-11-01	2015-06-23	0.0				
3	2013-06-15	2016-06-15	2015-11-01	2015-06-23	0.0				
4	2013-06-15	2016-06-15	2015-11-01	2015-06-23	0.0				
	forecast_cons_year	forecast_discount_energy	forecast_meter_rent_12m		forecast_price_energy_off_peak	forecast_price_energy_peak	\		
0	0	0	1.78	0.114481	0.098142				
1	0	0	1.78	0.114481	0.098142				
2	0	0	1.78	0.114481	0.098142				
3	0	0	1.78	0.114481	0.098142				
4	0	0	1.78	0.114481	0.098142				
	forecast_price_pow_off_peak	has_gas	imp_cons	margin_gross_pow_ele	origin_up	pow_max	y_price_off_peak_var	\	
0	40.606701	t	0.0	25.44	lxidpiddsbxsbosboudacockeimpuepw	43.648	0.000061		
1	40.606701	t	0.0	25.44	lxidpiddsbxsbosboudacockeimpuepw	43.648	0.000061		
2	40.606701	t	0.0	25.44	lxidpiddsbxsbosboudacockeimpuepw	43.648	0.000061		
3	40.606701	t	0.0	25.44	lxidpiddsbxsbosboudacockeimpuepw	43.648	0.000061		
4	40.606701	t	0.0	25.44	lxidpiddsbxsbosboudacockeimpuepw	43.648	0.000061		
	margin_net_pow_ele	nb_prod_act	net_margin	num_years_antig	margin_net_pow_ele	nb_prod_act	net_margin	num_years_antig	\
0	25.44	2	678.99	3	25.44	2	678.99	3	
1	25.44	2	678.99	3	25.44	2	678.99	3	
2	25.44	2	678.99	3	25.44	2	678.99	3	
3	25.44	2	678.99	3	25.44	2	678.99	3	
4	25.44	2	678.99	3	25.44	2	678.99	3	
	origin_up	pow_max	y_price_off_peak_var		origin_up	pow_max	y_price_off_peak_var		
0	lxidpiddsbxsbosboudacockeimpuepw	43.648	0.000061		lxidpiddsbxsbosboudacockeimpuepw	43.648	0.000061		
1	lxidpiddsbxsbosboudacockeimpuepw	43.648	0.000061		lxidpiddsbxsbosboudacockeimpuepw	43.648	0.000061		
2	lxidpiddsbxsbosboudacockeimpuepw	43.648	0.000061		lxidpiddsbxsbosboudacockeimpuepw	43.648	0.000061		
3	lxidpiddsbxsbosboudacockeimpuepw	43.648	0.000061		lxidpiddsbxsbosboudacockeimpuepw	43.648	0.000061		
4	lxidpiddsbxsbosboudacockeimpuepw	43.648	0.000061		lxidpiddsbxsbosboudacockeimpuepw	43.648	0.000061		

```

y_price_peak_var y_price_mid_peak_var y_price_off_peak_fix \
0      0.000026          0.00044        1.102785
1      0.000026          0.00044        1.102785
2      0.000026          0.00044        1.102785
3      0.000026          0.00044        1.102785
4      0.000026          0.00044        1.102785

y_price_peak_fix y_price_mid_peak_fix yv_price _off_peak p1 \
0      49.550703        22.022535      1.102846
1      49.550703        22.022535      1.102846
2      49.550703        22.022535      1.102846
3      49.550703        22.022535      1.102846
4      49.550703        22.022535      1.102846

yv_price_peak p2 yv_price_midpeak p3 6_price_off_peak_var \
0      49.550729        22.022975      0.00011
1      49.550729        22.022975      0.00011
2      49.550729        22.022975      0.00011
3      49.550729        22.022975      0.00011
4      49.550729        22.022975      0.00011

6_price_peak_var 6_price_mid_peak_var 6_price_off_peak_fix \
0      0.000038          0.000771      1.81949
1      0.000038          0.000771      1.81949
2      0.000038          0.000771      1.81949
3      0.000038          0.000771      1.81949
4      0.000038          0.000771      1.81949

6_price_peak_fix 6_price_mid_peak_fix 6mps_price_off_peak p1 \
0      85.199484        37.866442      1.819601
1      85.199484        37.866442      1.819601
2      85.199484        37.866442      1.819601
3      85.199484        37.866442      1.819601
4      85.199484        37.866442      1.819601

6mps_price_peak p2 6mps_price_mid_peak p3  churn
0      85.199522        37.867213      1
1      85.199522        37.867213      1
2      85.199522        37.867213      1
3      85.199522        37.867213      1
4      85.199522        37.867213      1

```

```
[3]: df.drop(columns=['Unnamed: 0'], axis=1, inplace=True)
df.columns
```

```
[3]: Index(['id', 'channel_sales', 'cons_12m', 'cons_gas_12m', 'cons_last_month',
       'date_activ', 'date_end', 'date_modif_prod', 'date_renewal',
       'forecast_cons_12m', 'forecast_cons_year', 'forecast_discount_energy',
       'forecast_meter_rent_12m', 'forecast_price_energy_off_peak',
       'forecast_price_energy_peak', 'forecast_price_pow_off_peak', 'has_gas',
       'imp_cons', 'margin_gross_pow_ele', 'margin_net_pow_ele', 'nb_prod_act',
       'net_margin', 'num_years_antig', 'origin_up', 'pow_max',
       'y_price_off_peak_var', 'y_price_peak_var', 'y_price_mid_peak_var',
       'y_price_off_peak_fix', 'y_price_peak_fix', 'y_price_mid_peak_fix',
       'yv_price_off_peak p1', 'yv_price_peak p2', 'yv_price_midpeak p3',
       '6_price_off_peak_var', '6_price_peak_var', '6_price_mid_peak_var',
       '6_price_off_peak_fix', '6_price_peak_fix', '6_price_mid_peak_fix',
       '6mps_price_off_peak p1', '6mps_price_peak p2',
       '6mps_price_mid_peak p3', 'churn'],
      dtype='object')
```

[4]: df.info()

#	Column	Non-Null Count	Dtype
0	id	175149	non-null object
1	channel_sales	175149	non-null object
2	cons_12m	175149	non-null int64
3	cons_gas_12m	175149	non-null int64
4	cons_last_month	175149	non-null int64
5	date_activ	175149	non-null object
6	date_end	175149	non-null object
7	date_modif_prod	175149	non-null object
8	date_renewal	175149	non-null object
9	forecast_cons_12m	175149	non-null float64
10	forecast_cons_year	175149	non-null int64
11	forecast_discount_energy	175149	non-null int64
12	forecast_meter_rent_12m	175149	non-null float64
13	forecast_price_energy_off_peak	175149	non-null float64
14	forecast_price_energy_peak	175149	non-null float64
15	forecast_price_pow_off_peak	175149	non-null float64
16	has_gas	175149	non-null object
17	imp_cons	175149	non-null float64
18	margin_gross_pow_ele	175149	non-null float64
19	margin_net_pow_ele	175149	non-null float64
20	nb_prod_act	175149	non-null int64
21	net_margin	175149	non-null float64
22	num_years_antig	175149	non-null int64
23	origin_up	175149	non-null object
24	pow_max	175149	non-null float64

```

25 y_price_off_peak_var      175149 non-null float64
26 y_price_peak_var         175149 non-null float64
27 y_price_mid_peak_var    175149 non-null float64
28 y_price_off_peak_fix    175149 non-null float64
29 y_price_peak_fix        175149 non-null float64
30 y_price_mid_peak_fix   175149 non-null float64
31 yv_price _off_peak p1   175149 non-null float64
32 yv_price_peak p2        175149 non-null float64
33 yv_price_midpeak p3     175149 non-null float64
34 6_price_off_peak_var   175149 non-null float64
35 6_price_peak_var       175149 non-null float64
36 6_price_mid_peak_var   175149 non-null float64
37 6_price_off_peak_fix   175149 non-null float64
38 6_price_peak_fix       175149 non-null float64
39 6_price_mid_peak_fix   175149 non-null float64
40 6mps_price_off_peak p1  175149 non-null float64
41 6mps_price_peak p2     175149 non-null float64
42 6mps_price_mid_peak p3 175149 non-null float64
43 churn                   175149 non-null int64
dtypes: float64(28), int64(8), object(8)
memory usage: 58.8+ MB

```

```
[5]: for i in ['date_activ', 'date_end', 'date_modif_prod', 'date_renewal']:
    df[i]=pd.to_datetime(df[i],infer_datetime_format=True)
```

```
[6]: datetime_columns = df.select_dtypes(include=['datetime64']).columns
datetime_columns
```

```
[6]: Index(['date_activ', 'date_end', 'date_modif_prod', 'date_renewal'],
          dtype='object')
```

```
[7]: print(f" Int & Float :{len(df.select_dtypes(include=['int','float']))}.
           ↵columns})\n Cat: { len(df.select_dtypes(include=['object','category']))}.
           ↵columns}):}\n DateTime: {len(df.select_dtypes(include=['datetime64']))}.
           ↵columns})")
```

```
Int & Float :36
Cat: 4
DateTime: 4
```

```
[8]: for i in df.select_dtypes(include=['object','category']).columns:
    print(f"{i}---->{df[i].nunique()}")
```

```
id---->14606
channel_sales---->8
has_gas---->2
origin_up---->6
```

```
[9]: price=pd.read_csv('/kaggle/input/bcgtask/price.csv',parse_dates=['price_date'])
price.head()
```

```
[9]:          id price_date  price_off_peak_var \
0  038af19179925da21a25619c5a24b745  2015-01-01           0.151367
1  038af19179925da21a25619c5a24b745  2015-02-01           0.151367
2  038af19179925da21a25619c5a24b745  2015-03-01           0.151367
3  038af19179925da21a25619c5a24b745  2015-04-01           0.149626
4  038af19179925da21a25619c5a24b745  2015-05-01           0.149626

      price_peak_var  price_mid_peak_var  price_off_peak_fix  price_peak_fix \
0              0.0             0.0           44.266931           0.0
1              0.0             0.0           44.266931           0.0
2              0.0             0.0           44.266931           0.0
3              0.0             0.0           44.266931           0.0
4              0.0             0.0           44.266931           0.0

      price_mid_peak_fix
0                  0.0
1                  0.0
2                  0.0
3                  0.0
4                  0.0
```

```
[10]: # jan , dec price difference of OFF PEAK'S
month=price.groupby(['id','price_date']).aggregate({'price_off_peak_var':
    ~'mean', 'price_off_peak_fix':'mean'}).reset_index()
month.head()
```

```
[10]:          id price_date  price_off_peak_var \
0  0002203ffbb812588b632b9e628cc38d  2015-01-01           0.126098
1  0002203ffbb812588b632b9e628cc38d  2015-02-01           0.126098
2  0002203ffbb812588b632b9e628cc38d  2015-03-01           0.128067
3  0002203ffbb812588b632b9e628cc38d  2015-04-01           0.128067
4  0002203ffbb812588b632b9e628cc38d  2015-05-01           0.128067

      price_off_peak_fix
0            40.565969
1            40.565969
2            40.728885
3            40.728885
4            40.728885
```

## 4 FEATURE ENGINEERING

```
[11]: j=month.groupby('id').first().reset_index()
d= month.groupby('id').last().reset_index()
d.columns
```

```
[11]: Index(['id', 'price_date', 'price_off_peak_var', 'price_off_peak_fix'],
           dtype='object')
```

```
[12]: def col_rename(dat,cols,modulation):
       for i in cols:
           dat=dat.rename(columns={i:f'{modulation}_{i}'})
       return dat
```

```
[13]: jan=col_rename(j,['price_off_peak_var', 'price_off_peak_fix'],'jan')
jan.head(3)
```

```
[13]: id price_date jan_price_off_peak_var \
0 0002203ffbb812588b632b9e628cc38d 2015-01-01 0.126098
1 0004351ebdd665e6ee664792efc4fd13 2015-01-01 0.148047
2 0010bcc39e42b3c2131ed2ce55246e3c 2015-01-01 0.150837

jan_price_off_peak_fix
0 40.565969
1 44.266931
2 44.444710
```

```
[14]: dec=col_rename(d,['price_off_peak_var', 'price_off_peak_fix'],'dec')
dec.head(3)
```

```
[14]: id price_date dec_price_off_peak_var \
0 0002203ffbb812588b632b9e628cc38d 2015-12-01 0.119906
1 0004351ebdd665e6ee664792efc4fd13 2015-12-01 0.143943
2 0010bcc39e42b3c2131ed2ce55246e3c 2015-12-01 0.201280

dec_price_off_peak_fix
0 40.728885
1 44.444710
2 45.944710
```

```
[15]: diff=pd.DataFrame()
diff['id']=jan['id'].to_list()
diff['price_date']=dec['price_date'].to_list()
diff["D-J_of_pk_fix"]=dec['dec_price_off_peak_fix']-jan['jan_price_off_peak_fix']
diff["D-J_of_pk_var"]=dec['dec_price_off_peak_var']-jan['jan_price_off_peak_var']
diff.head()
```

```
[15]: id price_date D-J_of_pk_fix D-J_of_pk_var
0 0002203ffbb812588b632b9e628cc38d 2015-12-01 0.162916 -0.006192
1 0004351ebdd665e6ee664792efc4fd13 2015-12-01 0.177779 -0.004104
2 0010bcc39e42b3c2131ed2ce55246e3c 2015-12-01 1.500000 0.050443
3 0010ee3855fdea87602a5b7aba8e42de 2015-12-01 0.162916 -0.010018
4 00114d74e963e47177db89bc70108537 2015-12-01 -0.000001 -0.003994
```

```
[16]: # merging the df and diff dataframe
df=pd.merge(df,diff,on='id')
df.head(3)
```

```
[16]: id channel_sales \
0 24011ae4ebbe3035111d65fa7c15bc57 foosdfpfkusacimwkcsosbicdxkicaua
1 24011ae4ebbe3035111d65fa7c15bc57 foosdfpfkusacimwkcsosbicdxkicaua
2 24011ae4ebbe3035111d65fa7c15bc57 foosdfpfkusacimwkcsosbicdxkicaua

cons_12m cons_gas_12m cons_last_month date_activ date_end \
0 0 54946 0 2013-06-15 2016-06-15
1 0 54946 0 2013-06-15 2016-06-15
2 0 54946 0 2013-06-15 2016-06-15

date_modif_prod date_renewal forecast_cons_12m forecast_cons_year \
0 2015-11-01 2015-06-23 0.0 0
1 2015-11-01 2015-06-23 0.0 0
2 2015-11-01 2015-06-23 0.0 0

forecast_discount_energy forecast_meter_rent_12m \
0 0 1.78
1 0 1.78
2 0 1.78

forecast_price_energy_off_peak forecast_price_energy_peak \
0 0.114481 0.098142
1 0.114481 0.098142
2 0.114481 0.098142

forecast_price_pow_off_peak has_gas imp_cons margin_gross_pow_ele \
0 40.606701 t 0.0 25.44
1 40.606701 t 0.0 25.44
2 40.606701 t 0.0 25.44

margin_net_pow_ele nb_prod_act net_margin num_years_antig \
0 25.44 2 678.99 3
1 25.44 2 678.99 3
2 25.44 2 678.99 3

origin_up pow_max y_price_off_peak_var \
```

```

0 lxitpiddsbxsbsboudacockeimpuepw 43.648 0.000061
1 lxitpiddsbxsbsboudacockeimpuepw 43.648 0.000061
2 lxitpiddsbxsbsboudacockeimpuepw 43.648 0.000061

y_price_peak_var y_price_mid_peak_var y_price_off_peak_fix \
0 0.000026 0.00044 1.102785
1 0.000026 0.00044 1.102785
2 0.000026 0.00044 1.102785

y_price_peak_fix y_price_mid_peak_fix yv_price_off_peak p1 \
0 49.550703 22.022535 1.102846
1 49.550703 22.022535 1.102846
2 49.550703 22.022535 1.102846

yv_price_peak p2 yv_price_midpeak p3 6_price_off_peak_var \
0 49.550729 22.022975 0.00011
1 49.550729 22.022975 0.00011
2 49.550729 22.022975 0.00011

6_price_peak_var 6_price_mid_peak_var 6_price_off_peak_fix \
0 0.000038 0.000771 1.81949
1 0.000038 0.000771 1.81949
2 0.000038 0.000771 1.81949

6_price_peak_fix 6_price_mid_peak_fix 6mps_price_off_peak p1 \
0 85.199484 37.866442 1.819601
1 85.199484 37.866442 1.819601
2 85.199484 37.866442 1.819601

6mps_price_peak p2 6mps_price_mid_peak p3 churn price_date \
0 85.199522 37.867213 1 2015-12-01
1 85.199522 37.867213 1 2015-12-01
2 85.199522 37.867213 1 2015-12-01

D-J_of_pk_fix D-J_of_pk_var
0 3.700961 0.020057
1 3.700961 0.020057
2 3.700961 0.020057

```

[17]: #AVG-PRICE CHANGES ACROSS- PERIOD

```

avgp=pd.DataFrame()
for i in ['price_off_peak_var', 'price_peak_var',
          'price_mid_peak_var', 'price_off_peak_fix', 'price_peak_fix',
          'price_mid_peak_fix']:
    avgp[f'M{i}']=price.groupby('id').aggregate({i:'mean'})
avgp.head(3)

```

```
[17]:          Mprice_off_peak_var  Mprice_peak_var \
id
0002203ffbb812588b632b9e628cc38d      0.124338      0.103794
0004351ebdd665e6ee664792efc4fd13      0.146426      0.000000
0010bcc39e42b3c2131ed2ce55246e3c      0.181558      0.000000

          Mprice_mid_peak_var  Mprice_off_peak_fix \
id
0002203ffbb812588b632b9e628cc38d      0.07316       40.701732
0004351ebdd665e6ee664792efc4fd13      0.00000       44.385450
0010bcc39e42b3c2131ed2ce55246e3c      0.00000       45.319710

          Mprice_peak_fix  Mprice_mid_peak_fix
id
0002203ffbb812588b632b9e628cc38d      24.421038     16.280694
0004351ebdd665e6ee664792efc4fd13      0.000000      0.000000
0010bcc39e42b3c2131ed2ce55246e3c      0.000000      0.000000
```

```
[18]: # CLACULATING THE DIFFERENCES OF
# var
avgp['mdiff_(peak-offpeak)var']=avgp['Mprice_peak_var']-avgp['Mprice_off_peak_var']
avgp['mdiff_(offpeak-midpeak)var']=avgp['Mprice_off_peak_var']-avgp['Mprice_mid_peak_var']
avgp['mdiff_(mid-peak)var']=avgp['Mprice_mid_peak_var']-avgp['Mprice_peak_var']

#fix
avgp['mdiff_(peak-offpeak)fix']=avgp['Mprice_peak_fix']-avgp['Mprice_off_peak_fix']
avgp['mdiff_(offpeak-midpeak)fix']=avgp['Mprice_off_peak_fix']-avgp['Mprice_mid_peak_fix']
avgp['mdiff_(mid-peak)fix']=avgp['Mprice_mid_peak_fix']-avgp['Mprice_peak_fix']

avgp=avgp.reset_index()
avgp.head()
```

```
[18]:   id  Mprice_off_peak_var  Mprice_peak_var \
0  0002203ffbb812588b632b9e628cc38d      0.124338      0.103794
1  0004351ebdd665e6ee664792efc4fd13      0.146426      0.000000
2  0010bcc39e42b3c2131ed2ce55246e3c      0.181558      0.000000
3  0010ee3855fdea87602a5b7aba8e42de      0.118757      0.098292
4  00114d74e963e47177db89bc70108537      0.147926      0.000000

          Mprice_mid_peak_var  Mprice_off_peak_fix  Mprice_peak_fix \
0            0.073160       40.701732      24.421038
1            0.000000       44.385450      0.000000
2            0.000000       45.319710      0.000000
3            0.069032       40.647427      24.388455
4            0.000000       44.266930      0.000000

          Mprice_mid_peak_fix  mdiff_(peak-offpeak)var  mdiff_(offpeak-midpeak)var \

```

```

0          16.280694           -0.020545        0.051178
1          0.000000           -0.146426        0.146426
2          0.000000           -0.181558        0.181558
3          16.258971           -0.020465        0.049725
4          0.000000           -0.147926        0.147926

    mdiff_(mid-peak)var  mdiff_(peak-offpeak)fix  mdiff_(offpeak-midpeak)fix \
0          -0.030633           -16.280694       24.421038
1          0.000000           -44.385450       44.385450
2          0.000000           -45.319710       45.319710
3          -0.029260           -16.258972       24.388456
4          0.000000           -44.266930       44.266930

    mdiff_(mid-peak)fix
0          -8.140345
1          0.000000
2          0.000000
3          -8.129484
4          0.000000

```

```
[19]: df=pd.merge(df,avgp[['id','mdiff_(peak-offpeak)var',\
                           'mdiff_(offpeak-midpeak)var',\
                           'mdiff_(mid-peak)var', 'mdiff_(peak-offpeak)fix',\
                           'mdiff_(offpeak-midpeak)fix', 'mdiff_(mid-peak)fix']],on='id')
df.head(4)
```

```
[19]:                      id            channel_sales \
0  24011ae4ebbe3035111d65fa7c15bc57  foosdfpfkusacimwkcsovbcidxkicaua
1  24011ae4ebbe3035111d65fa7c15bc57  foosdfpfkusacimwkcsovbcidxkicaua
2  24011ae4ebbe3035111d65fa7c15bc57  foosdfpfkusacimwkcsovbcidxkicaua
3  24011ae4ebbe3035111d65fa7c15bc57  foosdfpfkusacimwkcsovbcidxkicaua

    cons_12m  cons_gas_12m  cons_last_month date_activ   date_end \
0          0         54946                  0 2013-06-15 2016-06-15
1          0         54946                  0 2013-06-15 2016-06-15
2          0         54946                  0 2013-06-15 2016-06-15
3          0         54946                  0 2013-06-15 2016-06-15

    date_modif_prod date_renewal forecast_cons_12m forecast_cons_year \
0      2015-11-01  2015-06-23          0.0                 0
1      2015-11-01  2015-06-23          0.0                 0
2      2015-11-01  2015-06-23          0.0                 0
3      2015-11-01  2015-06-23          0.0                 0

    forecast_discount_energy  forecast_meter_rent_12m \
0                   0                     1.78
1                   0                     1.78
```

2	0	1.78
3	0	1.78
0	forecast_price_energy_off_peak	forecast_price_energy_peak \
1	0.114481	0.098142
2	0.114481	0.098142
3	0.114481	0.098142
0	forecast_price_pow_off_peak has_gas	imp_cons margin_gross_pow_ele \
1	40.606701	t 0.0 25.44
2	40.606701	t 0.0 25.44
3	40.606701	t 0.0 25.44
0	margin_net_pow_ele nb_prod_act	net_margin num_years_antig \
1	25.44 2	678.99 3
2	25.44 2	678.99 3
3	25.44 2	678.99 3
0	lxidpiddsbxsbosboudacockeimpuepw	origin_up pow_max y_price_off_peak_var \
1	43.648	0.000061
2	43.648	0.000061
3	43.648	0.000061
0	y_price_peak_var	y_price_mid_peak_var y_price_off_peak_fix \
1	0.000026	0.00044 1.102785
2	0.000026	0.00044 1.102785
3	0.000026	0.00044 1.102785
0	y_price_peak_fix	yv_price _off_peak p1 \
1	49.550703	22.022535 1.102846
2	49.550703	22.022535 1.102846
3	49.550703	22.022535 1.102846
0	yv_price_peak p2	yv_price_midpeak p3 6_price_off_peak_var \
1	49.550729	22.022975 0.00011
2	49.550729	22.022975 0.00011
3	49.550729	22.022975 0.00011
0	6_price_peak_var	6_price_mid_peak_var 6_price_off_peak_fix \
	0.000038	0.000771 1.81949

```

1          0.000038          0.000771        1.81949
2          0.000038          0.000771        1.81949
3          0.000038          0.000771        1.81949

  6_price_peak_fix  6_price_mid_peak_fix  6mps_price_off_peak p1 \
0          85.199484          37.866442      1.819601
1          85.199484          37.866442      1.819601
2          85.199484          37.866442      1.819601
3          85.199484          37.866442      1.819601

  6mps_price_peak p2  6mps_price_mid_peak p3  churn price_date \
0          85.199522          37.867213      1 2015-12-01
1          85.199522          37.867213      1 2015-12-01
2          85.199522          37.867213      1 2015-12-01
3          85.199522          37.867213      1 2015-12-01

  D-J_of_pk_fix  D-J_of_pk_var  mdiff_(peak-offpeak)var \
0          3.700961          0.020057      -0.024038
1          3.700961          0.020057      -0.024038
2          3.700961          0.020057      -0.024038
3          3.700961          0.020057      -0.024038

  mdiff_(offpeak-midpeak)var  mdiff_(mid-peak)var  mdiff_(peak-offpeak)fix \
0          0.058257          -0.034219      -18.590255
1          0.058257          -0.034219      -18.590255
2          0.058257          -0.034219      -18.590255
3          0.058257          -0.034219      -18.590255

  mdiff_(offpeak-midpeak)fix  mdiff_(mid-peak)fix
0          26.040925         -7.45067
1          26.040925         -7.45067
2          26.040925         -7.45067
3          26.040925         -7.45067

```

[20]: `price.columns`

```
[20]: Index(['id', 'price_date', 'price_off_peak_var', 'price_peak_var',
       'price_mid_peak_var', 'price_off_peak_fix', 'price_peak_fix',
       'price_mid_peak_fix'],
      dtype='object')
```

[21]: `avgpr_months=pd.DataFrame()`

[22]: `#MAX PRICE CHANGE ACROSS PERIODS`

```
for i in ['price_off_peak_var', 'price_peak_var',
          'price_mid_peak_var', 'price_off_peak_fix', 'price_peak_fix',
```

```

'price_mid_peak_fix']:
    avgpr_months[f"frm{x}{i}"] = price.groupby(['id', 'price_date']).
    aggregate({i: 'mean'}) # for max price calculation
avgpr_months = avgpr_months.reset_index()
avgpr_months.head(3)

```

[22]:

	id	price_date	frm{x}price_off_peak_var
0	0002203ffbb812588b632b9e628cc38d	2015-01-01	0.126098
1	0002203ffbb812588b632b9e628cc38d	2015-02-01	0.126098
2	0002203ffbb812588b632b9e628cc38d	2015-03-01	0.128067

	frm{x}price_peak_var	frm{x}price_mid_peak_var	frm{x}price_off_peak_fix
0	0.103975	0.070232	40.565969
1	0.103975	0.070232	40.565969
2	0.105842	0.073773	40.728885

	frm{x}price_peak_fix	frm{x}price_mid_peak_fix
0	24.339581	16.226389
1	24.339581	16.226389
2	24.437330	16.291555

[23]:

```

# var
avgpr_months['frmpdiff_(peak-offpeak)var']=avgpr_months['frm{x}price_peak_var']-avgpr_months['frm{x}price_off_peak_var']
avgpr_months['frmpmdiff_(offpeak-midpeak)var']=avgpr_months['frm{x}price_off_peak_var']-avgpr_months['frm{x}price_mid_peak_var']
avgpr_months['frmpmdiff_(mid-peak)var']=avgpr_months['frm{x}price_mid_peak_var']-avgpr_months['frm{x}price_peak_var']

#fix
avgpr_months['mdiff_(peak-offpeak)fix']=avgpr_months['frm{x}price_peak_fix']-avgpr_months['frm{x}price_off_peak_fix']
avgpr_months['mdiff_(offpeak-midpeak)fix']=avgpr_months['frm{x}price_off_peak_fix']-avgpr_months['frm{x}price_mid_peak_fix']
avgpr_months['mdiff_(mid-peak)fix']=avgpr_months['frm{x}price_mid_peak_fix']-avgpr_months['frm{x}price_peak_fix']

avgpr_months=avgpr_months.reset_index()
avgpr_months.head()

```

[23]:

	index	id	price_date	frm{x}price_off_peak_var
0	0	0002203ffbb812588b632b9e628cc38d	2015-01-01	0.126098
1	1	0002203ffbb812588b632b9e628cc38d	2015-02-01	0.126098
2	2	0002203ffbb812588b632b9e628cc38d	2015-03-01	0.128067
3	3	0002203ffbb812588b632b9e628cc38d	2015-04-01	0.128067
4	4	0002203ffbb812588b632b9e628cc38d	2015-05-01	0.128067

	frm{x}price_peak_var	frm{x}price_mid_peak_var	frm{x}price_off_peak_fix
0	0.103975	0.070232	40.565969
1	0.103975	0.070232	40.565969
2	0.105842	0.073773	40.728885
3	0.105842	0.073773	40.728885
4	0.105842	0.073773	40.728885

```

frmxprice_peak_fix  frmxpathice_mid_peak_fix  frmpdiff_(peak-offpeak)var \
0          24.339581           16.226389          -0.022123
1          24.339581           16.226389          -0.022123
2          24.437330           16.291555          -0.022225
3          24.437330           16.291555          -0.022225
4          24.437330           16.291555          -0.022225

frmpmdiff_(offpeak-midpeak)var  frmpmdiff_(mid-peak)var \
0          0.055866           -0.033743
1          0.055866           -0.033743
2          0.054294           -0.032069
3          0.054294           -0.032069
4          0.054294           -0.032069

mdiff_(peak-offpeak)fix  mdiff_(offpeak-midpeak)fix  mdiff_(mid-peak)fix
0          -16.226389          24.339581          -8.113192
1          -16.226389          24.339581          -8.113192
2          -16.291555          24.437330          -8.145775
3          -16.291555          24.437330          -8.145775
4          -16.291555          24.437330          -8.145775

```

```
[24]: maxpriceid_entire_period=avgpr_months.groupby(['id']).aggregate({
    'frmpdiff_(peak-offpeak)var':np.max,
    'frmpmdiff_(offpeak-midpeak)var':np.max, 'frmpmdiff_(mid-peak)var':np.
    ↪max,
    'mdiff_(peak-offpeak)fix':np.max, 'mdiff_(offpeak-midpeak)fix':np.max,
    'mdiff_(mid-peak)fix':np.max})
maxpriceid_entire_period=maxpriceid_entire_period.reset_index()
for i in ['frmpdiff_(peak-offpeak)var', 'frmpmdiff_(offpeak-midpeak)var',
    'frmpmdiff_(mid-peak)var', 'mdiff_(peak-offpeak)fix',
    'mdiff_(offpeak-midpeak)fix', 'mdiff_(mid-peak)fix']:
    maxpriceid_entire_period=maxpriceid_entire_period.rename(columns={i:
    ↪f"max_{i}"})
maxpriceid_entire_period.head()
```

```
[24]: id  max_frmpdiff_(peak-offpeak)var \
0  0002203ffbb812588b632b9e628cc38d          -0.018233
1  0004351ebdd665e6ee664792efc4fd13          -0.143943
2  0010bcc39e42b3c2131ed2ce55246e3c          -0.150837
3  0010ee3855fdea87602a5b7aba8e42de          -0.017683
4  00114d74e963e47177db89bc70108537          -0.145440

max_frmpmdiff_(offpeak-midpeak)var  max_frmpmdiff_(mid-peak)var \
0          0.055866           -0.027954
1          0.148405           0.000000
2          0.205742           0.000000
```

3	0.054440	-0.025976
4	0.149902	0.000000
0	max_mdiff_(peak-offpeak)fix	max_mdiff_(offpeak-midpeak)fix \
1	-16.226389	24.437330
2	-44.266931	44.444710
3	-44.444710	45.944710
4	-16.226389	24.437330
4	-44.266930	44.266931
0	max_mdiff_(mid-peak)fix	
1	-8.113192	
2	0.000000	
3	0.000000	
4	-8.113192	
4	0.000000	

```
[25]: minpriceid_entire_period=avgpr_months.groupby(['id']).aggregate({
    'frmpdiff_(peak-offpeak)var':np.min,
    'frmpmdiff_(offpeak-midpeak)var':np.min, 'frmpmdiff_(mid-peak)var':np.
    ↪min,
    'mdiff_(peak-offpeak)fix':np.min, 'mdiff_(offpeak-midpeak)fix':np.min,
    'mdiff_(mid-peak)fix':np.min})
minpriceid_entire_period=minpriceid_entire_period.reset_index()
for i in ['frmpdiff_(peak-offpeak)var', 'frmpmdiff_(offpeak-midpeak)var',
          'frmpmdiff_(mid-peak)var', 'mdiff_(peak-offpeak)fix',
          'mdiff_(offpeak-midpeak)fix', 'mdiff_(mid-peak)fix']:
    minpriceid_entire_period=minpriceid_entire_period.rename(columns={i:
    ↪f"min_{i}"})
minpriceid_entire_period.head()
```

0	id	min_frmpdiff_(peak-offpeak)var	\
0	0002203ffbb812588b632b9e628cc38d	-0.022225	
1	0004351ebdd665e6ee664792efc4fd13	-0.148405	
2	0010bcc39e42b3c2131ed2ce55246e3c	-0.205742	
3	0010ee3855fdea87602a5b7aba8e42de	-0.022581	
4	00114d74e963e47177db89bc70108537	-0.149902	
0	min_frpmdiff_(offpeak-midpeak)var	min_frpmdiff_(mid-peak)var	\
0	0.046187	-0.033743	
1	0.143943	0.000000	
2	0.150837	0.000000	
3	0.043659	-0.031859	
4	0.145440	0.000000	
0	min_mdiff_(peak-offpeak)fix	min_mdiff_(offpeak-midpeak)fix	\
0	-16.291555	24.339581	

```

1          -44.444710        44.266931
2          -45.944710        44.444710
3          -16.291555        24.339581
4          -44.266931        44.266930

    min_mdiff_(mid-peak)fix
0            -8.145775
1            0.000000
2            0.000000
3            -8.145775
4            0.000000

```

```
[26]: df=df.merge(maxpriceid_entire_period,on='id').
      merge(minpriceid_entire_period,on='id')
df.head()
```

```
[26]:           id      channel_sales \
0  24011ae4ebbe3035111d65fa7c15bc57  foosdfpfkusalimwkcsozbicdxkicaua
1  24011ae4ebbe3035111d65fa7c15bc57  foosdfpfkusalimwkcsozbicdxkicaua
2  24011ae4ebbe3035111d65fa7c15bc57  foosdfpfkusalimwkcsozbicdxkicaua
3  24011ae4ebbe3035111d65fa7c15bc57  foosdfpfkusalimwkcsozbicdxkicaua
4  24011ae4ebbe3035111d65fa7c15bc57  foosdfpfkusalimwkcsozbicdxkicaua

      cons_12m  cons_gas_12m  cons_last_month date_activ   date_end \
0          0        54946                  0 2013-06-15 2016-06-15
1          0        54946                  0 2013-06-15 2016-06-15
2          0        54946                  0 2013-06-15 2016-06-15
3          0        54946                  0 2013-06-15 2016-06-15
4          0        54946                  0 2013-06-15 2016-06-15

  date_modif_prod date_renewal forecast_cons_12m forecast_cons_year \
0    2015-11-01    2015-06-23        0.0                  0
1    2015-11-01    2015-06-23        0.0                  0
2    2015-11-01    2015-06-23        0.0                  0
3    2015-11-01    2015-06-23        0.0                  0
4    2015-11-01    2015-06-23        0.0                  0

forecast_discount_energy  forecast_meter_rent_12m \
0                      0                  1.78
1                      0                  1.78
2                      0                  1.78
3                      0                  1.78
4                      0                  1.78

forecast_price_energy_off_peak  forecast_price_energy_peak \
0                 0.114481        0.098142
1                 0.114481        0.098142
```

2	0.114481		0.098142
3	0.114481		0.098142
4	0.114481		0.098142
	forecast_price_pow_off_peak has_gas	imp_cons	margin_gross_pow_ele
0	40.606701	t	0.0
1	40.606701	t	0.0
2	40.606701	t	0.0
3	40.606701	t	0.0
4	40.606701	t	0.0
	margin_net_pow_ele nb_prod_act	net_margin	num_years_antig
0	25.44	2	678.99
1	25.44	2	678.99
2	25.44	2	678.99
3	25.44	2	678.99
4	25.44	2	678.99
	origin_up pow_max	y_price_off_peak_var	
0	lxidpiddsbxsbsboudacockeimpuepw	43.648	
1	lxidpiddsbxsbsboudacockeimpuepw	43.648	
2	lxidpiddsbxsbsboudacockeimpuepw	43.648	
3	lxidpiddsbxsbsboudacockeimpuepw	43.648	
4	lxidpiddsbxsbsboudacockeimpuepw	43.648	
	y_price_peak_var y_price_mid_peak_var	y_price_off_peak_fix	
0	0.000026	0.00044	
1	0.000026	0.00044	
2	0.000026	0.00044	
3	0.000026	0.00044	
4	0.000026	0.00044	
	y_price_peak_fix y_price_mid_peak_fix	yv_price _off_peak p1	
0	49.550703	22.022535	
1	49.550703	22.022535	
2	49.550703	22.022535	
3	49.550703	22.022535	
4	49.550703	22.022535	
	yv_price_peak p2 yv_price_midpeak p3	6_price_off_peak_var	
0	49.550729	22.022975	
1	49.550729	22.022975	
2	49.550729	22.022975	
3	49.550729	22.022975	
4	49.550729	22.022975	
	6_price_peak_var 6_price_mid_peak_var	6_price_off_peak_fix	

0	0.000038	0.000771	1.81949
1	0.000038	0.000771	1.81949
2	0.000038	0.000771	1.81949
3	0.000038	0.000771	1.81949
4	0.000038	0.000771	1.81949
	6_price_peak_fix	6_price_mid_peak_fix	6mps_price_off_peak p1 \
0	85.199484	37.866442	1.819601
1	85.199484	37.866442	1.819601
2	85.199484	37.866442	1.819601
3	85.199484	37.866442	1.819601
4	85.199484	37.866442	1.819601
	6mps_price_peak p2	6mps_price_mid_peak p3	churn price_date \
0	85.199522	37.867213	1 2015-12-01
1	85.199522	37.867213	1 2015-12-01
2	85.199522	37.867213	1 2015-12-01
3	85.199522	37.867213	1 2015-12-01
4	85.199522	37.867213	1 2015-12-01
	D-J_of_pk_fix	D-J_of_pk_var	mdiff_(peak-offpeak)var \
0	3.700961	0.020057	-0.024038
1	3.700961	0.020057	-0.024038
2	3.700961	0.020057	-0.024038
3	3.700961	0.020057	-0.024038
4	3.700961	0.020057	-0.024038
	mdiff_(offpeak-midpeak)var	mdiff_(mid-peak)var	mdiff_(peak-offpeak)fix \
0		0.058257	-0.034219 -18.590255
1		0.058257	-0.034219 -18.590255
2		0.058257	-0.034219 -18.590255
3		0.058257	-0.034219 -18.590255
4		0.058257	-0.034219 -18.590255
	mdiff_(offpeak-midpeak)fix	mdiff_(mid-peak)fix \	
0		26.040925	-7.45067
1		26.040925	-7.45067
2		26.040925	-7.45067
3		26.040925	-7.45067
4		26.040925	-7.45067
	max_frmpdiff_(peak-offpeak)var	max_frmpmdiff_(offpeak-midpeak)var \	
0		-0.017685	0.146033
1		-0.017685	0.146033
2		-0.017685	0.146033
3		-0.017685	0.146033
4		-0.017685	0.146033

```

max_frmpmdiff_(mid-peak)var  max_mdiff_(peak-offpeak)fix  \
0                      -0.025975                  -16.226389
1                      -0.025975                  -16.226389
2                      -0.025975                  -16.226389
3                      -0.025975                  -16.226389
4                      -0.025975                  -16.226389

max_mdiff_(offpeak-midpeak)fix  max_mdiff_(mid-peak)fix  \
0                      44.26693                   0.0
1                      44.26693                   0.0
2                      44.26693                   0.0
3                      44.26693                   0.0
4                      44.26693                   0.0

min_frmpmdiff_(peak-offpeak)var  min_frmpmdiff_(offpeak-midpeak)var  \
0                      -0.06055                  0.04366
1                      -0.06055                  0.04366
2                      -0.06055                  0.04366
3                      -0.06055                  0.04366
4                      -0.06055                  0.04366

min_frmpmdiff_(mid-peak)var  min_mdiff_(peak-offpeak)fix  \
0                      -0.085483                 -44.26693
1                      -0.085483                 -44.26693
2                      -0.085483                 -44.26693
3                      -0.085483                 -44.26693
4                      -0.085483                 -44.26693

min_mdiff_(offpeak-midpeak)fix  min_mdiff_(mid-peak)fix
0                      24.339581                -8.145775
1                      24.339581                -8.145775
2                      24.339581                -8.145775
3                      24.339581                -8.145775
4                      24.339581                -8.145775

```

[27]: df.shape

[27]: (175149, 65)

## ADDITIONAL FEATURES

- No of days of subscription = enddate-date of activity
- Lead Time to Renewal (Days): date\_renewal - date\_activ.

1. It gives you the raw number of days between the activation of a contract and its renewal

For example, if a customer activated their contract on January 1st and their renewal date is set for February 1st, the Lead Time to Renewal would be 31 days.

```
[28]: df['yrs_subscribed']=((df['date_end']-df['date_activ'])/365).dt.days
df['Leadtime_renewal']=((df['date_renewal']-df['date_activ'])/365).dt.days
```

```
[29]: df
```

```
[29]:
```

		id	channel_sales \			
0	24011ae4ebbe3035111d65fa7c15bc57	foosdfpfkusacimwkcsosbicdxkicaua				
1	24011ae4ebbe3035111d65fa7c15bc57	foosdfpfkusacimwkcsosbicdxkicaua				
2	24011ae4ebbe3035111d65fa7c15bc57	foosdfpfkusacimwkcsosbicdxkicaua				
3	24011ae4ebbe3035111d65fa7c15bc57	foosdfpfkusacimwkcsosbicdxkicaua				
4	24011ae4ebbe3035111d65fa7c15bc57	foosdfpfkusacimwkcsosbicdxkicaua				
...		...	...			
175144	563dde550fd624d7352f3de77c0cdfcd		MISSING			
175145	563dde550fd624d7352f3de77c0cdfcd		MISSING			
175146	563dde550fd624d7352f3de77c0cdfcd		MISSING			
175147	563dde550fd624d7352f3de77c0cdfcd		MISSING			
175148	563dde550fd624d7352f3de77c0cdfcd		MISSING			
		cons_12m	cons_gas_12m	cons_last_month	date_activ	date_end \
0		0	54946		0	2013-06-15 2016-06-15
1		0	54946		0	2013-06-15 2016-06-15
2		0	54946		0	2013-06-15 2016-06-15
3		0	54946		0	2013-06-15 2016-06-15
4		0	54946		0	2013-06-15 2016-06-15
...		...	...	...	...	...
175144		8730	0		0	2009-12-18 2016-12-17
175145		8730	0		0	2009-12-18 2016-12-17
175146		8730	0		0	2009-12-18 2016-12-17
175147		8730	0		0	2009-12-18 2016-12-17
175148		8730	0		0	2009-12-18 2016-12-17
		date_modif_prod	date_renewal	forecast_cons_12m	forecast_cons_year \	
0		2015-11-01	2015-06-23	0.00	0	
1		2015-11-01	2015-06-23	0.00	0	
2		2015-11-01	2015-06-23	0.00	0	
3		2015-11-01	2015-06-23	0.00	0	
4		2015-11-01	2015-06-23	0.00	0	
...		...	...	...	...	
175144		2009-12-18	2015-12-21	762.41	0	
175145		2009-12-18	2015-12-21	762.41	0	
175146		2009-12-18	2015-12-21	762.41	0	
175147		2009-12-18	2015-12-21	762.41	0	
175148		2009-12-18	2015-12-21	762.41	0	
		forecast_discount_energy	forecast_meter_rent_12m \			
0		0	1.78			
1		0	1.78			

2	0	1.78		
3	0	1.78		
4	0	1.78		
...	...	...		
175144	0	1.07		
175145	0	1.07		
175146	0	1.07		
175147	0	1.07		
175148	0	1.07		
forecast_price_energy_off_peak forecast_price_energy_peak \				
0	0.114481	0.098142		
1	0.114481	0.098142		
2	0.114481	0.098142		
3	0.114481	0.098142		
4	0.114481	0.098142		
...	...	...		
175144	0.167086	0.088454		
175145	0.167086	0.088454		
175146	0.167086	0.088454		
175147	0.167086	0.088454		
175148	0.167086	0.088454		
forecast_price_pow_off_peak has_gas imp_cons margin_gross_pow_ele \				
0	40.606701	t	0.0	25.44
1	40.606701	t	0.0	25.44
2	40.606701	t	0.0	25.44
3	40.606701	t	0.0	25.44
4	40.606701	t	0.0	25.44
...	...	...	...	...
175144	45.311378	f	0.0	11.84
175145	45.311378	f	0.0	11.84
175146	45.311378	f	0.0	11.84
175147	45.311378	f	0.0	11.84
175148	45.311378	f	0.0	11.84
margin_net_pow_ele nb_prod_act net_margin num_years_antig \				
0	25.44	2	678.99	3
1	25.44	2	678.99	3
2	25.44	2	678.99	3
3	25.44	2	678.99	3
4	25.44	2	678.99	3
...	...	...	...	...
175144	11.84	1	96.34	6
175145	11.84	1	96.34	6
175146	11.84	1	96.34	6
175147	11.84	1	96.34	6

175148	11.84	1	96.34	6
		origin_up	pow_max	y_price_off_peak_var \
0	lxidpiddsbxsbsbosboudacockeimpuepw	43.648		0.000061
1	lxidpiddsbxsbsbosboudacockeimpuepw	43.648		0.000061
2	lxidpiddsbxsbsbosboudacockeimpuepw	43.648		0.000061
3	lxidpiddsbxsbsbosboudacockeimpuepw	43.648		0.000061
4	lxidpiddsbxsbsbosboudacockeimpuepw	43.648		0.000061
...	...	...	...	...
175144	ldkssxwpmemidmecebumciepifcamkci	10.392		0.000006
175145	ldkssxwpmemidmecebumciepifcamkci	10.392		0.000006
175146	ldkssxwpmemidmecebumciepifcamkci	10.392		0.000006
175147	ldkssxwpmemidmecebumciepifcamkci	10.392		0.000006
175148	ldkssxwpmemidmecebumciepifcamkci	10.392		0.000006
		y_price_peak_var	y_price_mid_peak_var	y_price_off_peak_fix \
0	2.627605e-05		0.00044	1.102785e+00
1	2.627605e-05		0.00044	1.102785e+00
2	2.627605e-05		0.00044	1.102785e+00
3	2.627605e-05		0.00044	1.102785e+00
4	2.627605e-05		0.00044	1.102785e+00
...	...	...	...	...
175144	1.503433e-07		0.00000	3.818182e-13
175145	1.503433e-07		0.00000	3.818182e-13
175146	1.503433e-07		0.00000	3.818182e-13
175147	1.503433e-07		0.00000	3.818182e-13
175148	1.503433e-07		0.00000	3.818182e-13
		y_price_peak_fix	y_price_mid_peak_fix	yv_price _off_peak p1 \
0	49.550703		22.022535	1.102846
1	49.550703		22.022535	1.102846
2	49.550703		22.022535	1.102846
3	49.550703		22.022535	1.102846
4	49.550703		22.022535	1.102846
...	...	...	...	...
175144	0.000000		0.000000	0.000006
175145	0.000000		0.000000	0.000006
175146	0.000000		0.000000	0.000006
175147	0.000000		0.000000	0.000006
175148	0.000000		0.000000	0.000006
		yv_price_peak p2	yv_price_midpeak p3	6_price_off_peak_var \
0	4.955073e+01		22.022975	0.000110
1	4.955073e+01		22.022975	0.000110
2	4.955073e+01		22.022975	0.000110
3	4.955073e+01		22.022975	0.000110
4	4.955073e+01		22.022975	0.000110

...	...	...	...	
175144	1.503433e-07	0.000000	0.000005	
175145	1.503433e-07	0.000000	0.000005	
175146	1.503433e-07	0.000000	0.000005	
175147	1.503433e-07	0.000000	0.000005	
175148	1.503433e-07	0.000000	0.000005	
	6_price_peak_var	6_price_mid_peak_var	6_price_off_peak_fix	\
0	3.817340e-05	0.000771	1.81949	
1	3.817340e-05	0.000771	1.81949	
2	3.817340e-05	0.000771	1.81949	
3	3.817340e-05	0.000771	1.81949	
4	3.817340e-05	0.000771	1.81949	
...	...	...	...	
175144	1.350021e-07	0.000000	0.000000	
175145	1.350021e-07	0.000000	0.000000	
175146	1.350021e-07	0.000000	0.000000	
175147	1.350021e-07	0.000000	0.000000	
175148	1.350021e-07	0.000000	0.000000	
	6_price_peak_fix	6_price_mid_peak_fix	6mps_price_off_peak p1	\
0	85.199484	37.866442	1.819601	
1	85.199484	37.866442	1.819601	
2	85.199484	37.866442	1.819601	
3	85.199484	37.866442	1.819601	
4	85.199484	37.866442	1.819601	
...	...	...	...	
175144	0.000000	0.000000	0.000005	
175145	0.000000	0.000000	0.000005	
175146	0.000000	0.000000	0.000005	
175147	0.000000	0.000000	0.000005	
175148	0.000000	0.000000	0.000005	
	6mps_price_peak p2	6mps_price_mid_peak p3	churn price_date	\
0	8.519952e+01	37.867213	1 2015-12-01	
1	8.519952e+01	37.867213	1 2015-12-01	
2	8.519952e+01	37.867213	1 2015-12-01	
3	8.519952e+01	37.867213	1 2015-12-01	
4	8.519952e+01	37.867213	1 2015-12-01	
...	...	...	...	
175144	1.350021e-07	0.000000	0 2015-12-01	
175145	1.350021e-07	0.000000	0 2015-12-01	
175146	1.350021e-07	0.000000	0 2015-12-01	
175147	1.350021e-07	0.000000	0 2015-12-01	
175148	1.350021e-07	0.000000	0 2015-12-01	
	D-J_of_pk_fix	D-J_of_pk_var	mdiff_(peak-offpeak)var	\

0	3.700961	0.020057	-0.024038
1	3.700961	0.020057	-0.024038
2	3.700961	0.020057	-0.024038
3	3.700961	0.020057	-0.024038
4	3.700961	0.020057	-0.024038
...	...	...	...
175144	-0.000001	-0.004628	-0.081317
175145	-0.000001	-0.004628	-0.081317
175146	-0.000001	-0.004628	-0.081317
175147	-0.000001	-0.004628	-0.081317
175148	-0.000001	-0.004628	-0.081317
	mdiff_(offpeak-midpeak)var	mdiff_(mid-peak)var	\
0	0.058257	-0.034219	
1	0.058257	-0.034219	
2	0.058257	-0.034219	
3	0.058257	-0.034219	
4	0.058257	-0.034219	
...	...	...	
175144	0.168662	-0.087344	
175145	0.168662	-0.087344	
175146	0.168662	-0.087344	
175147	0.168662	-0.087344	
175148	0.168662	-0.087344	
	mdiff_(peak-offpeak)fix	mdiff_(offpeak-midpeak)fix	\
0	-18.590255	26.040925	
1	-18.590255	26.040925	
2	-18.590255	26.040925	
3	-18.590255	26.040925	
4	-18.590255	26.040925	
...	...	...	
175144	-44.266930	44.266930	
175145	-44.266930	44.266930	
175146	-44.266930	44.266930	
175147	-44.266930	44.266930	
175148	-44.266930	44.266930	
	mdiff_(mid-peak)fix	max_frmpdiff_(peak-offpeak)var	\
0	-7.45067	-0.017685	
1	-7.45067	-0.017685	
2	-7.45067	-0.017685	
3	-7.45067	-0.017685	
4	-7.45067	-0.017685	
...	...	...	
175144	0.00000	-0.079057	
175145	0.00000	-0.079057	

175146	0.00000	-0.079057
175147	0.00000	-0.079057
175148	0.00000	-0.079057
	max_frmpmdiff_(offpeak-midpeak)var	max_frmpmdiff_(mid-peak)var \
0	0.146033	-0.025975
1	0.146033	-0.025975
2	0.146033	-0.025975
3	0.146033	-0.025975
4	0.146033	-0.025975
...	...	...
175144	0.170590	-0.086905
175145	0.170590	-0.086905
175146	0.170590	-0.086905
175147	0.170590	-0.086905
175148	0.170590	-0.086905
	max_mdiff_(peak-offpeak)fix	max_mdiff_(offpeak-midpeak)fix \
0	-16.226389	44.266930
1	-16.226389	44.266930
2	-16.226389	44.266930
3	-16.226389	44.266930
4	-16.226389	44.266930
...	...	...
175144	-44.266930	44.266931
175145	-44.266930	44.266931
175146	-44.266930	44.266931
175147	-44.266930	44.266931
175148	-44.266930	44.266931
	max_mdiff_(mid-peak)fix	min_frmpdiff_(peak-offpeak)var \
0	0.0	-0.060550
1	0.0	-0.060550
2	0.0	-0.060550
3	0.0	-0.060550
4	0.0	-0.060550
...	...	...
175144	0.0	-0.082932
175145	0.0	-0.082932
175146	0.0	-0.082932
175147	0.0	-0.082932
175148	0.0	-0.082932
	min_frmpmdiff_(offpeak-midpeak)var	min_frmpmdiff_(mid-peak)var \
0	0.043660	-0.085483
1	0.043660	-0.085483
2	0.043660	-0.085483

```

3                      0.043660          -0.085483
4                      0.043660          -0.085483
...
175144                  ...          ...
175145                  0.165962          -0.087658
175146                  0.165962          -0.087658
175147                  0.165962          -0.087658
175148                  0.165962          -0.087658

      min_mdiff_(peak-offpeak)fix  min_mdiff_(offpeak-midpeak)fix \
0                  -44.266930          24.339581
1                  -44.266930          24.339581
2                  -44.266930          24.339581
3                  -44.266930          24.339581
4                  -44.266930          24.339581
...
175144                  ...          ...
175145                  -44.266931          44.266930
175146                  -44.266931          44.266930
175147                  -44.266931          44.266930
175148                  -44.266931          44.266930

      min_mdiff_(mid-peak)fix  yrs_subscribed  Leadtime_renewal
0                  -8.145775          3              2
1                  -8.145775          3              2
2                  -8.145775          3              2
3                  -8.145775          3              2
4                  -8.145775          3              2
...
175144                  ...          ...
175145                  0.000000          7              6
175146                  0.000000          7              6
175147                  0.000000          7              6
175148                  0.000000          7              6

[175149 rows x 67 columns]

```

```
[30]: lt=df.groupby('Leadtime_renewal')['churn'].mean().sort_values(ascending=False).
       ↪reset_index()
lt
```

```
[30]:   Leadtime_renewal      churn
0                 2  0.141592
1                 1  0.137283
2                 3  0.128419
3                12  0.095238
4                 4  0.091094
```

```
5           11  0.083333
6             6  0.075796
7             5  0.075482
8            10  0.060467
9             7  0.048064
10            9  0.044444
11            8  0.012195
12            0  0.000000
```

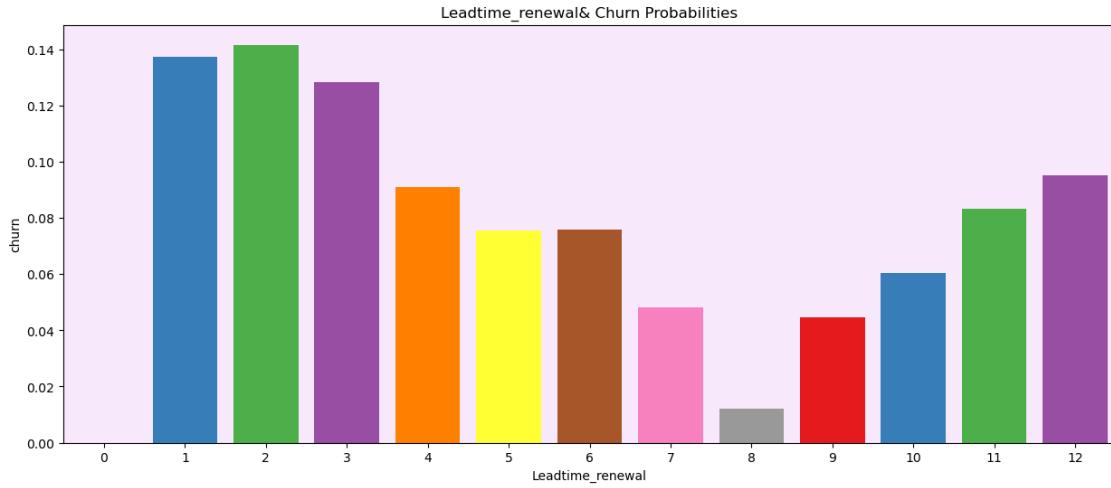
```
[31]: plt.rcParams['axes.facecolor']='#f2d8fa96'
```

```
[32]: yrs=df.groupby('yrs_subscribed')['churn'].mean().sort_values(ascending=False).
       ↪reset_index()
yrs
```

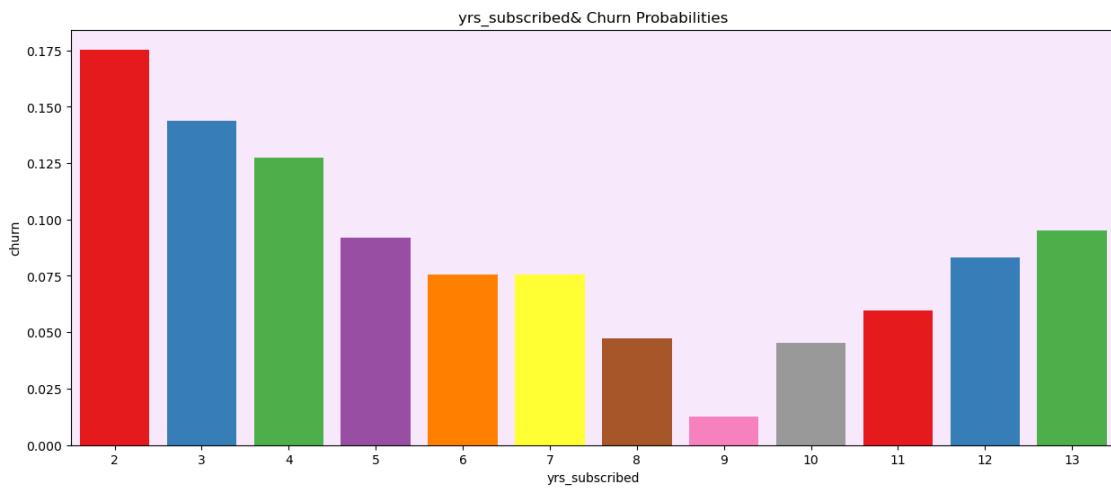
```
[32]:    yrs_subscribed      churn
0                  2  0.175410
1                  3  0.143799
2                  4  0.127406
3                 13  0.095238
4                  5  0.091863
5                 12  0.083333
6                  7  0.075433
7                  6  0.075385
8                 11  0.059810
9                  8  0.047306
10                10  0.045455
11                9  0.012500
```

```
[33]: def barplot(da,a,b,pal,r):
        plt.figure(figsize=(15,6))
        sns.barplot(data=da,x=a,y=b,palette=pal,saturation=r)
        plt.title(f"{a}& Churn Probabilities")
        plt.show()
```

```
[34]: barplot(lt,'Leadtime_renewal','churn','Set1',1)
```



```
[35]: barplott(yrs, 'yrs_subscribed', 'churn', 'Set1', 1)
```



```
[36]: donttouch=df.copy() # i will kill if you touch this for any of it.
```

```
[112]: oldf=df.copy()# old copy for graphs
ft=df.copy() # use this for transformations
```

```
[113]: c=['cons_12m','cons_gas_12m','cons_last_month','forecast_cons_12m','forecast_cons_year','forecast_
↳ 'forecast_price_pow_off_peak','imp_cons','margin_gross_pow_ele','margin_net_pow_ele','nb_pr...
```

```

↳ ↳'y_price_peak_var','y_price_mid_peak_var','y_price_off_peak_fix','y_price_peak_fix','y_price_
↳ _off_peak p1','yv_price_peak p2','yv_price_midpeak
↳ p3','6_price_off_peak_var',
↳ ↳'6_price_peak_var','6_price_mid_peak_var','6_price_off_peak_fix','6_price_peak_fix','6_price_
↳ p1','6mps_price_peak p2','6mps_price_mid_peak p3',
↳ ↳'D-J_of_pk_fix','D-J_of_pk_var','mdiff_(peak-offpeak)var','mdiff_(offpeak-midpeak)var','mdi
↳ ↳'mdiff_(mid-peak)fix','max_frmpdiff_(peak-offpeak)var','max_frmpmdiff_(offpeak-midpeak)var'
↳ ↳'max_mdiff_(offpeak-midpeak)fix','max_mdiff_(mid-peak)fix','min_frmpdiff_(peak-offpeak)var
'&nbsp;'min_mdiff_(offpeak-midpeak)fix','min_mdiff_(mid-peak)fix','yrs_subscribed',
'Leadtime_renewal']

```

## 5 DATA TRANSFORMATION–YEOJOHNSON METHOD

- Its a simple transformation method used in statistics and data analysis to stabilize the variance and make the data more closely follow the gaussian distribution.
- It is extension of box-cox transformation method( only applied for strictly for positive values) but this can be applied for positives, negatives and zeors.
- Stabilize variance: Using this we can easily stabilize the variance of data. It will make the spread of data more consistent and accross different groups or condition.
- The below is th Yeo-Johnson transformation formula, with random variable y and parameter Lambda:

```
[8]: from IPython import display
display.Image("C:/Users/ELCOT/Downloads/New folder/yeo.png")
```

[8]:

$$\psi(\lambda, y) = \begin{cases} ((y + 1)^\lambda - 1)/\lambda & \text{if } \lambda \neq 0, y \geq 0 \\ \log(y + 1) & \text{if } \lambda = 0, y \geq 0 \\ -[(-y + 1)^{2-\lambda} - 1]/(2 - \lambda) & \text{if } \lambda \neq 2, y < 0 \\ -\log(-y + 1) & \text{if } \lambda = 2, y < 0 \end{cases}$$

- Here the parameter Lambda can take any real values which includes 0.ie; (-inf, +inf). For differnt values of Lambda different transofrmations will be applied. And they are:
- Lambda=0 , Log transformation(natural logarithm) for positive values and negative values
- Lambda=1 , No transformation(identity transformation).

- Other values of Lambda, applied power transformations with different behaviors.
- The Lambda value will be chosen based on best one which try to give variance stability with good amount of spread for gaussian distribution. Some of the statistical method used to estimate Lambda is MLE(Maximum Likelihood Estimation).
- One of the disadvantage of this transformation method is , we can't assure that it will work good for all the data. we should make the trial and error analysis with other methods of transformations also. Especially data is extremely skewed and multimodal distributions.
- One of the plus about this method is we can apply to the data whether it is continuous or discrete in nature with values(positive, negative and even zeros).

```
[114]: from scipy.stats import yeojohnson as y
```

```
[115]: #YEJOHNSON TRANSOFRIMATION -- SUPPORTS FOR BOTH NEGATIVE AND POSITIVE VALUES IN A
#FEATURE( SIMILAR TO BOXCOP TRANSFORMATION)

def Transform(dataframe):
    for i in c:
        dataframe[i],_=y(dataframe[i]))
    return dataframe

ft=Transform(ft)
```

## 6 OUTLIER TREATMENTS - Z SCORE METHOD

```
[116]: #Outlier treatment: z score

# Select numeric columns (excluding datetime and object columns)
numeric_cols = ft.select_dtypes(include=['float64', 'int64']).columns
numeric_cols=numeric_cols.drop(['forecast_discount_energy','churn'])

# Calculate Z-scores for each numeric column
z_scores = (ft[numeric_cols] - ft[numeric_cols].mean()) / ft[numeric_cols].std()

# Define a Z-score threshold for outlier removal
z_score_threshold = 3

# Create a mask to identify outliers based on Z-scores
outliers_mask = np.abs(z_scores) > z_score_threshold

# Apply the mask to remove outliers
ft_out = ft[~outliers_mask.any(axis=1)]
```

```
[117]: ft_out
```

[117] :

		<b>id</b>	<b>channel_sales</b>	\			
24	764c75f661154dac3a6c254cd082ea7d	foosdfpfkusalacimwkcscosbicdxkicaua					
25	764c75f661154dac3a6c254cd082ea7d	foosdfpfkusalacimwkcscosbicdxkicaua					
26	764c75f661154dac3a6c254cd082ea7d	foosdfpfkusalacimwkcscosbicdxkicaua					
27	764c75f661154dac3a6c254cd082ea7d	foosdfpfkusalacimwkcscosbicdxkicaua					
28	764c75f661154dac3a6c254cd082ea7d	foosdfpfkusalacimwkcscosbicdxkicaua					
...	...	...	...				
175144	563dde550fd624d7352f3de77c0cdfcd			MISSING			
175145	563dde550fd624d7352f3de77c0cdfcd			MISSING			
175146	563dde550fd624d7352f3de77c0cdfcd			MISSING			
175147	563dde550fd624d7352f3de77c0cdfcd			MISSING			
175148	563dde550fd624d7352f3de77c0cdfcd			MISSING			
		<b>cons_12m</b>	<b>cons_gas_12m</b>	<b>cons_last_month</b>	<b>date_activ</b>	<b>date_end</b>	\
24	6.828974	-0.0		0.0	2010-04-16	2016-04-16	
25	6.828974	-0.0		0.0	2010-04-16	2016-04-16	
26	6.828974	-0.0		0.0	2010-04-16	2016-04-16	
27	6.828974	-0.0		0.0	2010-04-16	2016-04-16	
28	6.828974	-0.0		0.0	2010-04-16	2016-04-16	
...	...	...	...	...	...	...	
175144	10.196953	-0.0		0.0	2009-12-18	2016-12-17	
175145	10.196953	-0.0		0.0	2009-12-18	2016-12-17	
175146	10.196953	-0.0		0.0	2009-12-18	2016-12-17	
175147	10.196953	-0.0		0.0	2009-12-18	2016-12-17	
175148	10.196953	-0.0		0.0	2009-12-18	2016-12-17	
		<b>date_modif_prod</b>	<b>date_renewal</b>	<b>forecast_cons_12m</b>	<b>forecast_cons_year</b>	\	
24	2010-04-16	2015-04-17	6.849719	0.0			
25	2010-04-16	2015-04-17	6.849719	0.0			
26	2010-04-16	2015-04-17	6.849719	0.0			
27	2010-04-16	2015-04-17	6.849719	0.0			
28	2010-04-16	2015-04-17	6.849719	0.0			
...	...	...	...	...	...	...	
175144	2009-12-18	2015-12-21	18.345274	0.0			
175145	2009-12-18	2015-12-21	18.345274	0.0			
175146	2009-12-18	2015-12-21	18.345274	0.0			
175147	2009-12-18	2015-12-21	18.345274	0.0			
175148	2009-12-18	2015-12-21	18.345274	0.0			
		<b>forecast_discount_energy</b>	<b>forecast_meter_rent_12m</b>	\			
24		0	5.120009				
25		0	5.120009				
26		0	5.120009				
27		0	5.120009				
28		0	5.120009				
...	...	...	...				
175144		0	0.774527				

175145	0	0.774527
175146	0	0.774527
175147	0	0.774527
175148	0	0.774527
	forecast_price_energy_off_peak	forecast_price_energy_peak \
24	0.174584	0.086345
25	0.174584	0.086345
26	0.174584	0.086345
27	0.174584	0.086345
28	0.174584	0.086345
...	...	...
175144	0.176013	0.086880
175145	0.176013	0.086880
175146	0.176013	0.086880
175147	0.176013	0.086880
175148	0.176013	0.086880
	forecast_price_pow_off_peak	has_gas imp_cons margin_gross_pow_ele \
24	1704.515088	f -0.0 6.053563
25	1704.515088	f -0.0 6.053563
26	1704.515088	f -0.0 6.053563
27	1704.515088	f -0.0 6.053563
28	1704.515088	f -0.0 6.053563
...	...	...
175144	1786.531993	f -0.0 3.919026
175145	1786.531993	f -0.0 3.919026
175146	1786.531993	f -0.0 3.919026
175147	1786.531993	f -0.0 3.919026
175148	1786.531993	f -0.0 3.919026
	margin_net_pow_ele	nb_prod_act net_margin num_years_antig \
24	6.053064	0.117272 2.496395 1.258073
25	6.053064	0.117272 2.496395 1.258073
26	6.053064	0.117272 2.496395 1.258073
27	6.053064	0.117272 2.496395 1.258073
28	6.053064	0.117272 2.496395 1.258073
...	...	...
175144	3.918791	0.117272 7.456686 1.258073
175145	3.918791	0.117272 7.456686 1.258073
175146	3.918791	0.117272 7.456686 1.258073
175147	3.918791	0.117272 7.456686 1.258073
175148	3.918791	0.117272 7.456686 1.258073
	origin_up	pow_max y_price_off_peak_var \
24	kamkkxfxxuwbds1kwifmmcsiusiuosws	0.746777 0.000005
25	kamkkxfxxuwbds1kwifmmcsiusiuosws	0.746777 0.000005

26	kamkkxfxxuwbds1kwifmmcsiusiuosws	0.746777	0.000005
27	kamkkxfxxuwbds1kwifmmcsiusiuosws	0.746777	0.000005
28	kamkkxfxxuwbds1kwifmmcsiusiuosws	0.746777	0.000005
...	...	...	...
175144	ldkssxwpmemidmecebumciepifcamkci	0.737249	0.000005
175145	ldkssxwpmemidmecebumciepifcamkci	0.737249	0.000005
175146	ldkssxwpmemidmecebumciepifcamkci	0.737249	0.000005
175147	ldkssxwpmemidmecebumciepifcamkci	0.737249	0.000005
175148	ldkssxwpmemidmecebumciepifcamkci	0.737249	0.000005
24	y_price_peak_var	y_price_mid_peak_var	y_price_off_peak_fix \
25	2.551665e-07	-0.0	7.308431e-03
26	2.551665e-07	-0.0	7.308431e-03
27	2.551665e-07	-0.0	7.308431e-03
28	2.551665e-07	-0.0	7.308431e-03
...	...	...	...
175144	1.501067e-07	-0.0	3.819191e-13
175145	1.501067e-07	-0.0	3.819191e-13
175146	1.501067e-07	-0.0	3.819191e-13
175147	1.501067e-07	-0.0	3.819191e-13
175148	1.501067e-07	-0.0	3.819191e-13
24	y_price_peak_fix	y_price_mid_peak_fix	yv_price _off_peak p1 \
25	-0.0	-0.0	0.007314
26	-0.0	-0.0	0.007314
27	-0.0	-0.0	0.007314
28	-0.0	-0.0	0.007314
...	...	...	...
175144	-0.0	-0.0	0.000006
175145	-0.0	-0.0	0.000006
175146	-0.0	-0.0	0.000006
175147	-0.0	-0.0	0.000006
175148	-0.0	-0.0	0.000006
24	yv_price_peak p2	yv_price_midpeak p3	6_price_off_peak_var \
25	2.558508e-07	-0.0	0.000005
26	2.558508e-07	-0.0	0.000005
27	2.558508e-07	-0.0	0.000005
28	2.558508e-07	-0.0	0.000005
...	...	...	...
175144	1.503432e-07	-0.0	0.000005
175145	1.503432e-07	-0.0	0.000005
175146	1.503432e-07	-0.0	0.000005
175147	1.503432e-07	-0.0	0.000005

175148	1.503432e-07	-0.0	0.000005	
24	6_price_peak_var	6_price_mid_peak_var	6_price_off_peak_fix	\
24	1.347807e-07	-0.0	-0.0	
25	1.347807e-07	-0.0	-0.0	
26	1.347807e-07	-0.0	-0.0	
27	1.347807e-07	-0.0	-0.0	
28	1.347807e-07	-0.0	-0.0	
...	...	...	...	
175144	1.347807e-07	-0.0	-0.0	
175145	1.347807e-07	-0.0	-0.0	
175146	1.347807e-07	-0.0	-0.0	
175147	1.347807e-07	-0.0	-0.0	
175148	1.347807e-07	-0.0	-0.0	
24	6_price_peak_fix	6_price_mid_peak_fix	6mps_price_off_peak p1	\
24	-0.0	-0.0	0.000005	
25	-0.0	-0.0	0.000005	
26	-0.0	-0.0	0.000005	
27	-0.0	-0.0	0.000005	
28	-0.0	-0.0	0.000005	
...	...	...	...	
175144	-0.0	-0.0	0.000005	
175145	-0.0	-0.0	0.000005	
175146	-0.0	-0.0	0.000005	
175147	-0.0	-0.0	0.000005	
175148	-0.0	-0.0	0.000005	
24	6mps_price_peak p2	6mps_price_mid_peak p3	churn price_date	\
24	1.350021e-07	-0.0	0 2015-12-01	
25	1.350021e-07	-0.0	0 2015-12-01	
26	1.350021e-07	-0.0	0 2015-12-01	
27	1.350021e-07	-0.0	0 2015-12-01	
28	1.350021e-07	-0.0	0 2015-12-01	
...	...	...	...	
175144	1.350021e-07	-0.0	0 2015-12-01	
175145	1.350021e-07	-0.0	0 2015-12-01	
175146	1.350021e-07	-0.0	0 2015-12-01	
175147	1.350021e-07	-0.0	0 2015-12-01	
175148	1.350021e-07	-0.0	0 2015-12-01	
24	D-J_of_pk_fix	D-J_of_pk_var	mdiff_(peak-offpeak)var	\
24	0.178813	-0.004725	-0.083655	
25	0.178813	-0.004725	-0.083655	
26	0.178813	-0.004725	-0.083655	
27	0.178813	-0.004725	-0.083655	
28	0.178813	-0.004725	-0.083655	

```

...
175144      ...      ...
175145      -0.000001   -0.004682   ...
175146      -0.000001   -0.004682   ...
175147      -0.000001   -0.004682   ...
175148      -0.000001   -0.004682   ...

    mdiff_(offpeak-midpeak)var  mdiff_(mid-peak)var  \
24          0.247128      -0.036089
25          0.247128      -0.036089
26          0.247128      -0.036089
27          0.247128      -0.036089
28          0.247128      -0.036089
...
175144      ...
175145      0.243469      -0.035959
175146      0.243469      -0.035959
175147      0.243469      -0.035959
175148      0.243469      -0.035959

    mdiff_(peak-offpeak)fix  mdiff_(offpeak-midpeak)fix  \
24          -116.555088    793.885635
25          -116.555088    793.885635
26          -116.555088    793.885635
27          -116.555088    793.885635
28          -116.555088    793.885635
...
175144      ...
175145      -116.150237   789.904364
175146      -116.150237   789.904364
175147      -116.150237   789.904364
175148      -116.150237   789.904364

    mdiff_(mid-peak)fix  max_frmpdiff_(peak-offpeak)var  \
24          0.0          -0.075504
25          0.0          -0.075504
26          0.0          -0.075504
27          0.0          -0.075504
28          0.0          -0.075504
...
175144      ...
175145      0.0          -0.075203
175146      0.0          -0.075203
175147      0.0          -0.075203
175148      0.0          -0.075203

max_frmpmdiff_(offpeak-midpeak)var  max_frmpmdiff_(mid-peak)var  \

```

24	0.238161	-0.032558
25	0.238161	-0.032558
26	0.238161	-0.032558
27	0.238161	-0.032558
28	0.238161	-0.032558
...	...	...
175144	0.234750	-0.032470
175145	0.234750	-0.032470
175146	0.234750	-0.032470
175147	0.234750	-0.032470
175148	0.234750	-0.032470
 max_mdiff_(peak-offpeak)fix max_mdiff_(offpeak-midpeak)fix \		
24	-99.797374	1673.962443
25	-99.797374	1673.962443
26	-99.797374	1673.962443
27	-99.797374	1673.962443
28	-99.797374	1673.962443
...	...	...
175144	-99.797371	1659.943479
175145	-99.797371	1659.943479
175146	-99.797371	1659.943479
175147	-99.797371	1659.943479
175148	-99.797371	1659.943479
 max_mdiff_(mid-peak)fix min_frmpdiff_(peak-offpeak)var \		
24	0.0	-0.085300
25	0.0	-0.085300
26	0.0	-0.085300
27	0.0	-0.085300
28	0.0	-0.085300
...	...	...
175144	0.0	-0.083618
175145	0.0	-0.083618
175146	0.0	-0.083618
175147	0.0	-0.083618
175148	0.0	-0.083618
 min_frmpmdiff_(offpeak-midpeak)var min_frmpmdiff_(mid-peak)var \		
24	0.275277	-0.044072
25	0.275277	-0.044072
26	0.275277	-0.044072
27	0.275277	-0.044072
28	0.275277	-0.044072
...	...	...
175144	0.270782	-0.043763
175145	0.270782	-0.043763

```

175146          0.270782          -0.043763
175147          0.270782          -0.043763
175148          0.270782          -0.043763

min_mdiff_(peak-offpeak)fix min_mdiff_(offpeak-midpeak)fix \
24          -231.144085          655.036192
25          -231.144085          655.036192
26          -231.144085          655.036192
27          -231.144085          655.036192
28          -231.144085          655.036192
...
175144          ...          ...
175145          -229.749069          655.036160
175146          -229.749069          655.036160
175147          -229.749069          655.036160
175148          -229.749069          655.036160

min_mdiff_(mid-peak)fix yrs_subscribed Leadtime_renewal
24          0.0          1.587724          1.800734
25          0.0          1.587724          1.800734
26          0.0          1.587724          1.800734
27          0.0          1.587724          1.800734
28          0.0          1.587724          1.800734
...
175144          ...          ...
175145          0.0          1.674061          1.956498
175146          0.0          1.674061          1.956498
175147          0.0          1.674061          1.956498
175148          0.0          1.674061          1.956498

```

[148546 rows x 67 columns]

sns.pairplot(ft\_out)

[118]: file1=ft\_out.to\_csv('cleaned.csv')

[119]: ft\_out.describe().round(1)

```

[119]:      cons_12m  cons_gas_12m  cons_last_month  forecast_cons_12m \
count    148546.0      148546.0      148546.0      148546.0
mean      11.2          0.3          6.0          21.5
std       2.3          0.7          4.8          7.4
min       3.4         -0.0          0.0          0.0
25%       9.7          0.0          0.0         16.3
50%      10.8         -0.0          7.5         20.9
75%      12.2         -0.0          9.3         26.3
max      18.8          1.9         16.6         44.9

```

	forecast_cons_year	forecast_discount_energy	forecast_meter_rent_12m	\		
count	148546.0	148546.0	148546.0			
mean	4.7	0.4	5.0			
std	4.2	3.0	2.2			
min	0.0	0.0	0.0			
25%	0.0	0.0	3.7			
50%	6.5	0.0	3.9			
75%	8.6	0.0	7.6			
max	13.4	30.0	11.6			
	forecast_price_energy_off_peak	forecast_price_energy_peak	\			
count	148546.0	148546.0				
mean	0.1	0.0				
std	0.0	0.0				
min	0.1	0.0				
25%	0.1	0.0				
50%	0.1	0.1				
75%	0.2	0.1				
max	0.2	0.1				
	forecast_price_pow_off_peak	imp_cons	margin_gross_pow_ele	\		
count	148546.0	148546.0	148546.0			
mean	1614.9	2.9	5.1			
std	154.7	2.6	1.6			
min	1073.6	-0.0	0.0			
25%	1418.6	0.0	4.3			
50%	1704.5	3.7	5.3			
75%	1704.5	5.2	6.1			
max	1999.7	8.9	10.8			
	margin_net_pow_ele	nb_prod_act	net_margin	num_years_antig	pow_max	\
count	148546.0	148546.0	148546.0	148546.0	148546.0	
mean	5.1	0.1	8.0	1.2	0.7	
std	1.6	0.0	2.8	0.1	0.0	
min	0.0	0.1	0.0	1.0	0.7	
25%	4.3	0.1	6.1	1.1	0.7	
50%	5.3	0.1	7.9	1.2	0.7	
75%	6.1	0.1	9.9	1.3	0.8	
max	10.8	0.1	16.2	1.5	0.8	
	y_price_off_peak_var	y_price_peak_var	y_price_mid_peak_var	\		
count	148546.0	148546.0	148546.0			
mean	0.0	0.0	0.0			
std	0.0	0.0	0.0			
min	-0.0	-0.0	-0.0			
25%	0.0	0.0	0.0			
50%	0.0	0.0	-0.0			

75%	0.0	0.0	0.0
max	0.0	0.0	0.0
count	148546.0	148546.0	148546.0
mean	0.0	0.0	0.0
std	0.0	0.0	0.0
min	-0.0	-0.0	-0.0
25%	0.0	0.0	0.0
50%	0.0	-0.0	-0.0
75%	0.0	0.0	0.0
max	0.1	0.0	0.0
count	148546.0	148546.0	148546.0
mean	0.0	0.0	0.0
std	0.0	0.0	0.0
min	-0.0	-0.0	-0.0
25%	0.0	0.0	0.0
50%	0.0	0.0	-0.0
75%	0.0	0.0	0.0
max	0.1	0.0	0.0
count	148546.0	148546.0	148546.0
mean	0.0	0.0	0.0
std	0.0	0.0	0.0
min	-0.0	-0.0	-0.0
25%	0.0	0.0	0.0
50%	0.0	0.0	-0.0
75%	0.0	0.0	0.0
max	0.1	0.0	0.0
count	148546.0	148546.0	148546.0
mean	0.0	0.0	0.0
std	0.0	0.0	0.0
min	-0.0	-0.0	-0.0
25%	0.0	0.0	0.0
50%	0.0	0.0	-0.0
75%	0.0	0.0	0.0
max	0.0	0.0	0.0
count	148546.0	148546.0	148546.0
mean	0.0	0.0	0.0
std	0.0	0.0	0.0
min	-0.0	-0.0	-0.0
25%	0.0	0.0	0.0
50%	-0.0	-0.0	-0.0
75%	0.0	-0.0	-0.0
max	0.1	0.1	0.0
count	148546.0	148546.0	148546.0
mean	0.0	0.0	0.0
std	0.0	0.0	0.0

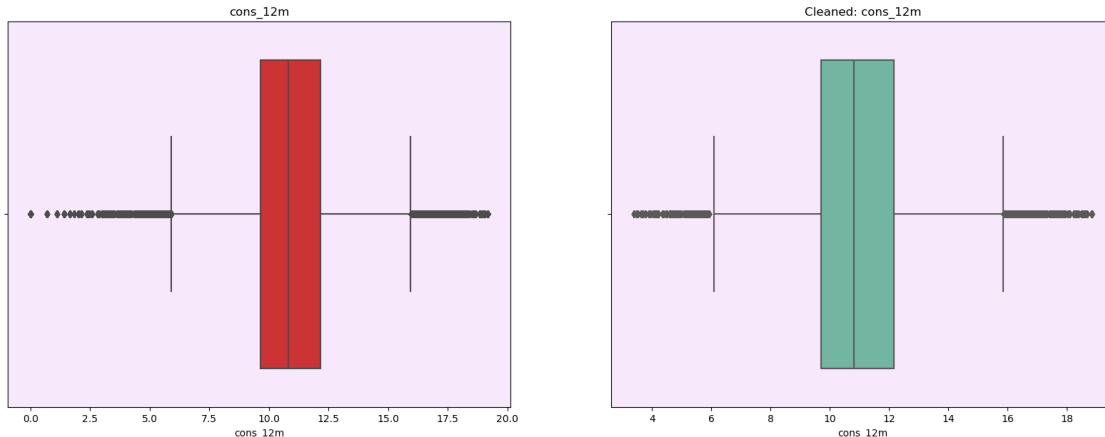
min	-0.0	-0.0	-0.0
25%	0.0	0.0	0.0
50%	0.0	0.0	-0.0
75%	0.0	0.0	0.0
max	0.1	0.1	0.0
count	148546.0	148546.0	148546.0
mean	0.1	0.2	-0.0
std	0.3	0.4	0.0
min	0.0	-1.4	-0.0
25%	0.0	0.0	-0.0
50%	0.0	0.2	-0.0
75%	0.0	0.2	-0.0
max	1.0	2.5	0.0
count	148546.0	148546.0	148546.0
mean	0.2	-0.0	
std	0.1	0.0	
min	0.0	-0.0	
25%	0.1	-0.0	
50%	0.2	-0.0	
75%	0.2	0.0	
max	0.3	0.0	
count	148546.0	148546.0	148546.0
mean	-85.0	594.7	
std	41.0	258.6	
min	-121.9	0.0	
25%	-116.5	260.0	
50%	-116.2	790.4	
75%	-31.9	793.2	
max	0.0	847.2	
count	148546.0	148546.0	148546.0
mean	-0.4	-0.1	
std	0.6	0.1	
min	-1.2	-0.2	
25%	-1.2	-0.1	
50%	0.0	-0.1	
75%	0.0	-0.0	
max	0.0	0.0	
count	148546.0	148546.0	148546.0
mean	0.0	0.0	
std	0.0	0.0	
min	0.0	0.0	
25%	0.0	0.0	
50%	0.0	0.0	
75%	0.0	0.0	
max	0.0	0.0	

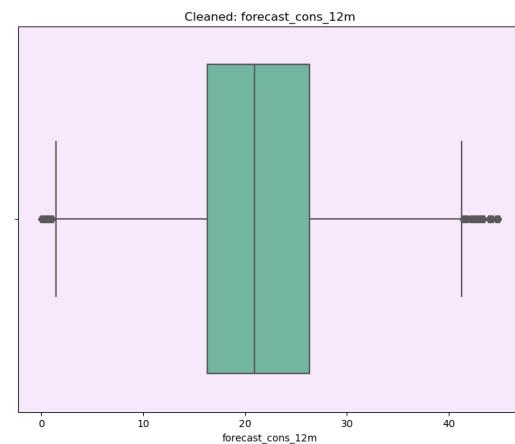
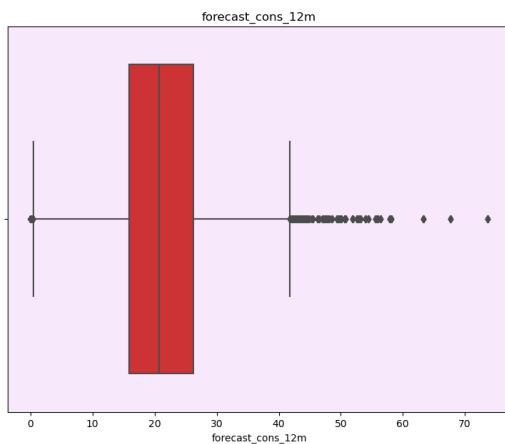
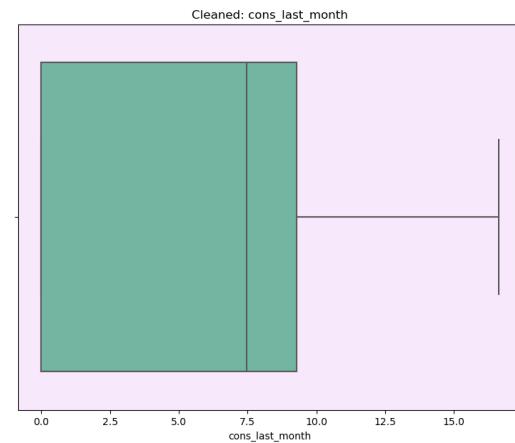
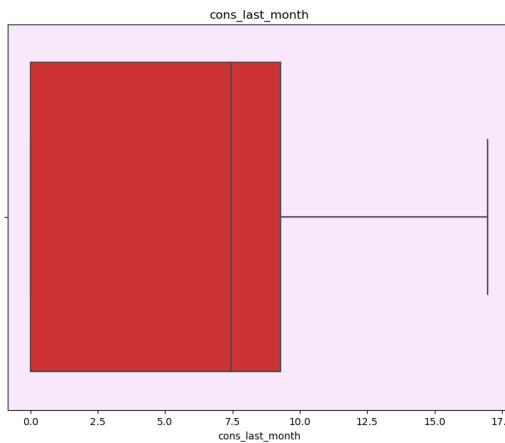
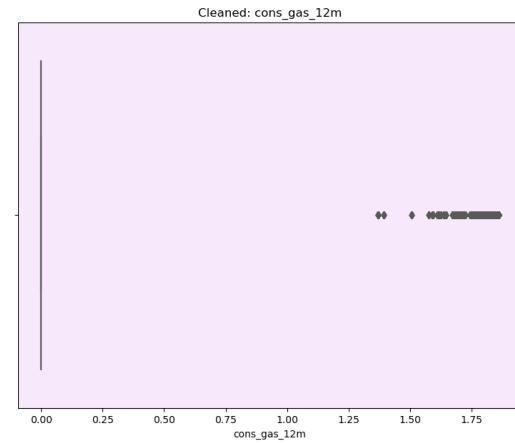
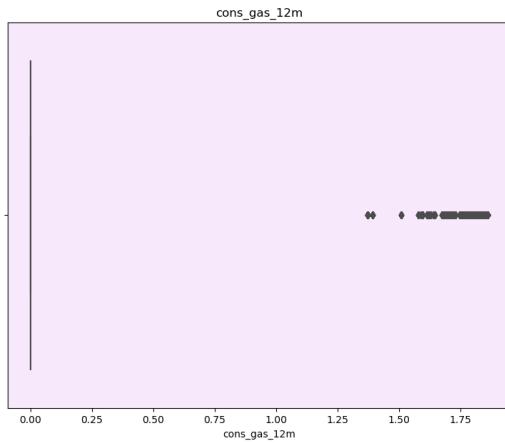
count	148546.0	148546.0
mean	0.2	-0.0
std	0.1	0.0
min	0.0	-0.0
25%	0.1	-0.0
50%	0.2	-0.0
75%	0.2	0.0
max	0.3	0.0
 max_mdiff_(peak-offpeak)fix max_mdiff_(offpeak-midpeak)fix \		
count	148546.0	148546.0
mean	-73.4	1237.5
std	34.5	584.3
min	-104.6	0.0
25%	-99.8	481.8
50%	-99.8	1674.0
75%	-28.6	1674.0
max	0.0	1962.8
 max_mdiff_(mid-peak)fix min_frmpdiff_(peak-offpeak)var \		
count	148546.0	148546.0
mean	-0.4	-0.1
std	0.5	0.1
min	-1.0	-0.2
25%	-1.0	-0.2
50%	0.0	-0.1
75%	0.0	-0.0
max	0.0	0.0
 min_frmpmdiff_(offpeak-midpeak)var min_frmpmdiff_(mid-peak)var \		
count	148546.0	148546.0
mean	0.2	-0.0
std	0.1	0.0
min	0.0	-0.0
25%	0.1	-0.0
50%	0.2	-0.0
75%	0.2	0.0
max	0.4	0.0
 min_mdiff_(peak-offpeak)fix min_mdiff_(offpeak-midpeak)fix \		
count	148546.0	148546.0
mean	-165.1	494.3
std	87.5	210.3
min	-259.2	0.0
25%	-231.1	221.7
50%	-231.1	655.0
75%	-51.7	655.0

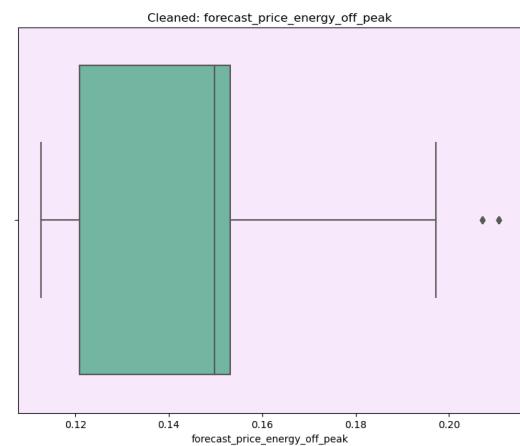
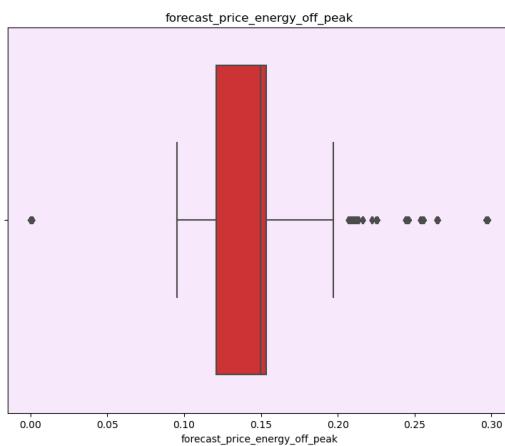
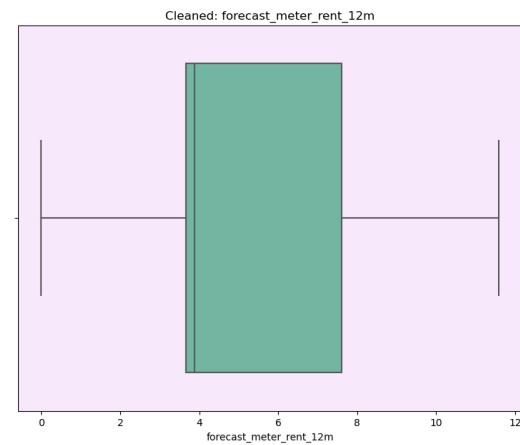
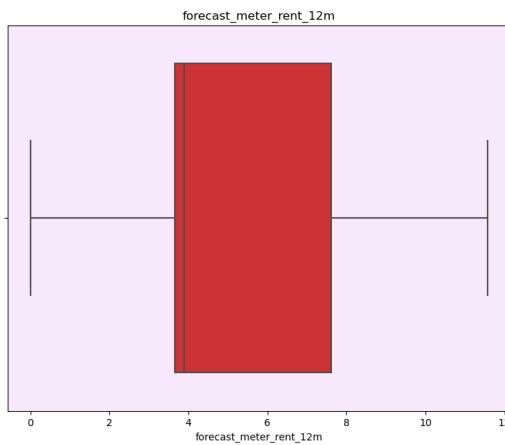
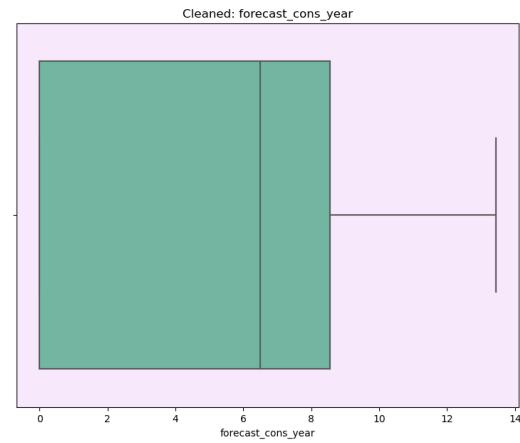
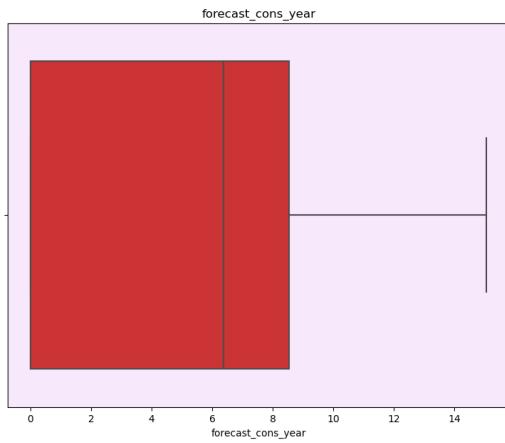
max	0.0	701.1	
count	148546.0	148546.0	148546.0
mean	-0.4	1.5	1.7
std	0.6	0.2	0.3
min	-1.2	1.2	1.1
25%	-1.2	1.4	1.4
50%	0.0	1.5	1.6
75%	0.0	1.6	1.8
max	0.0	2.0	2.5

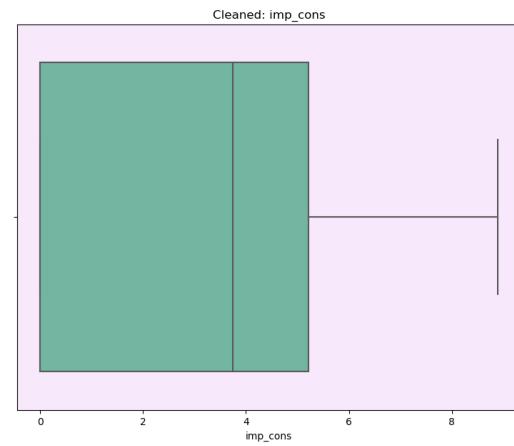
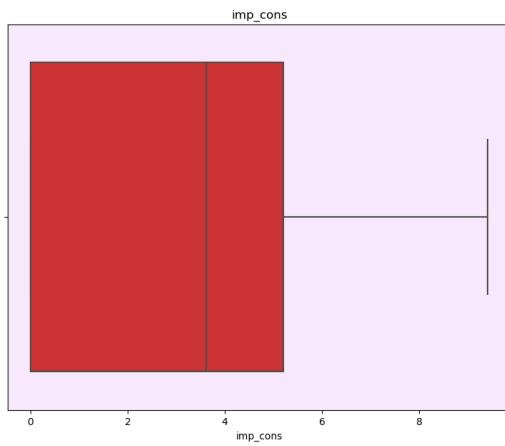
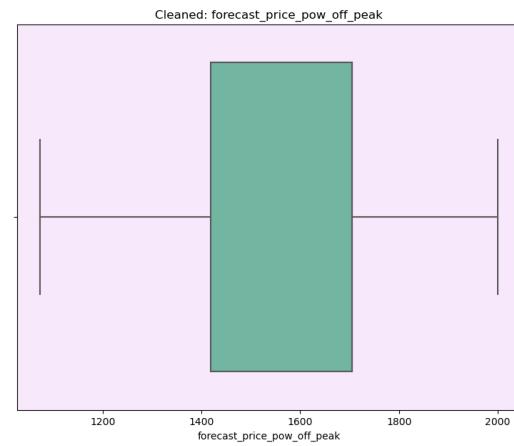
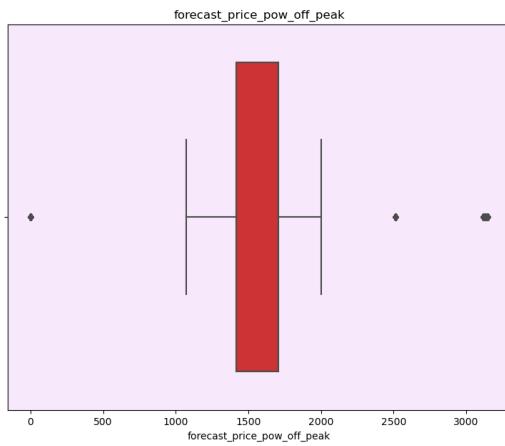
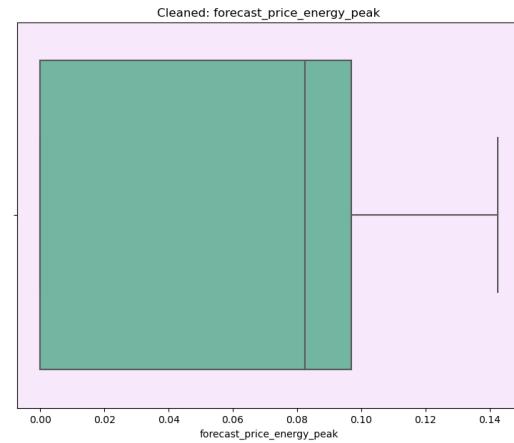
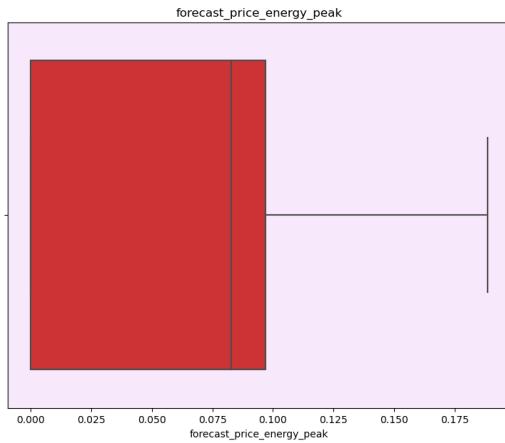
## 7 DATA VISUALIZATION (Before / After Transformations)

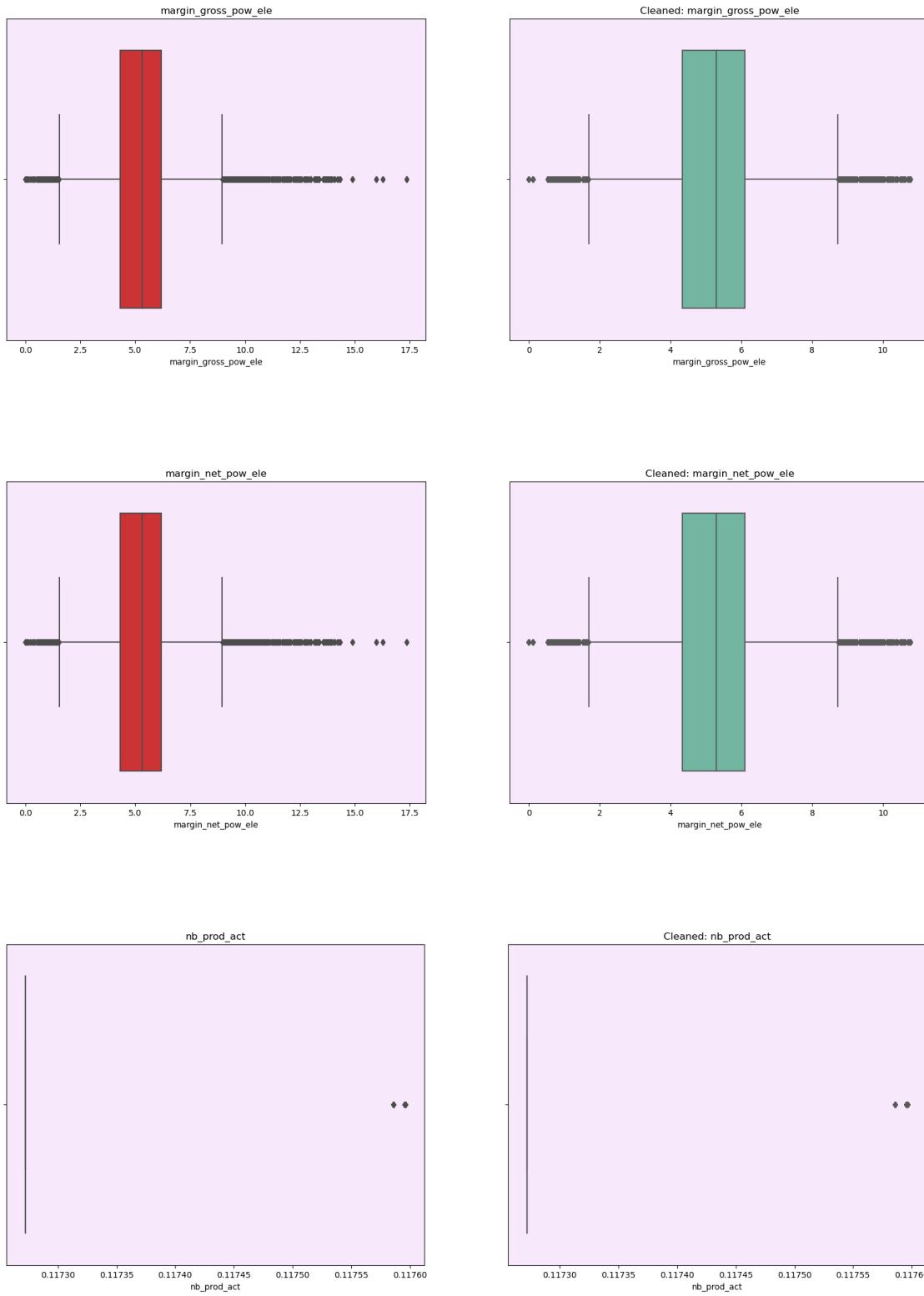
```
[128]: #after
for i in c:
    fig,axs=plt.subplots(1,2,figsize=(20,7))
    a=sns.boxplot(data=ft,x=i,palette='Set1',ax=axs[0])
    a.set_title(f"{i}")
    b=sns.boxplot(data=ft_out,x=i,palette='Set2',ax=axs[1])
    b.set_title(f"Cleaned: {i}")
    plt.show()
```

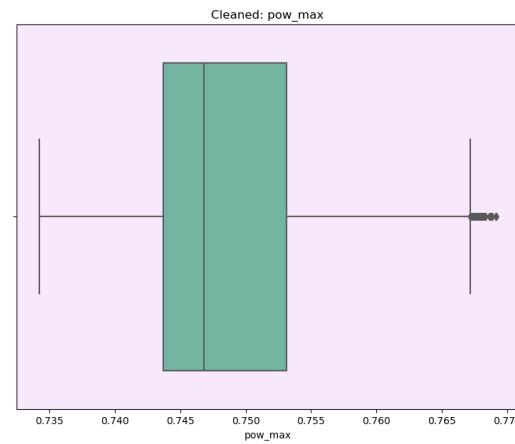
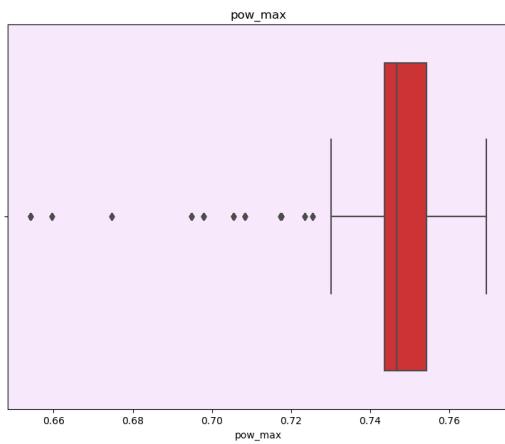
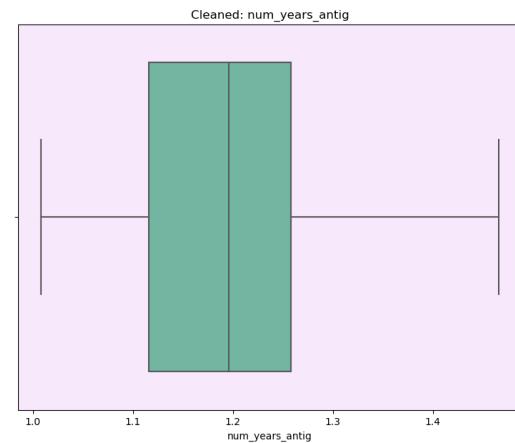
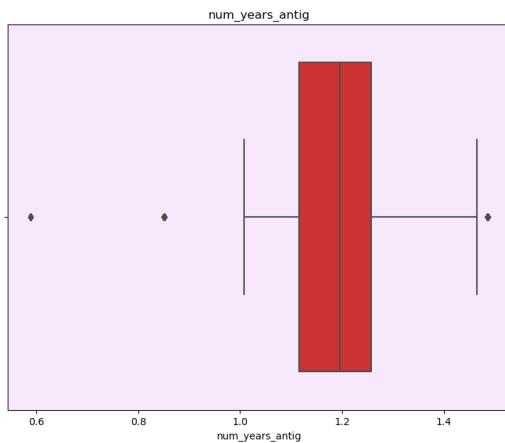
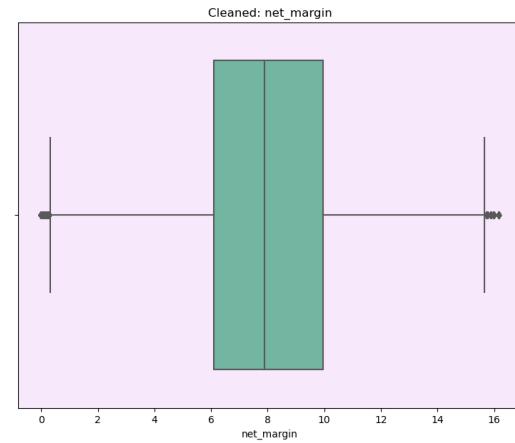
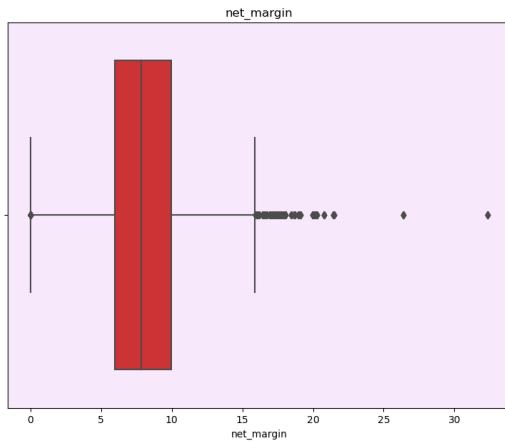


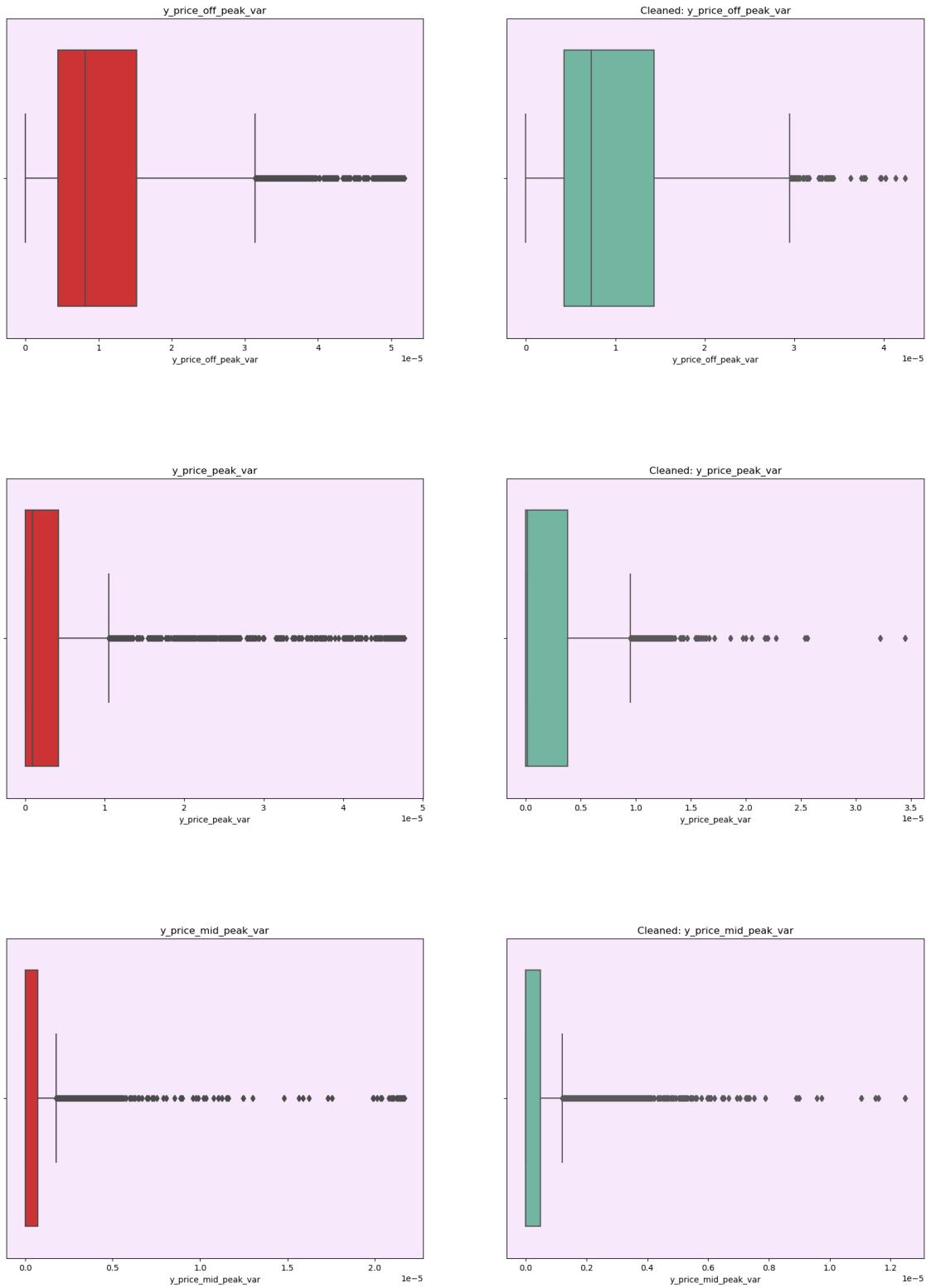


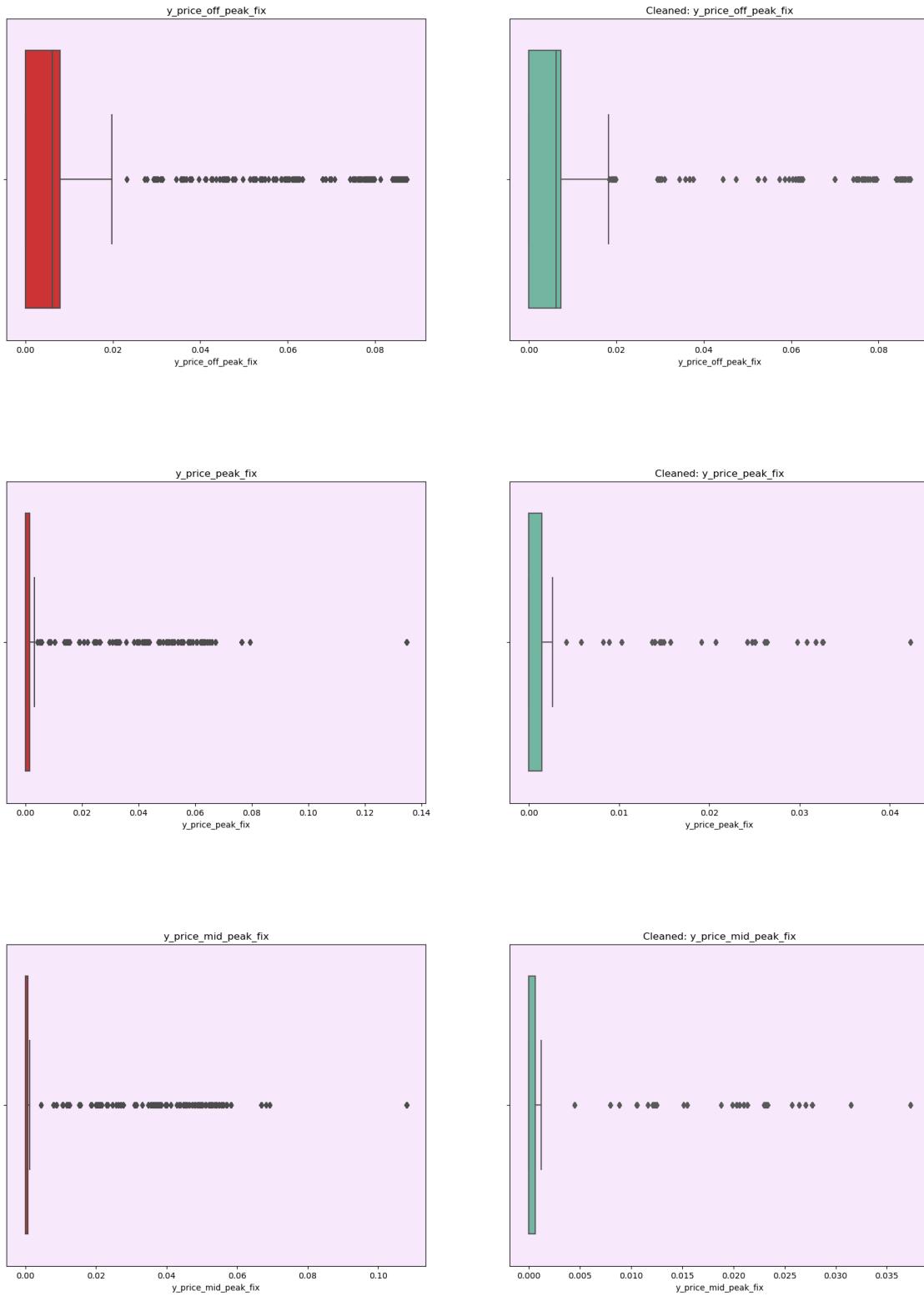


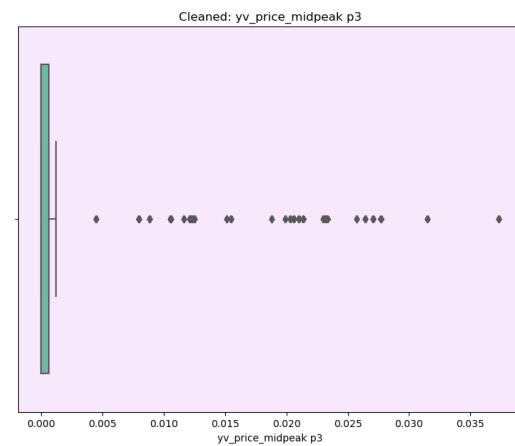
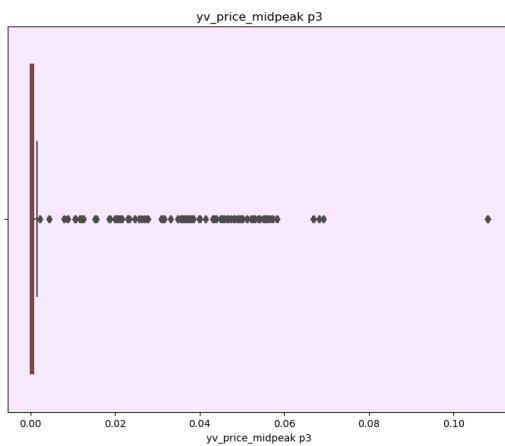
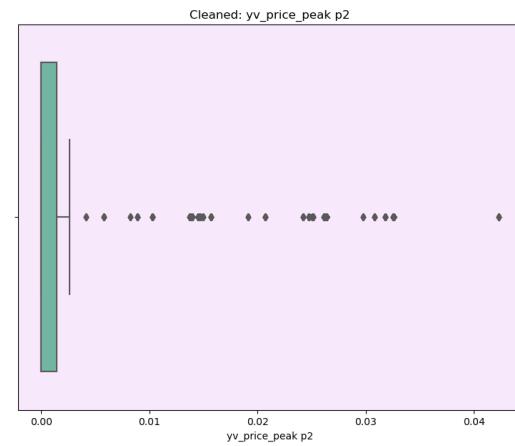
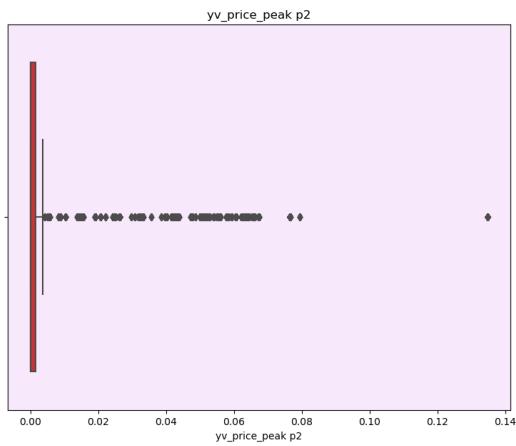
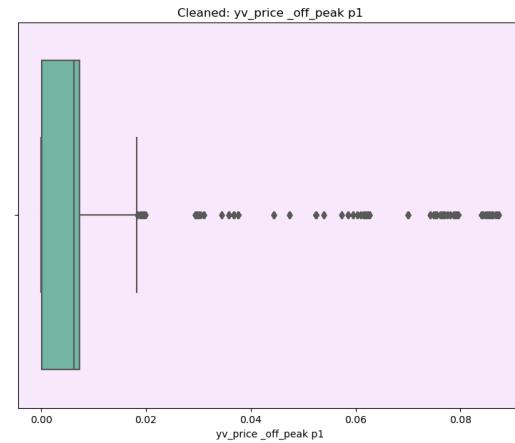
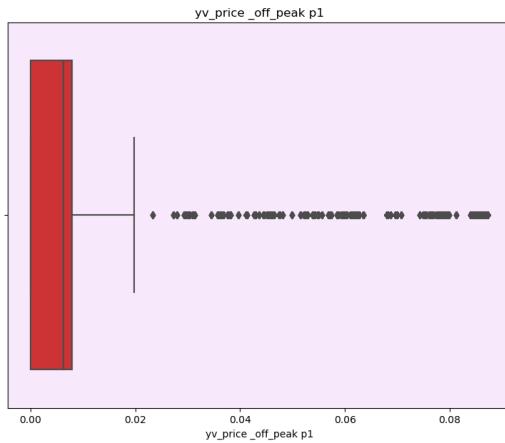


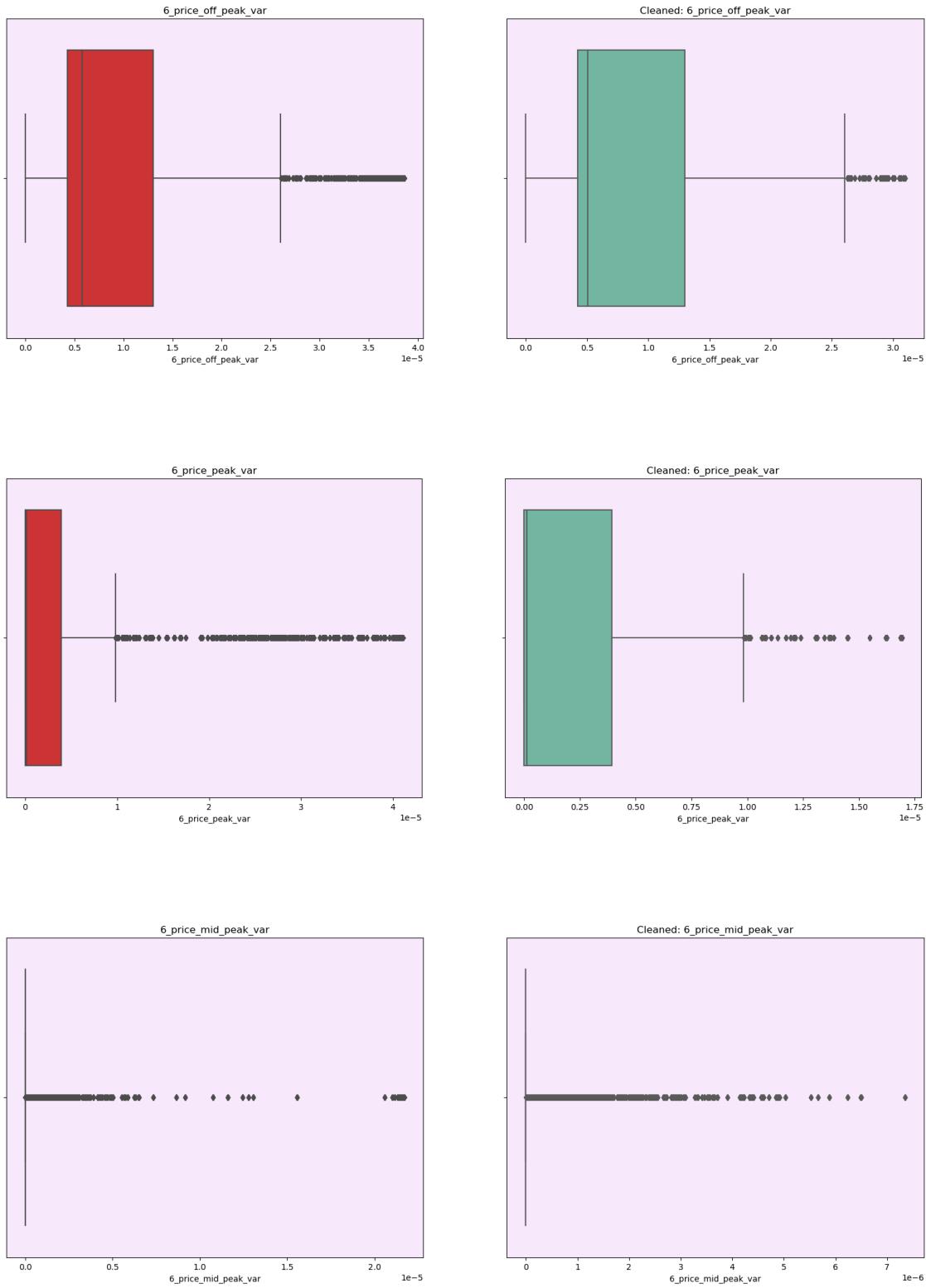


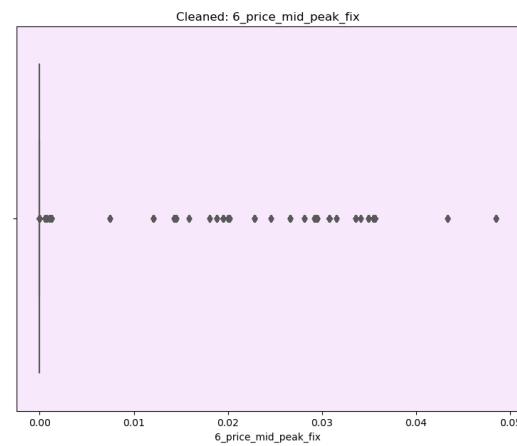
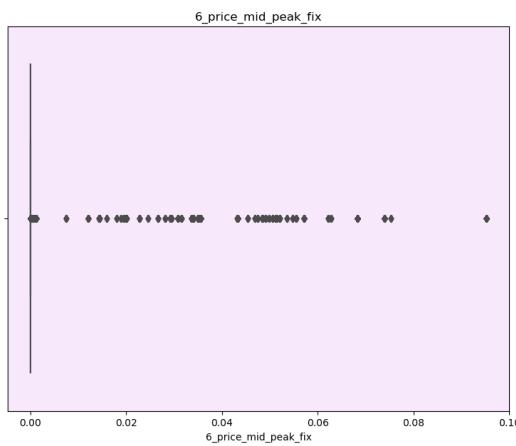
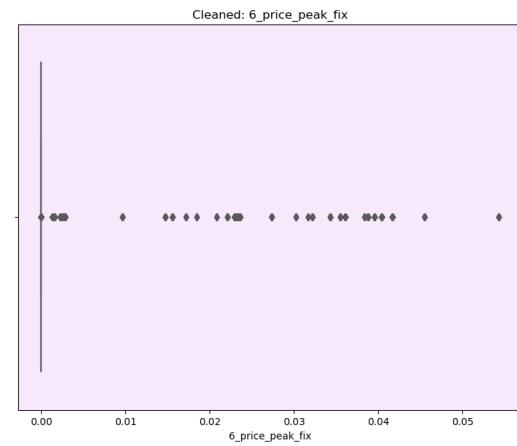
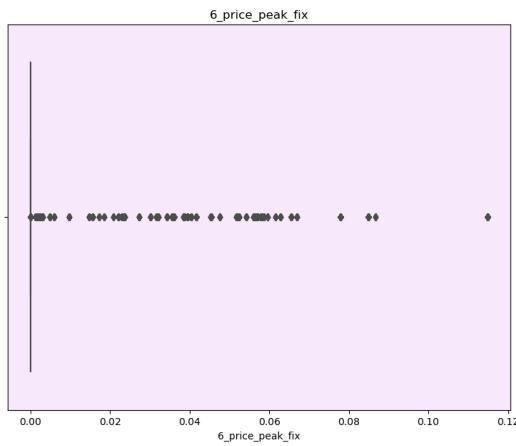
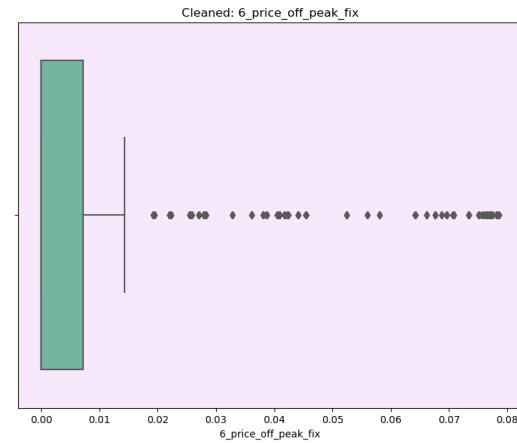
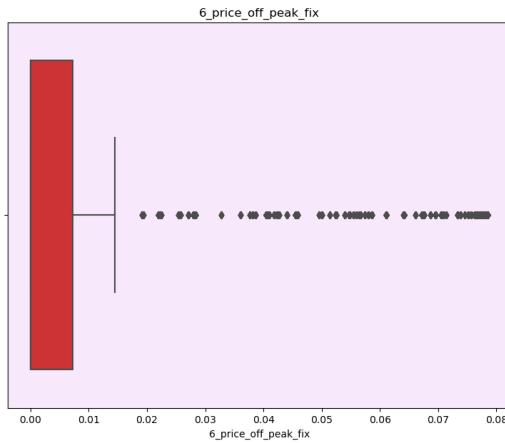


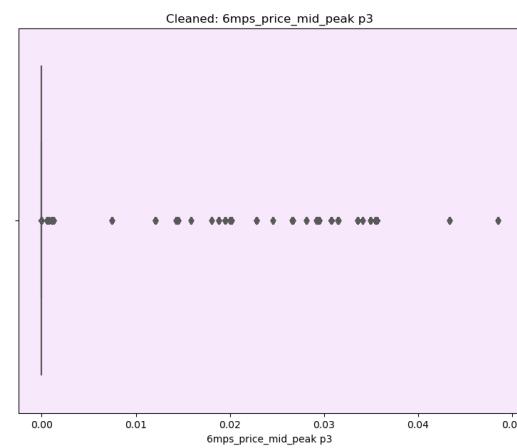
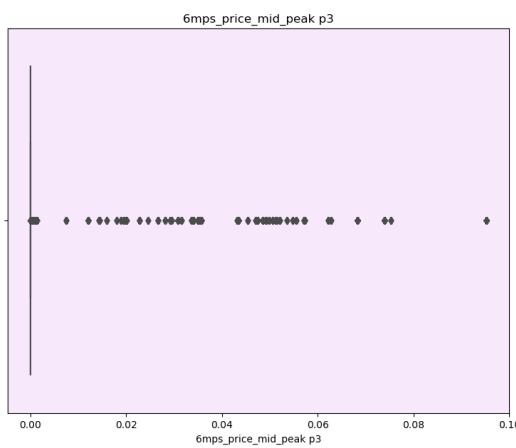
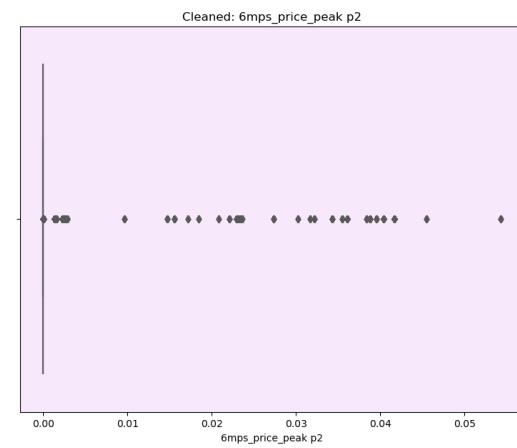
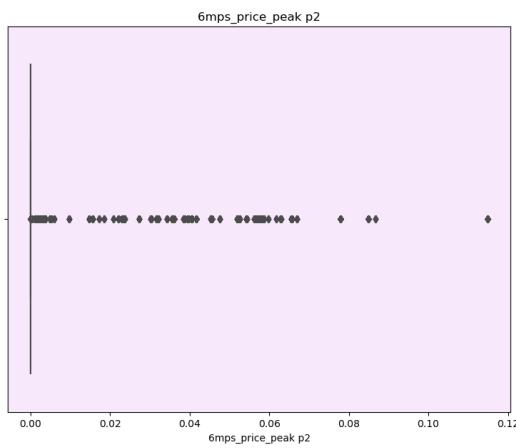
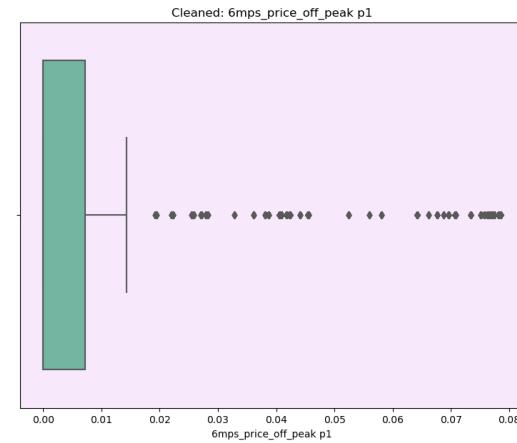
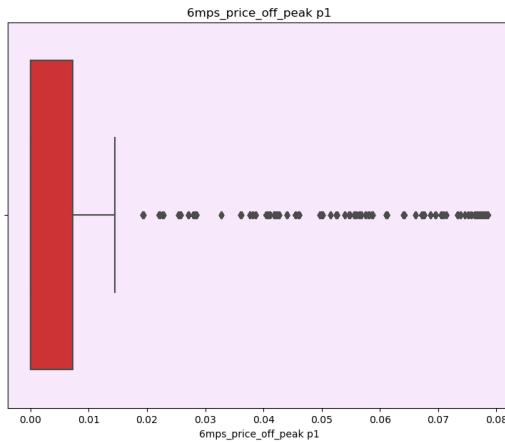


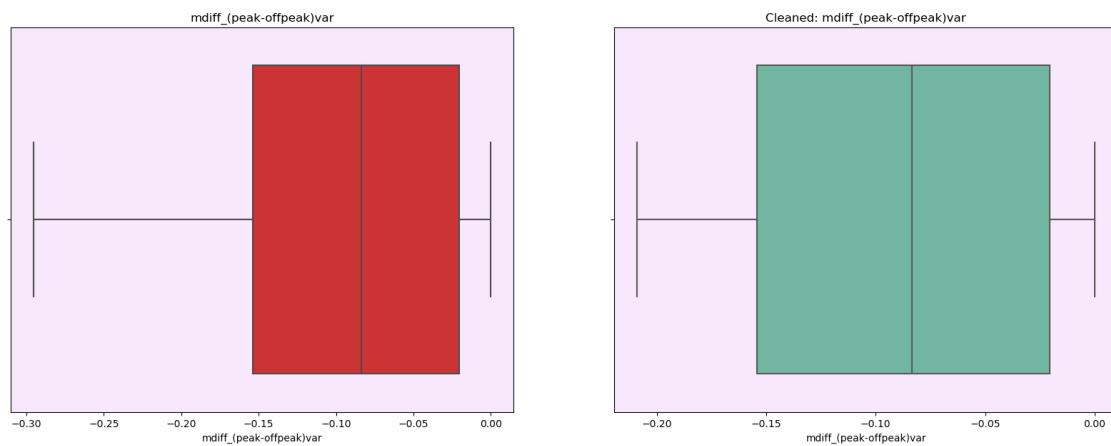
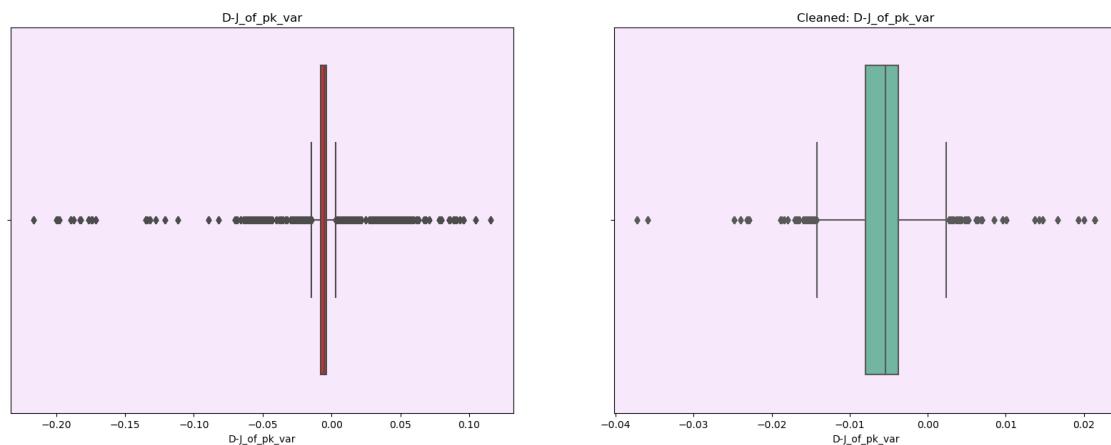
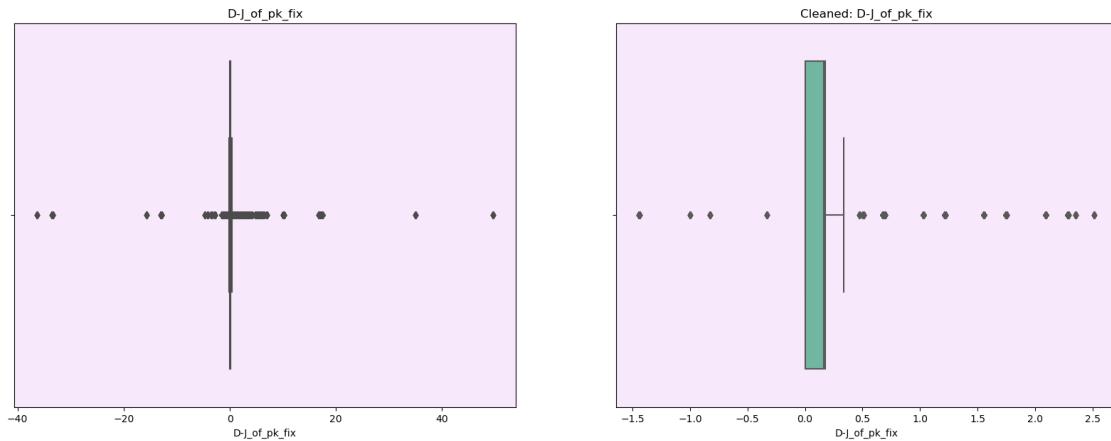


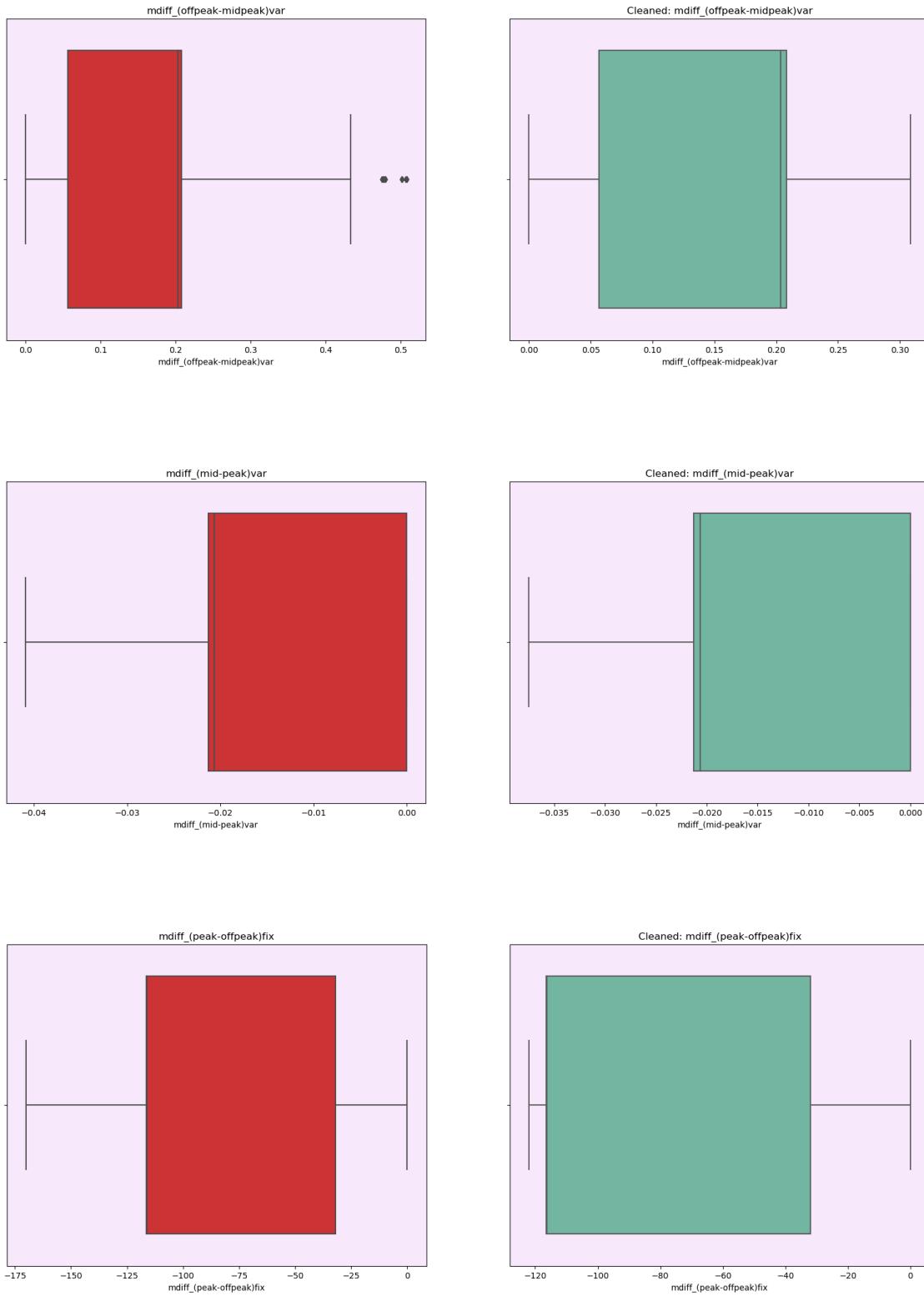


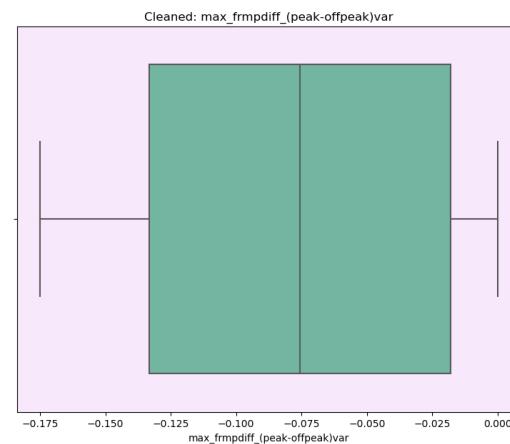
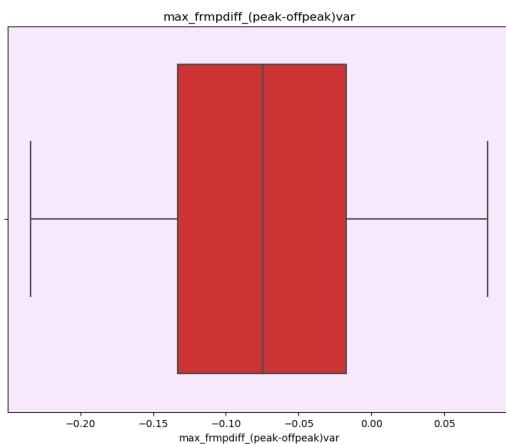
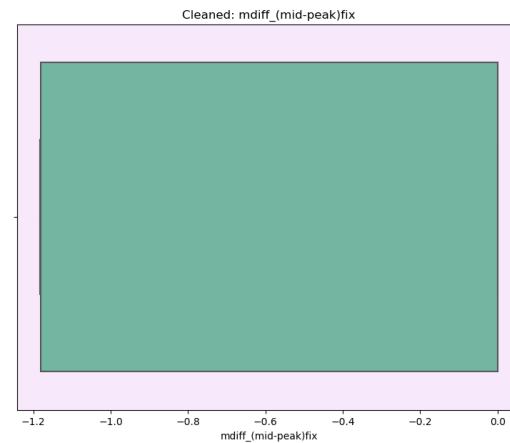
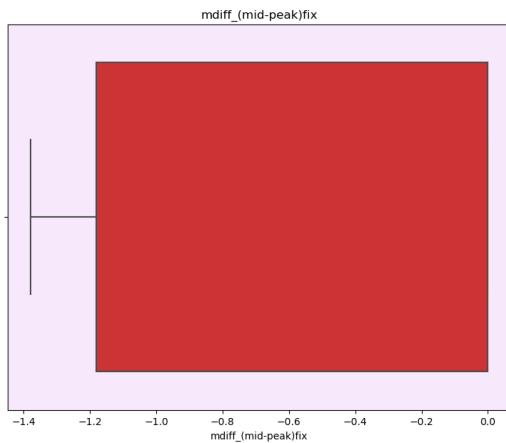
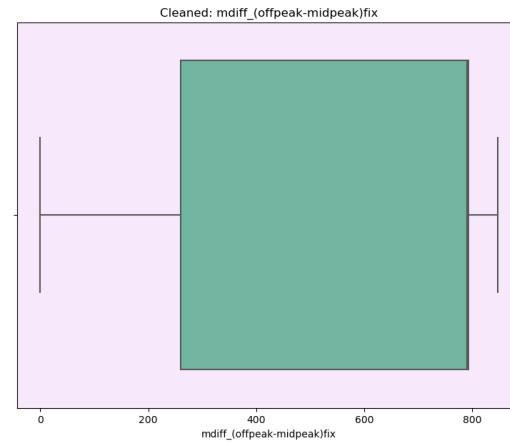
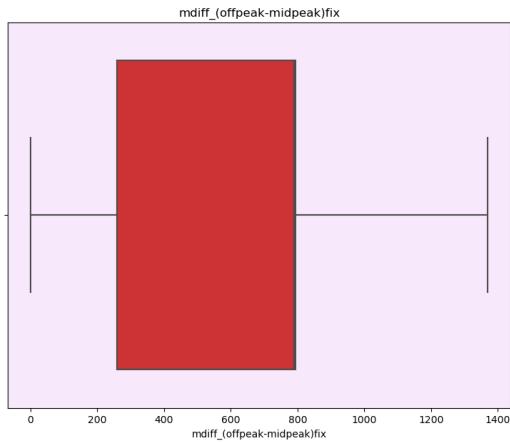


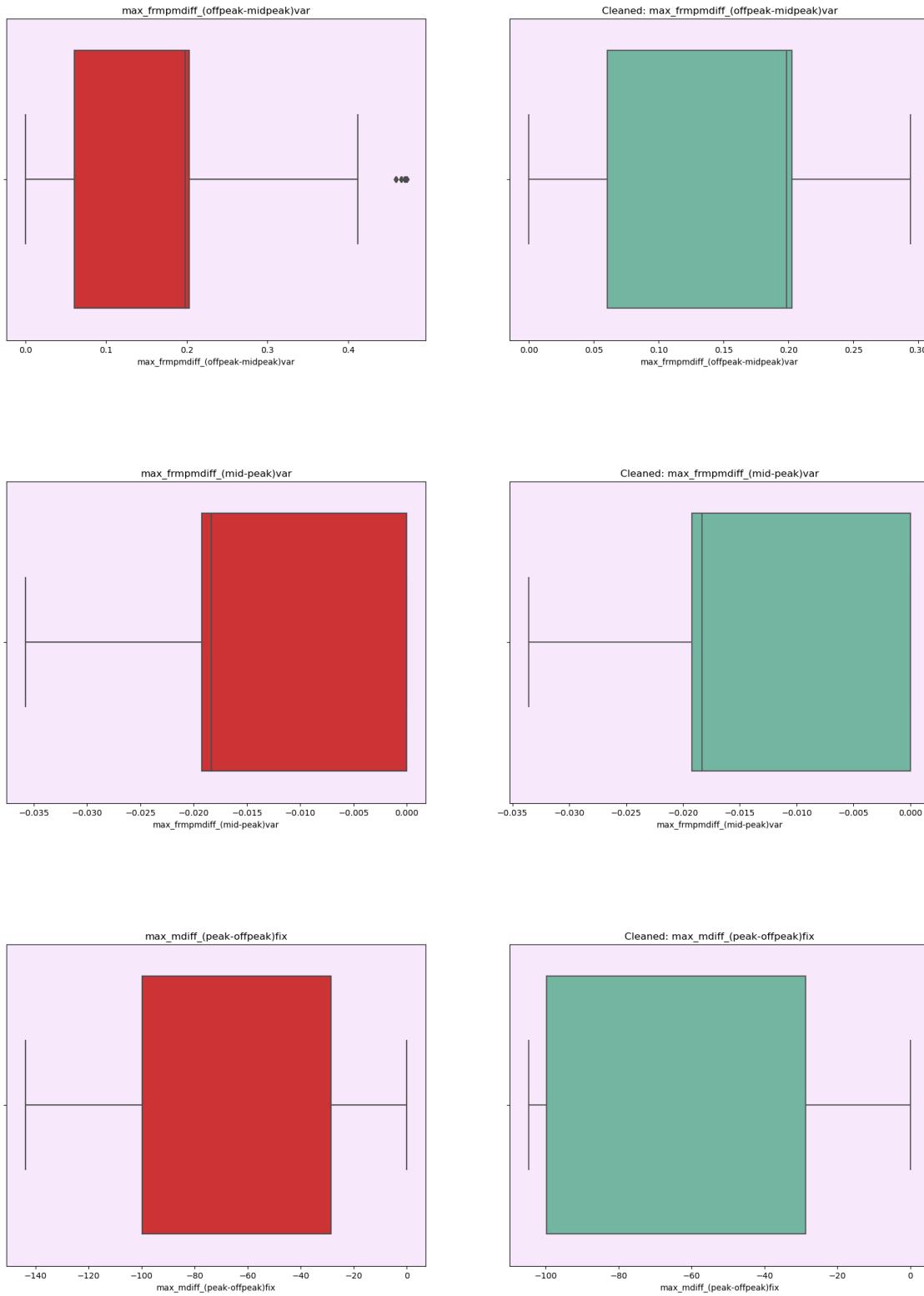


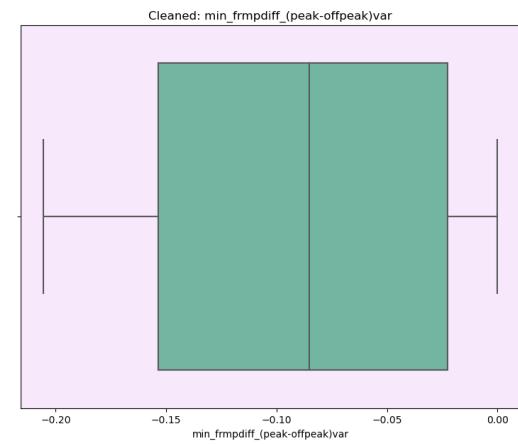
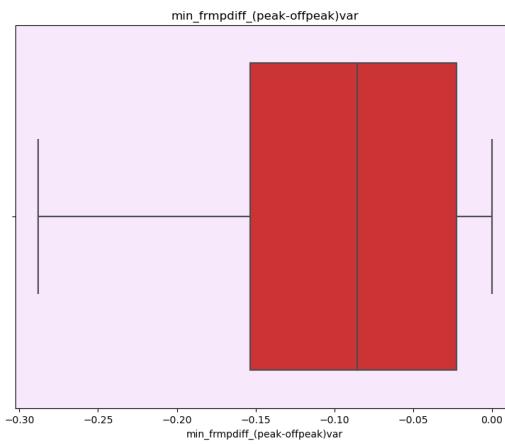
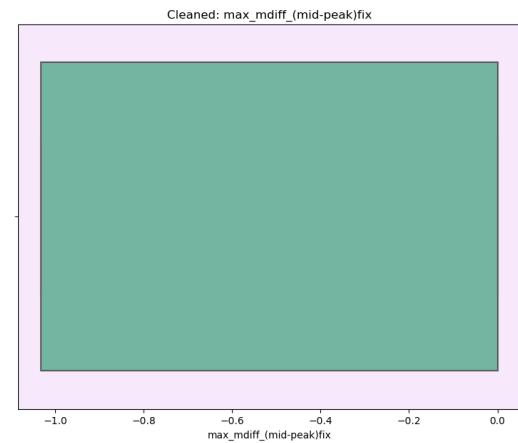
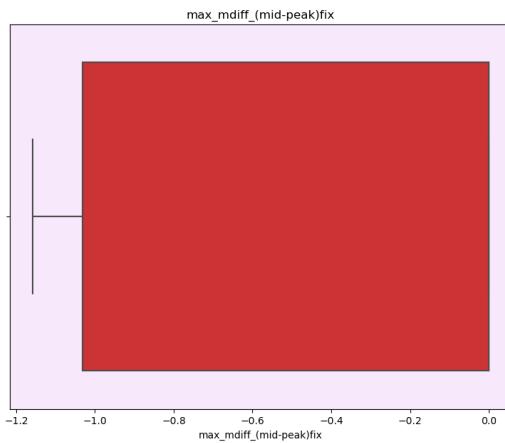
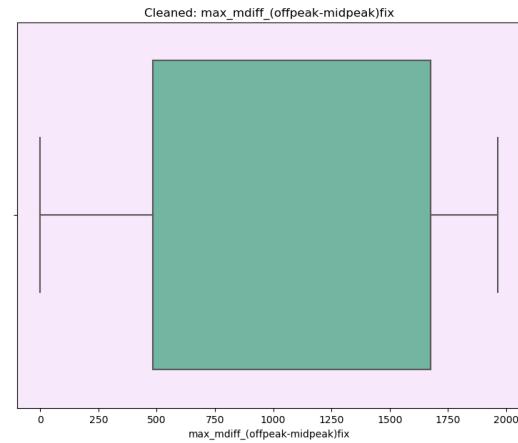
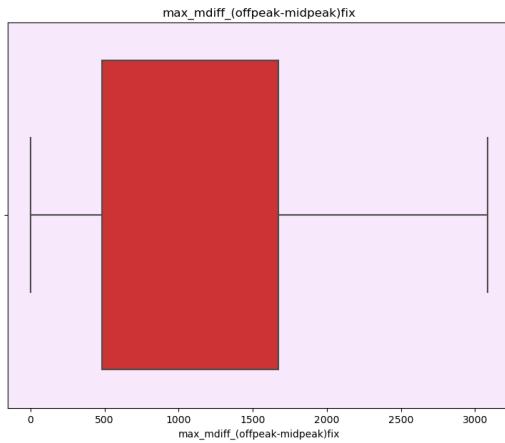


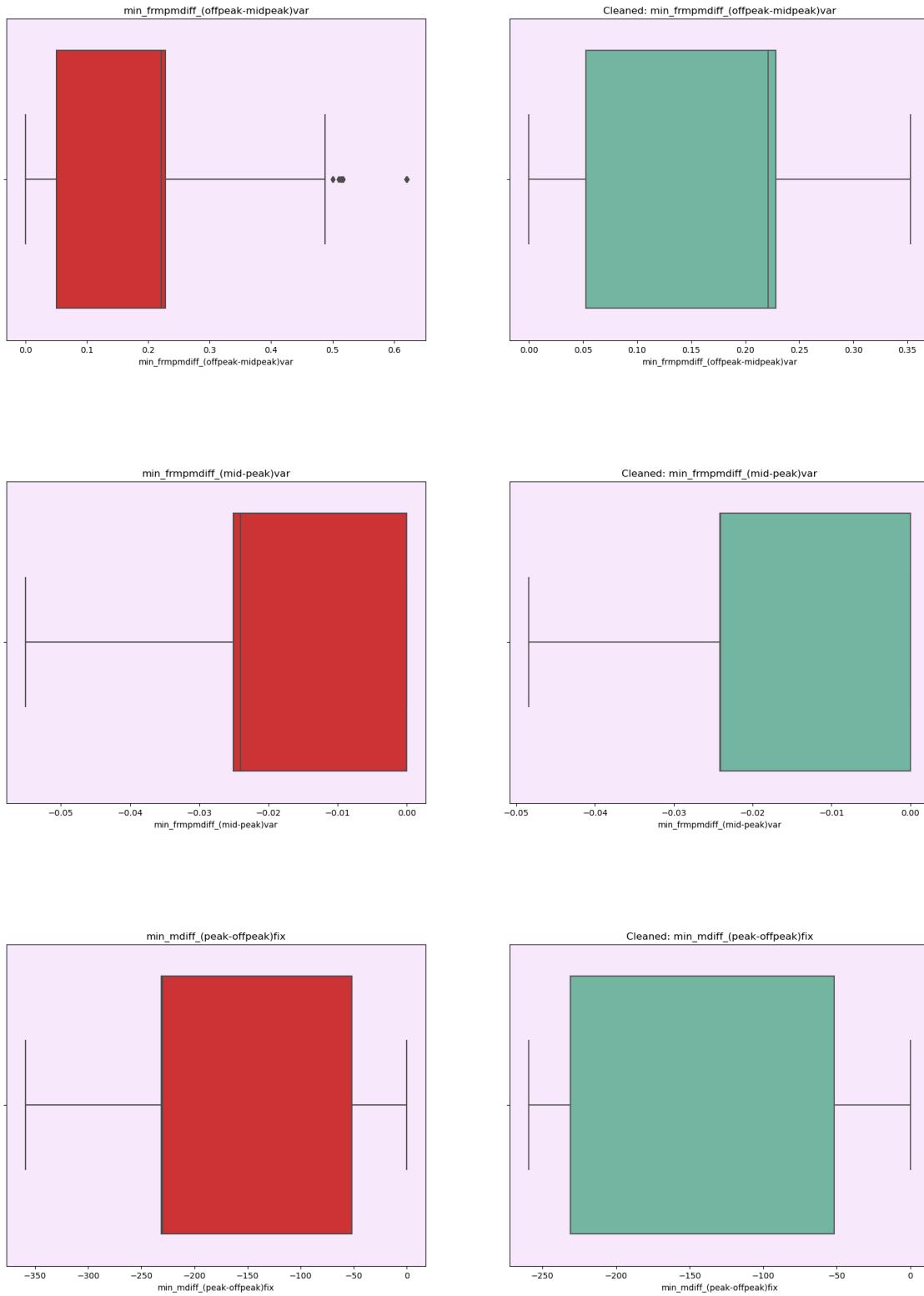


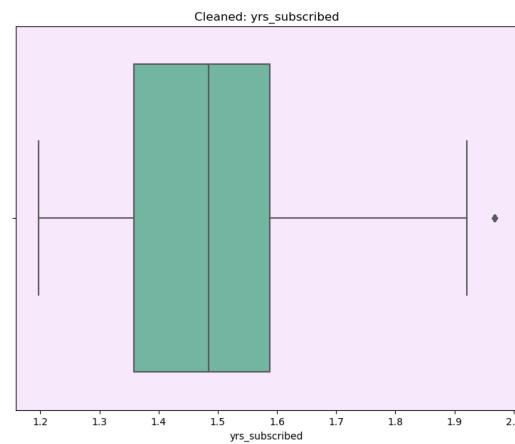
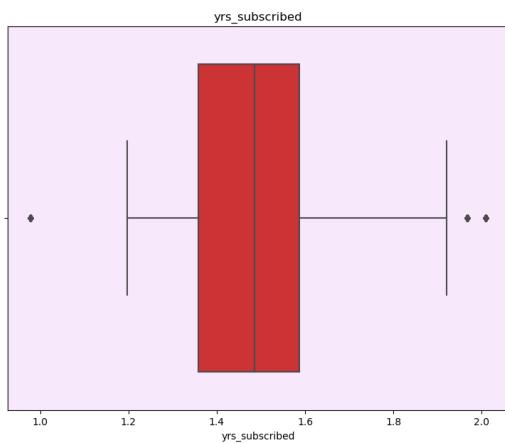
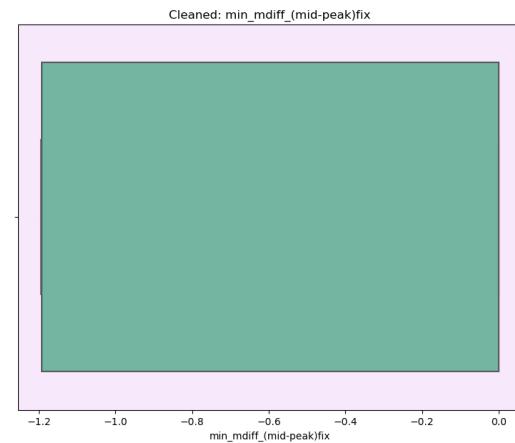
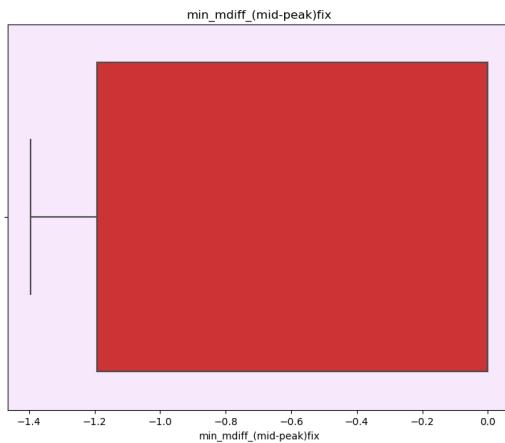
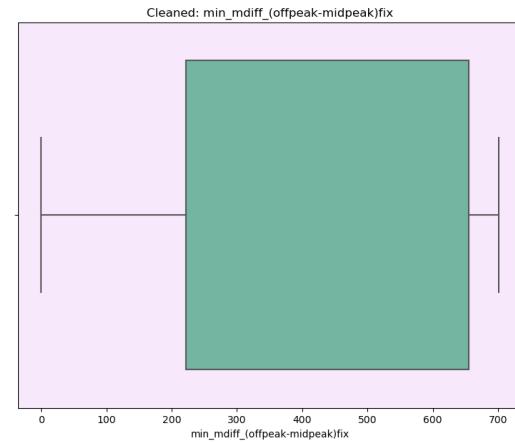
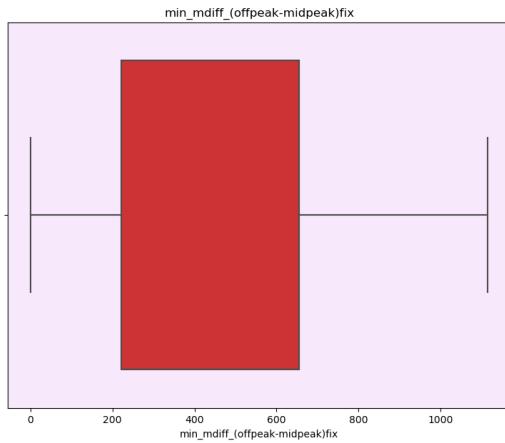


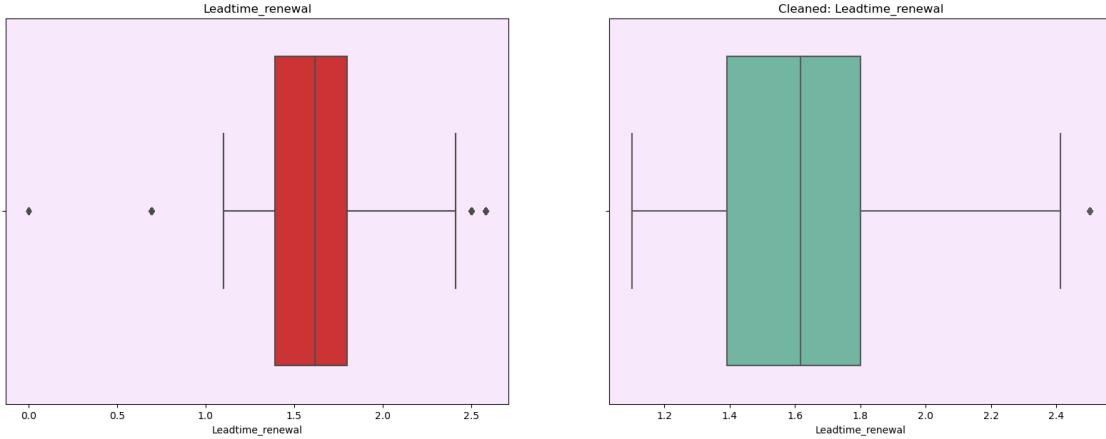










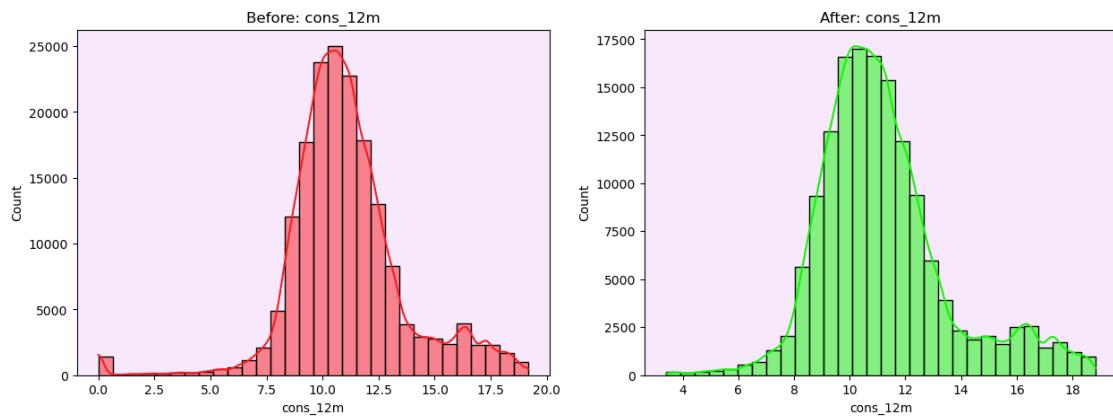


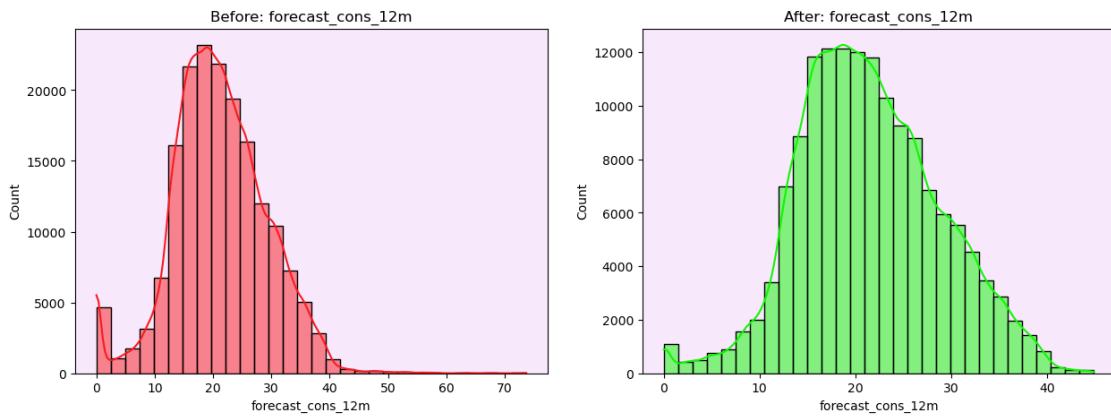
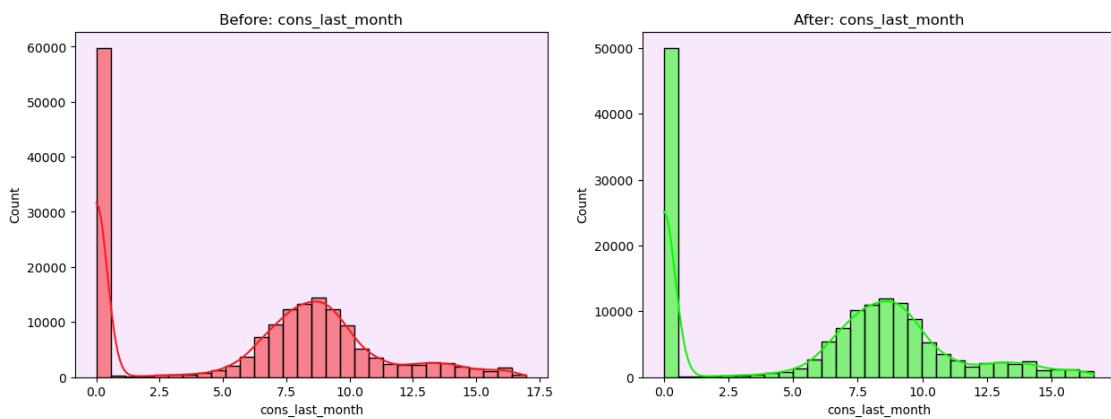
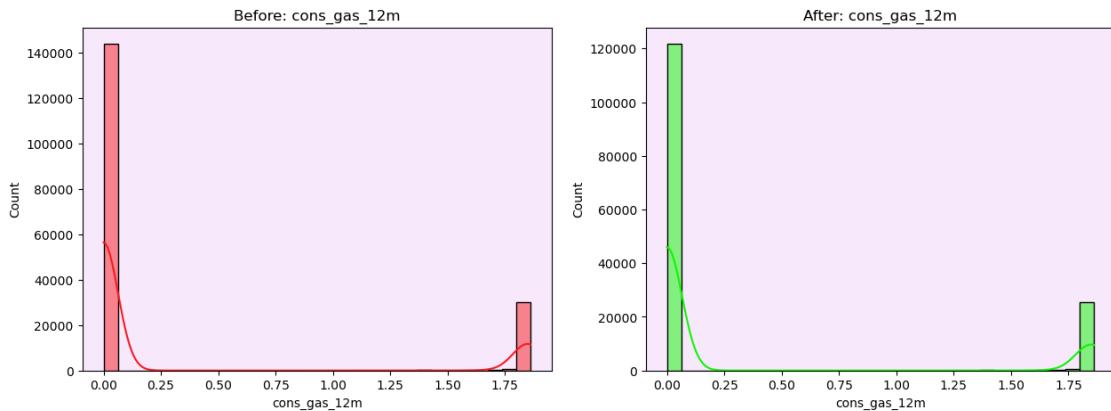
```
[121]: def transfoplots():
    for i in c:

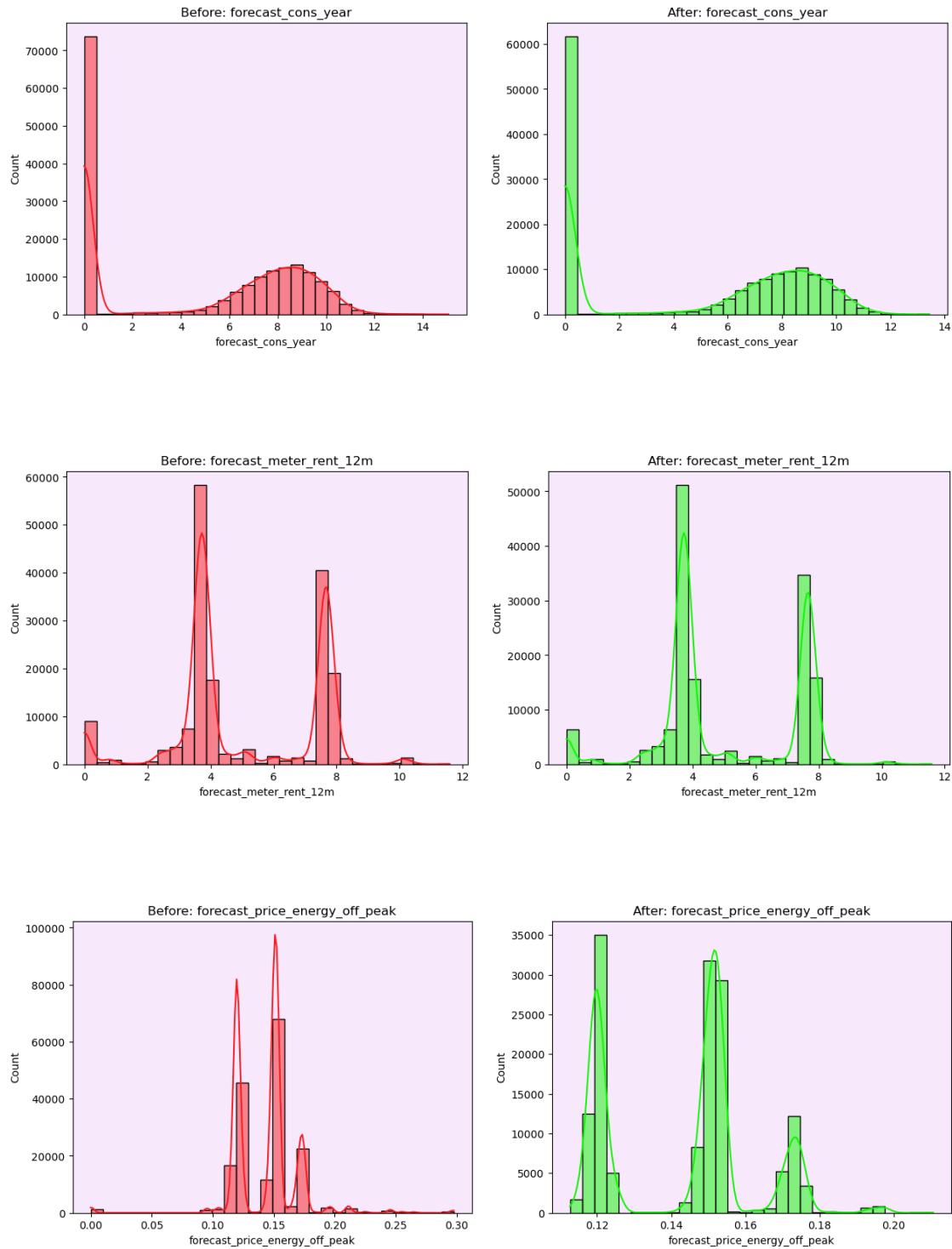
        fig,axs=plt.subplots(1,2,figsize=(15,5))
        a=sns.histplot(x=ft[i],ax=axs[0],kde=True,bins=30,color="#f71c20")
        a.set_title(f"Before: {i}")
        a.set_xlabel(f"{i}")

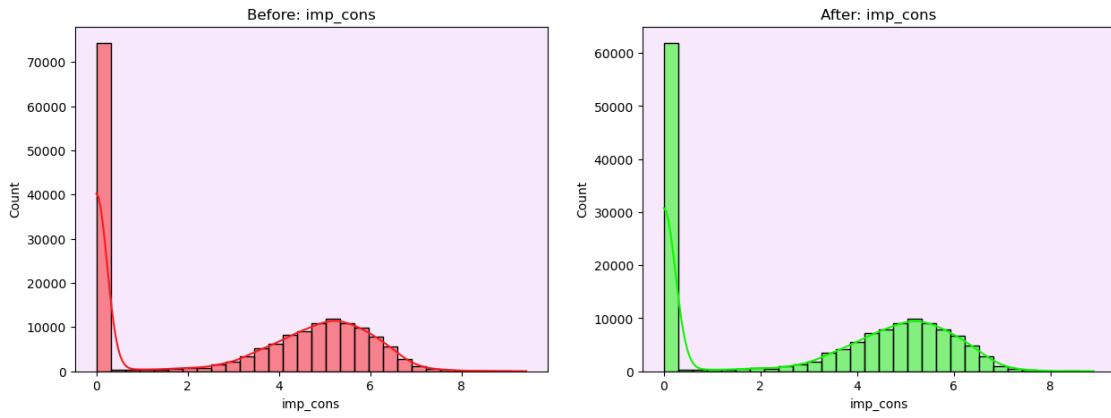
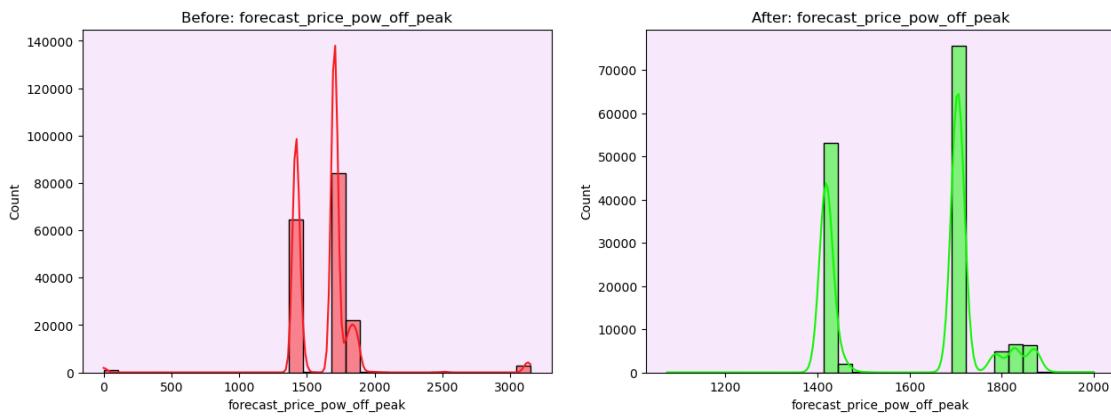
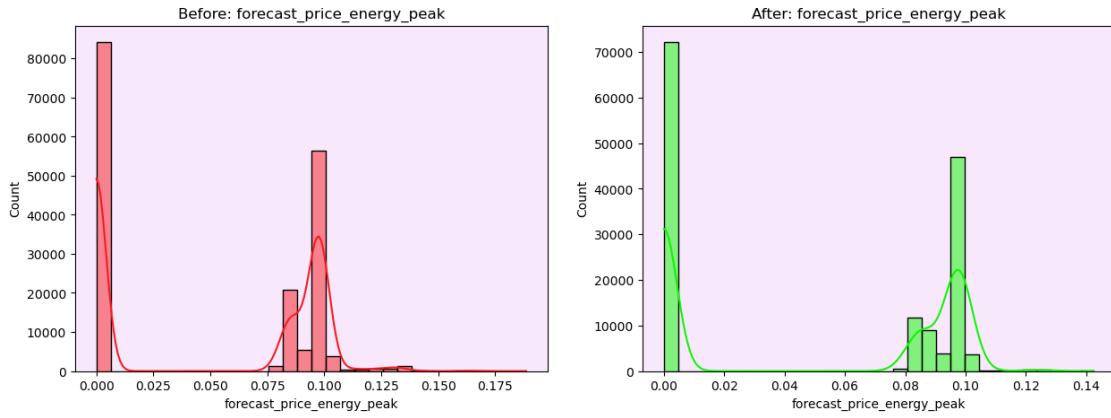
        b=sns.histplot(x=ft_out[i],ax=axs[1],kde=True,bins=30,color="#0ff701")
        b.set_title(f"After: {i}")
        b.set_xlabel(f"{i}")

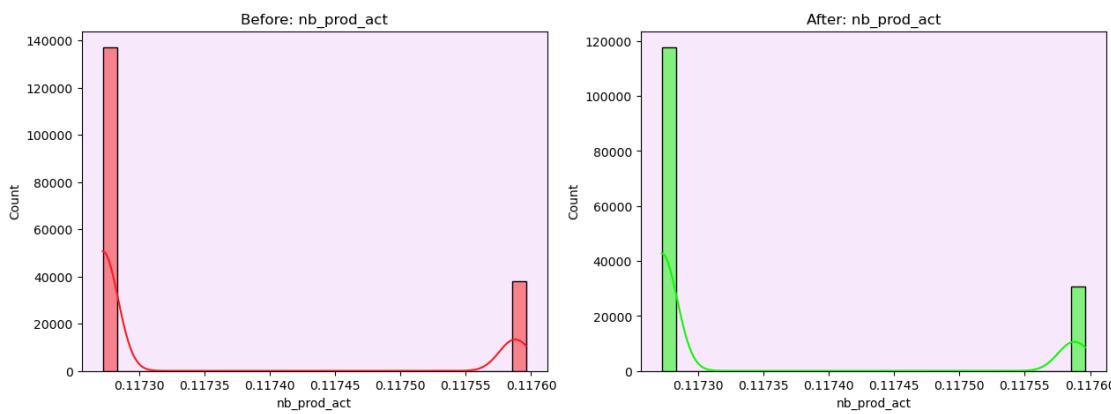
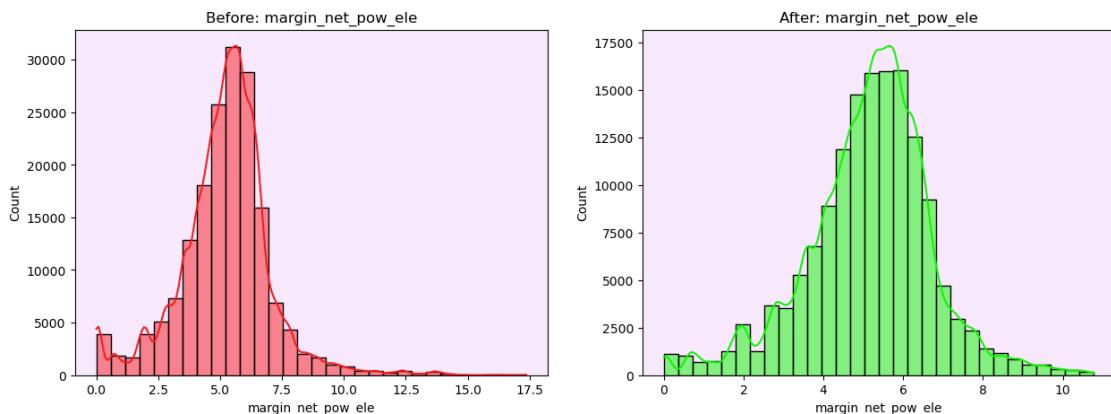
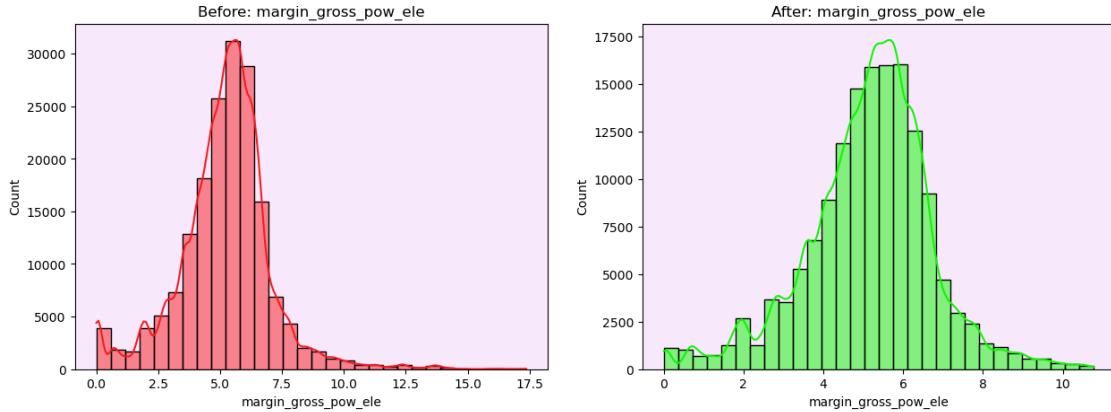
    plt.show()
transfoplots()
```

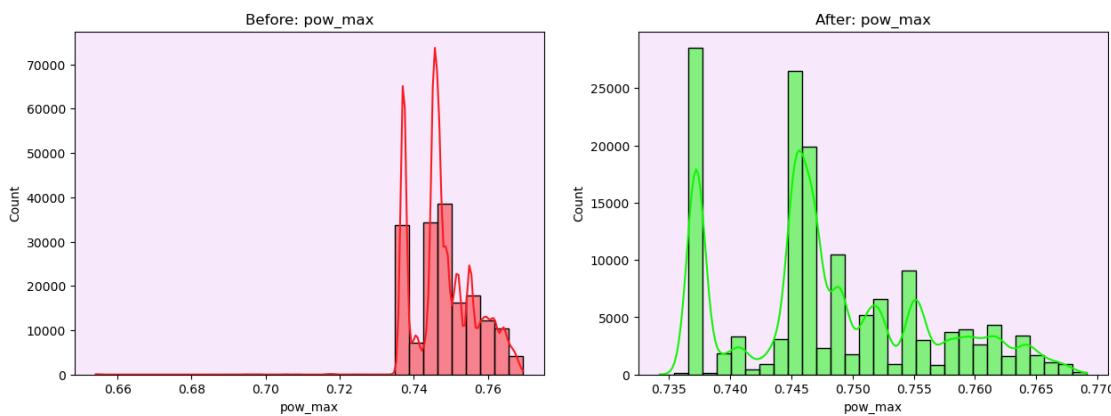
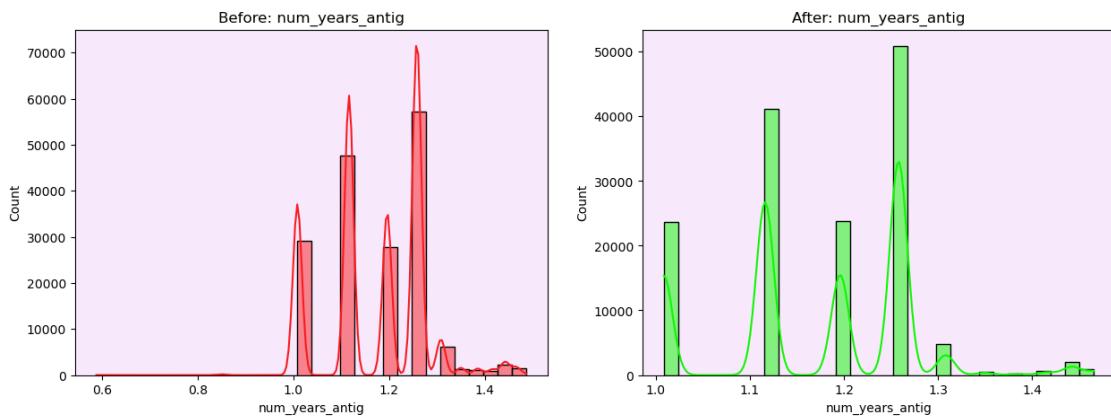
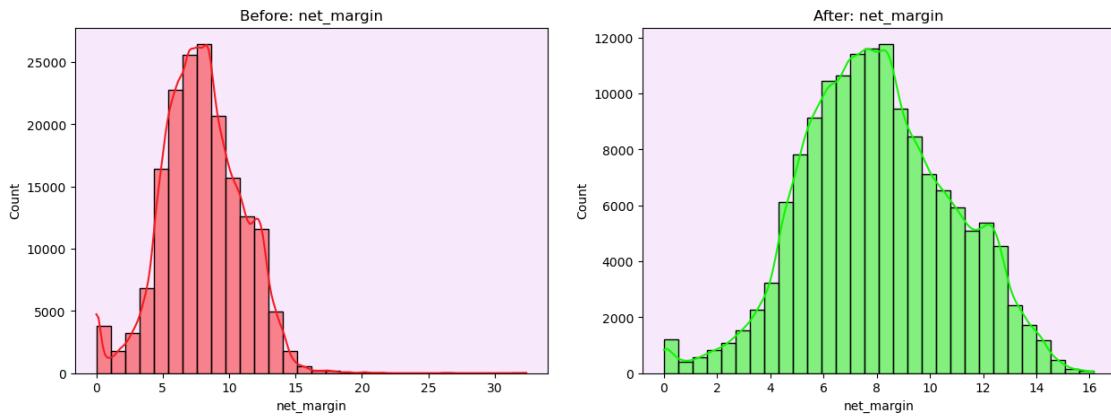


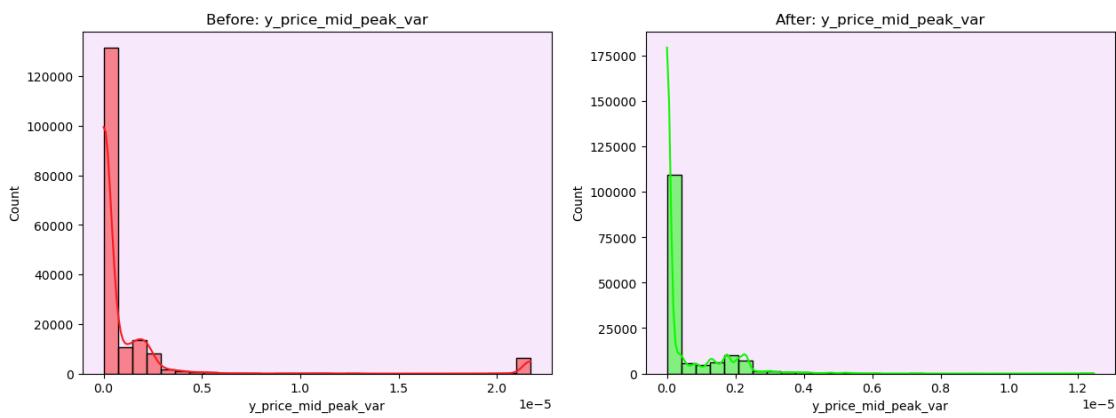
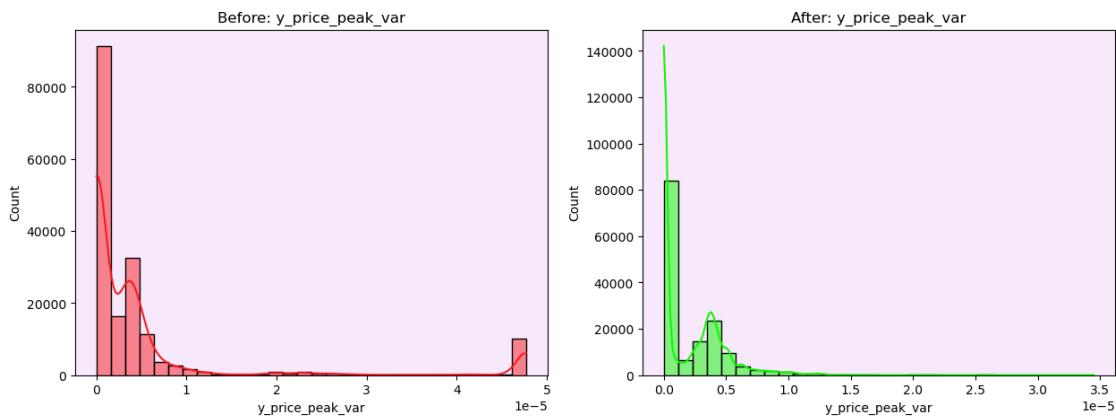
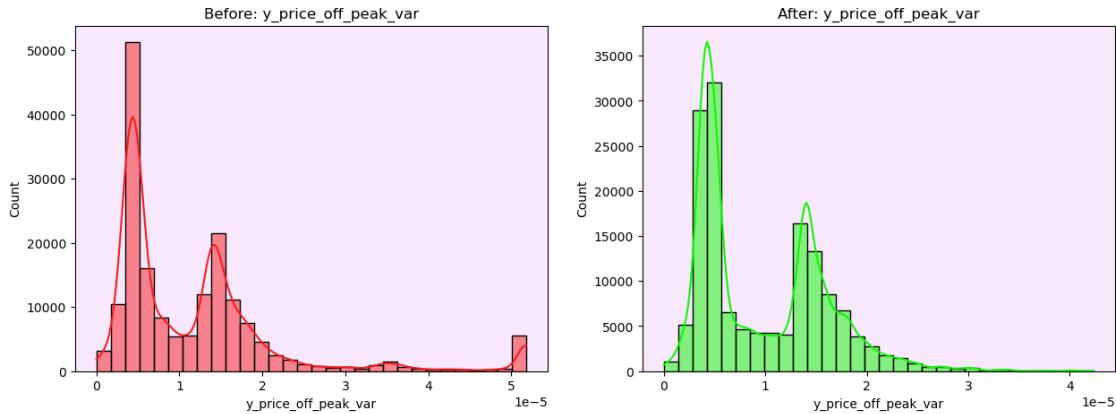


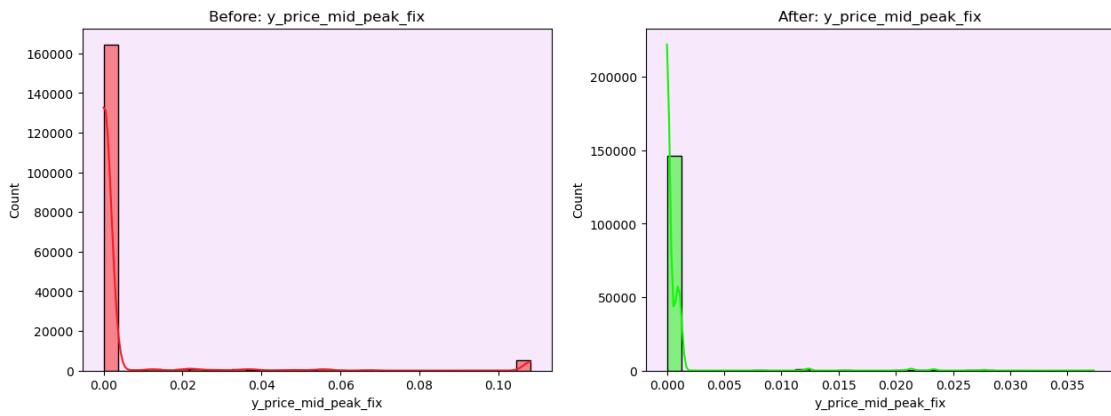
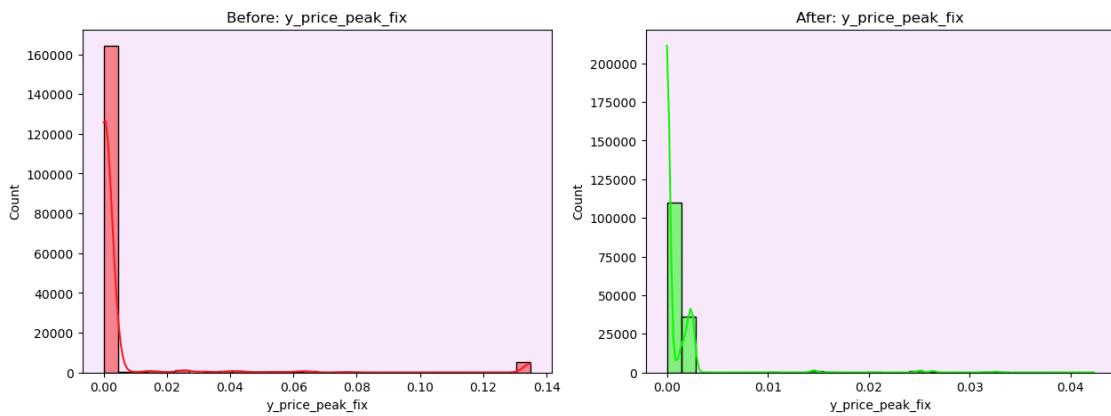
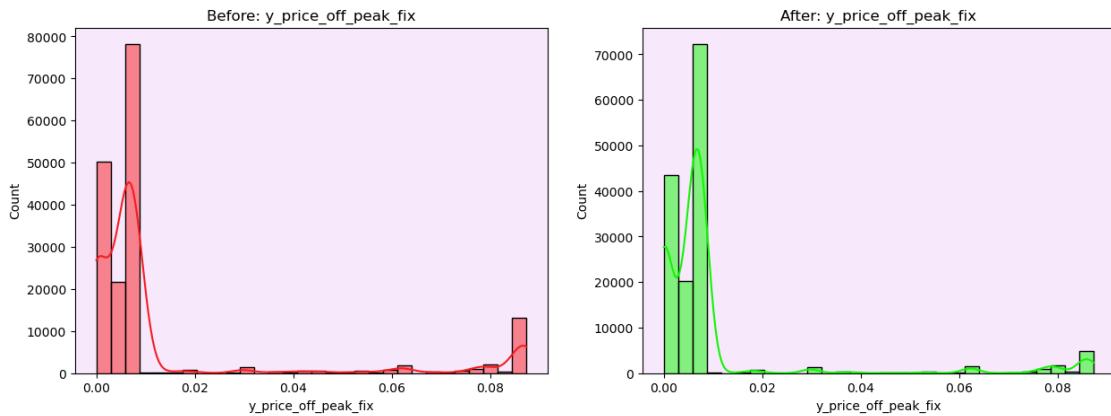


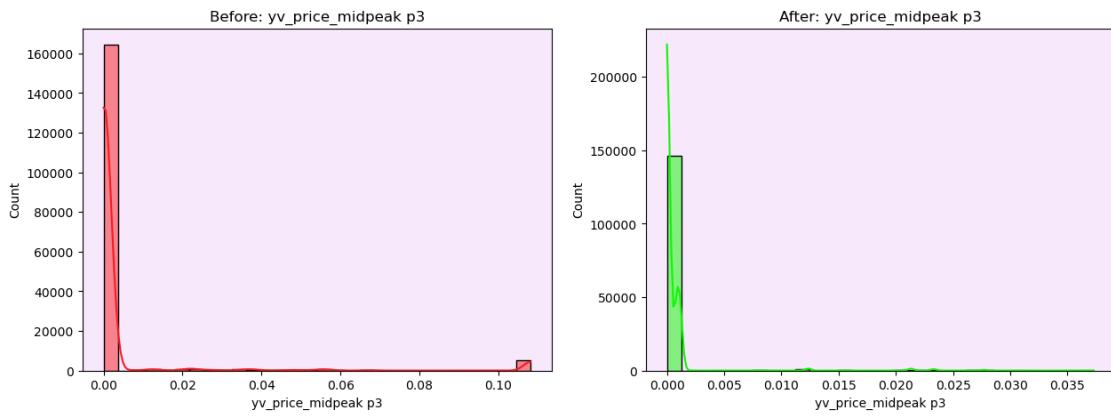
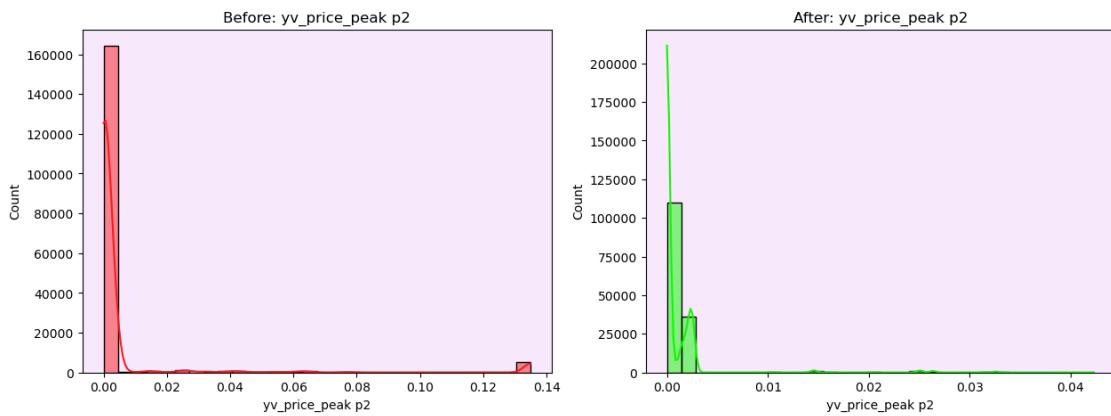
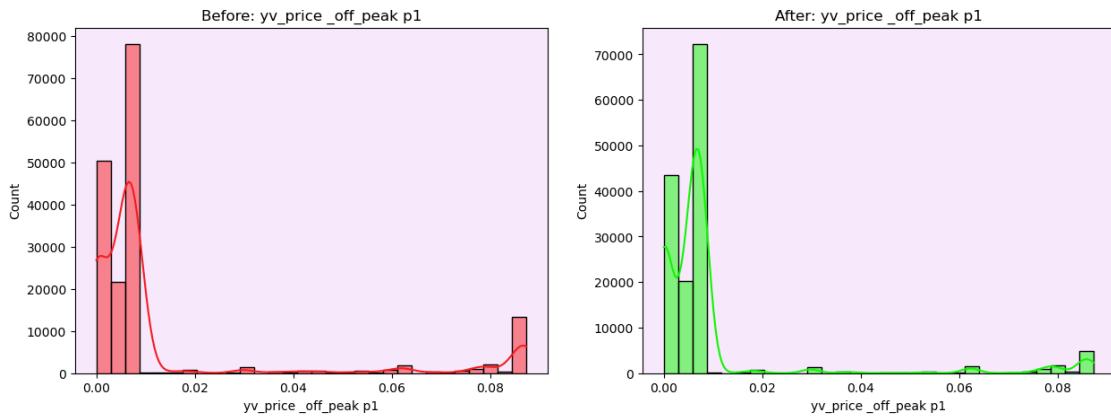


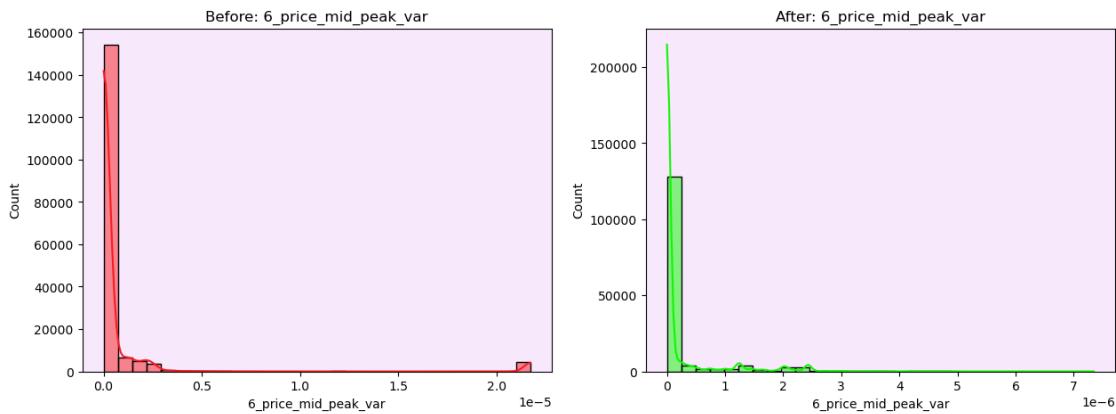
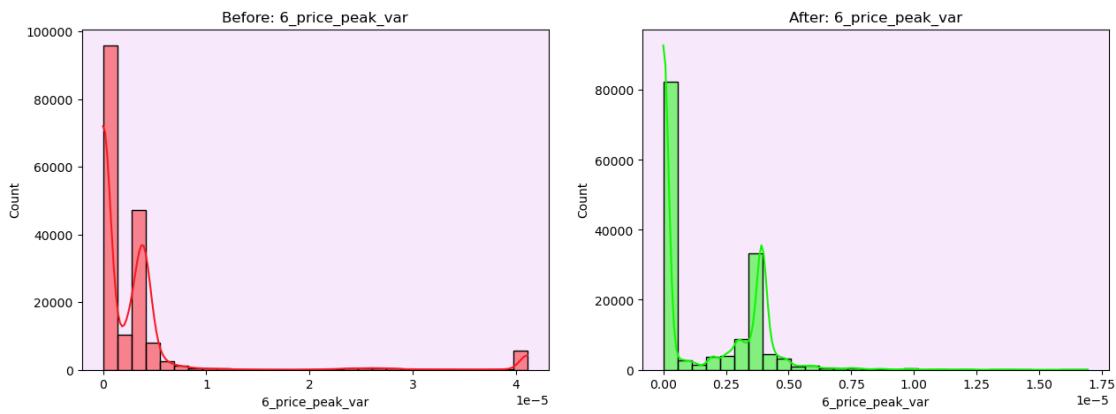
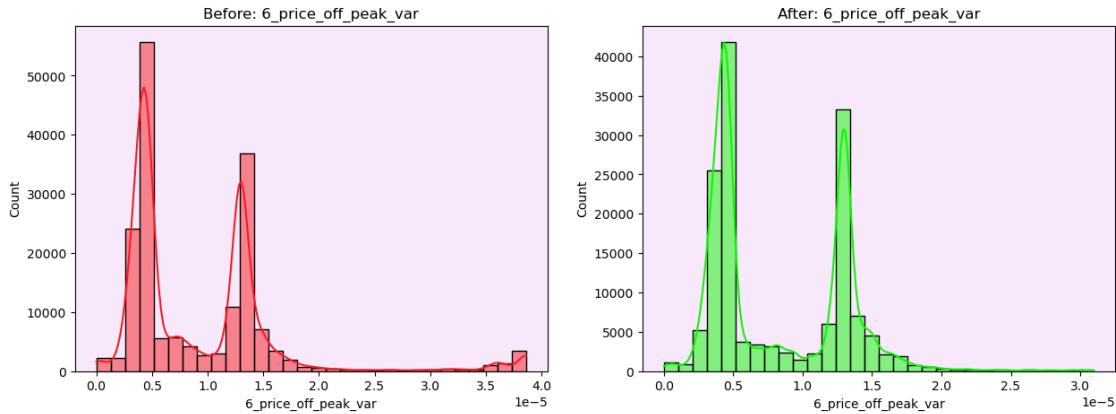


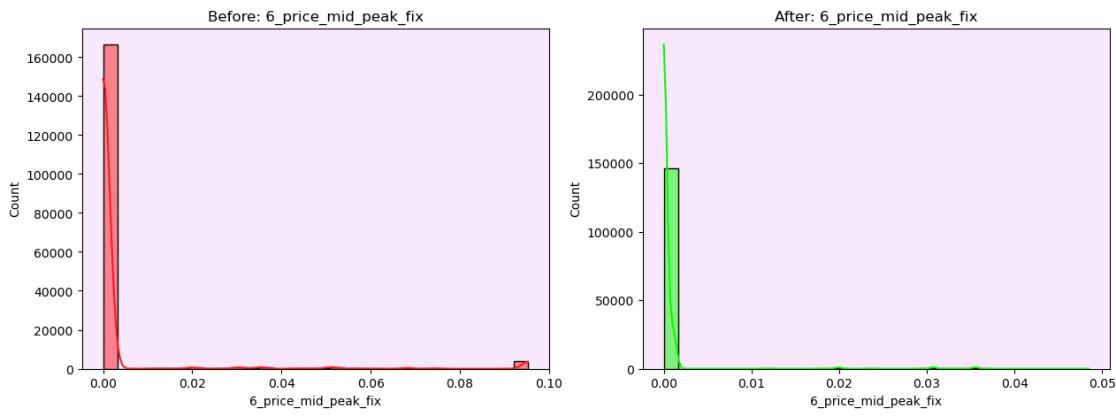
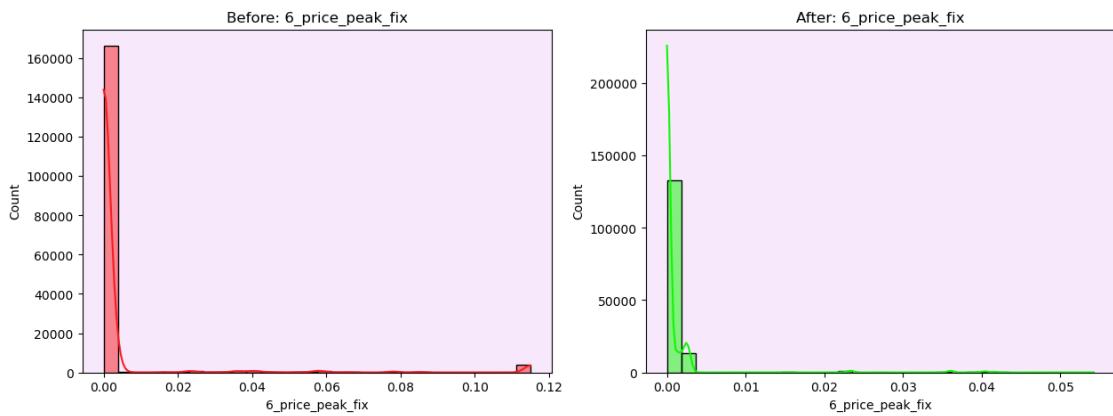
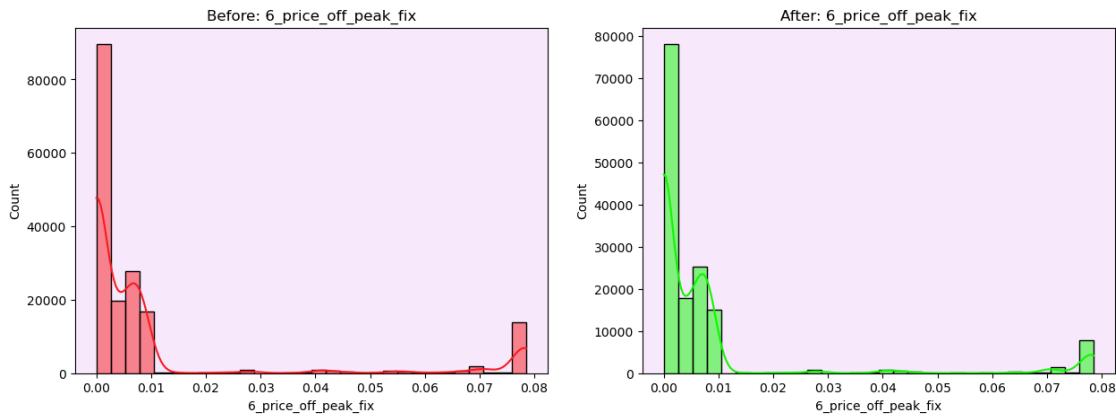


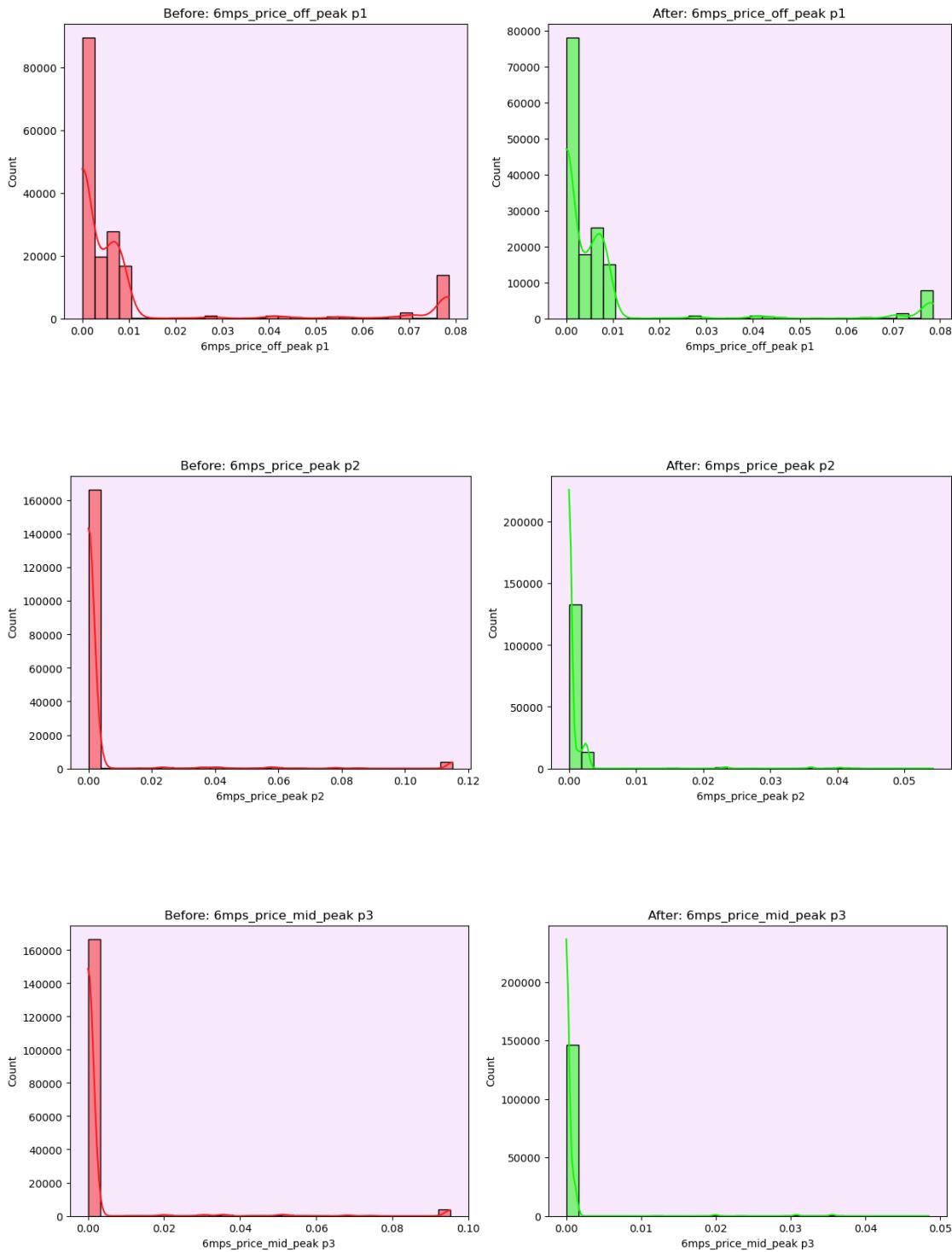


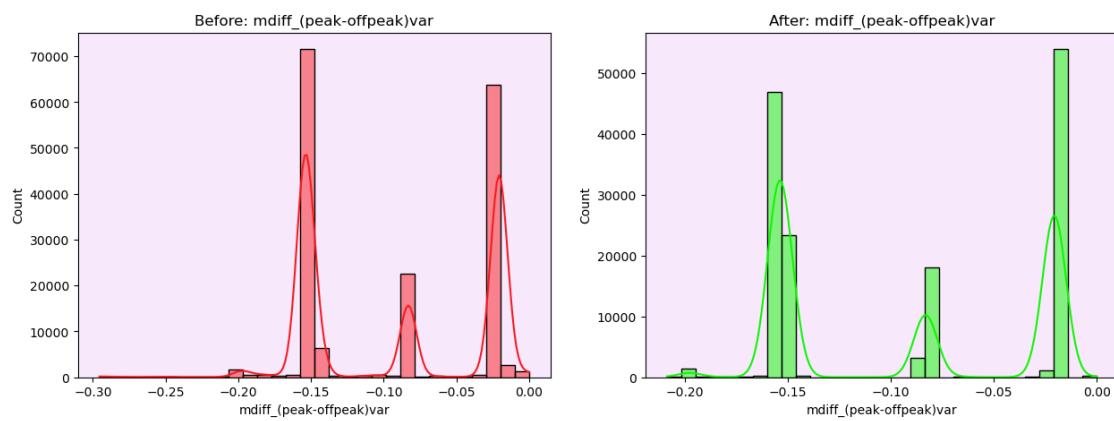
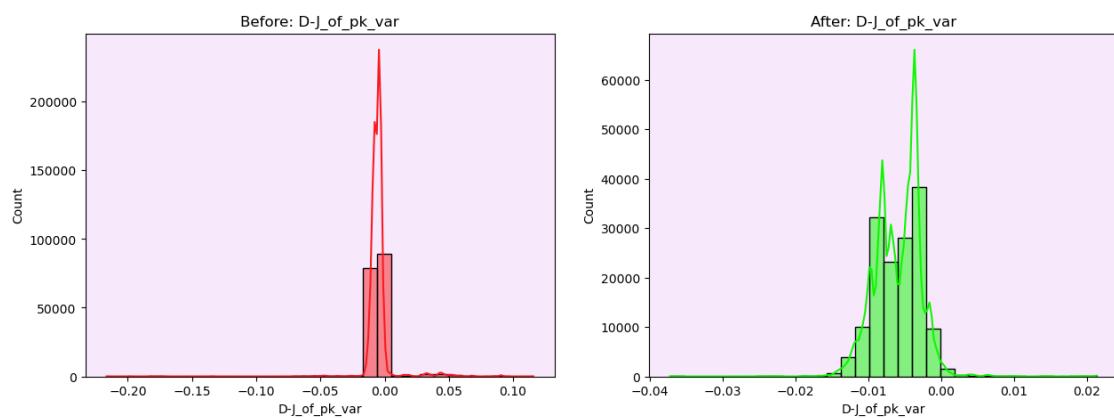
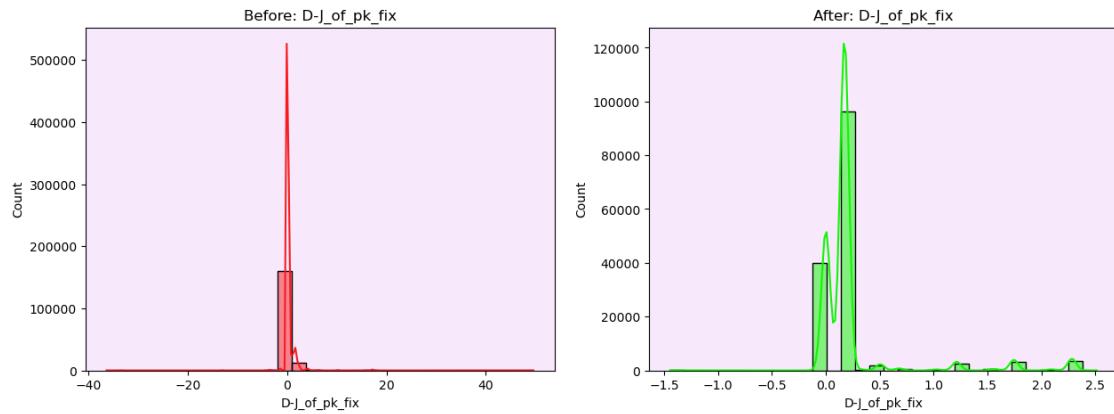


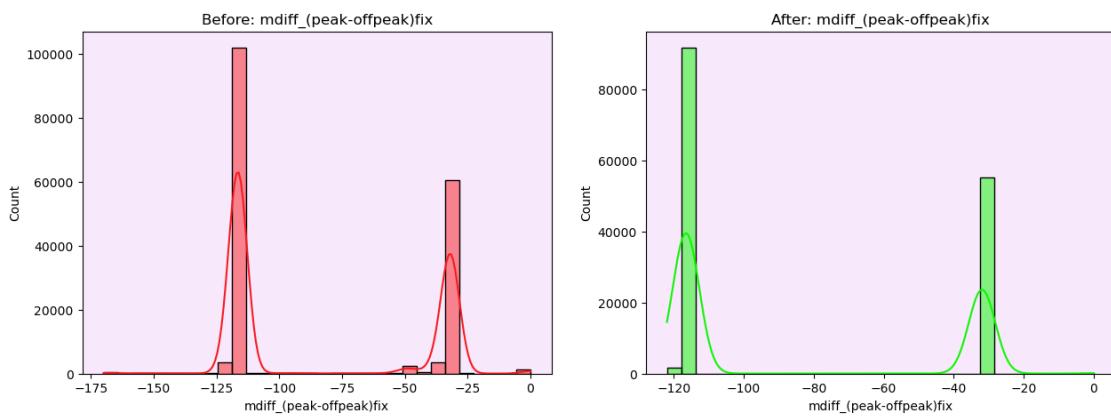
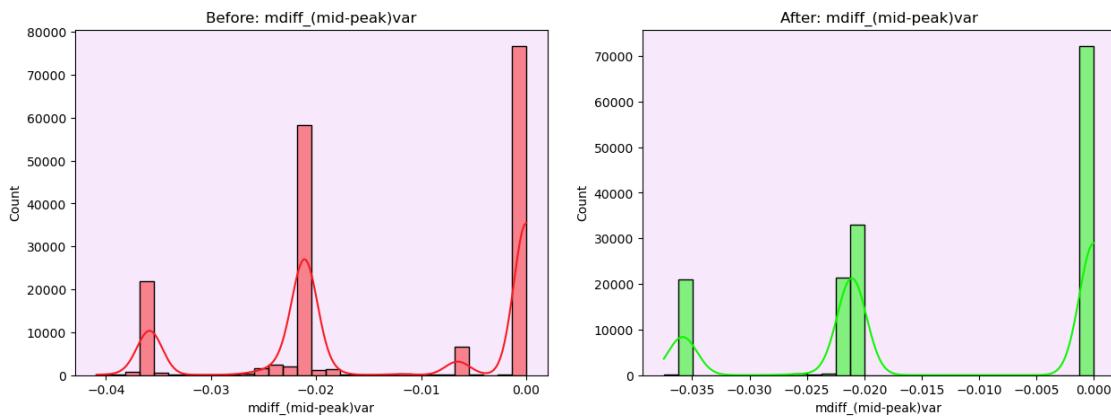
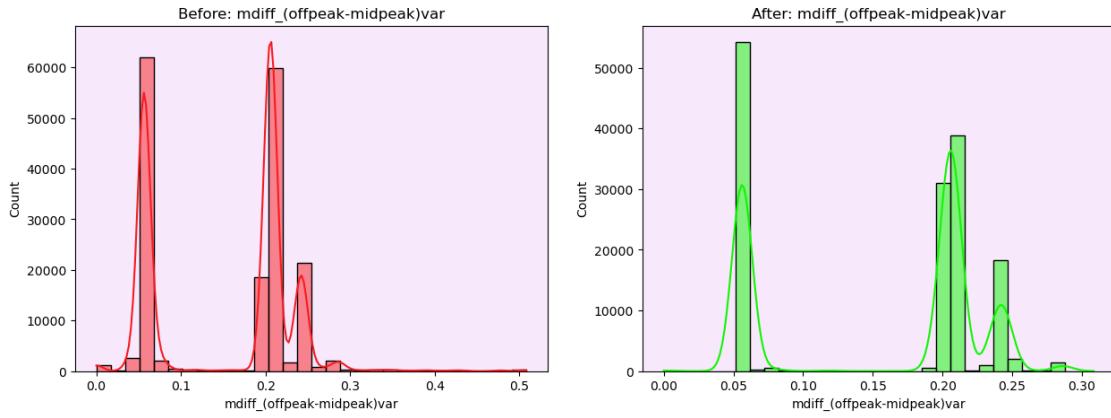


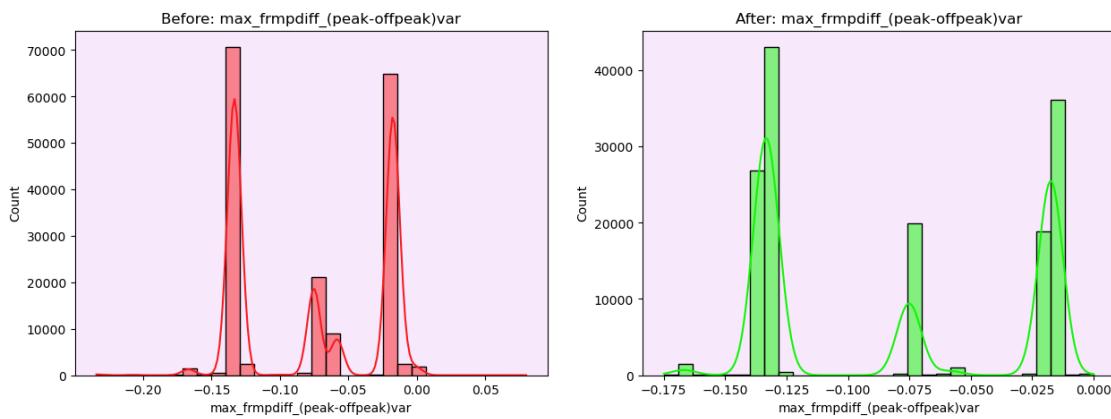
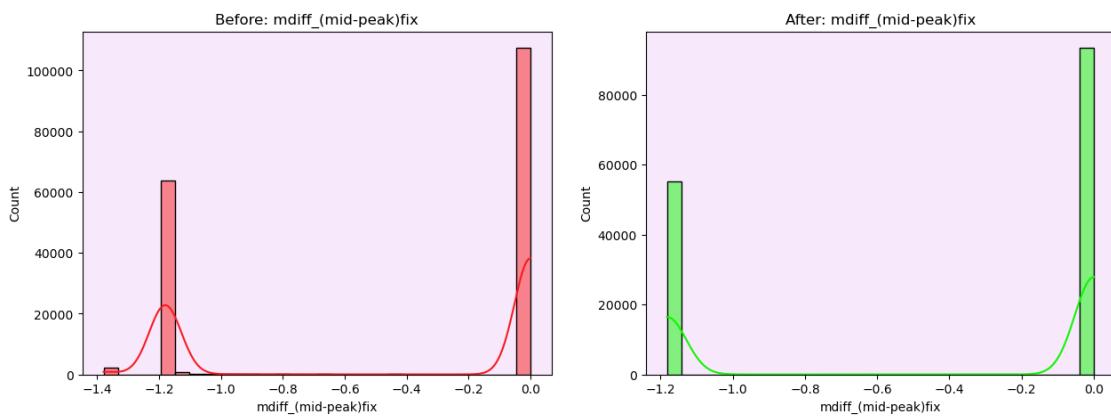
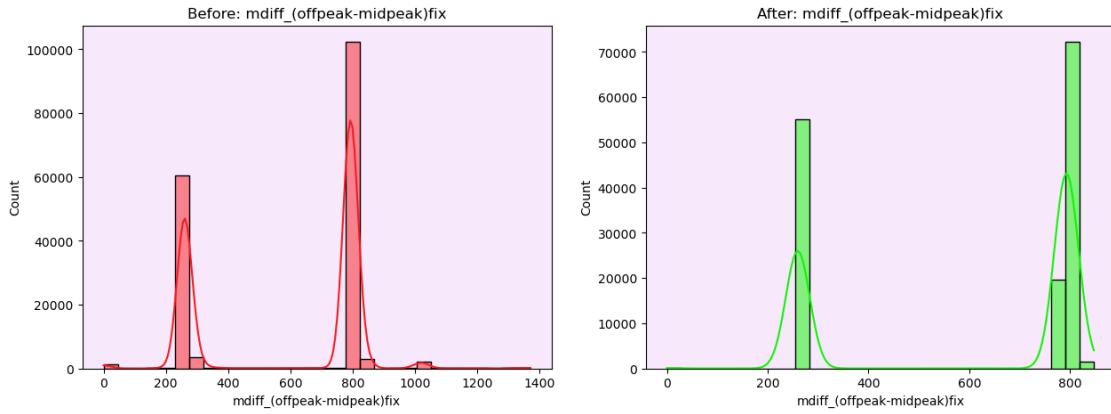


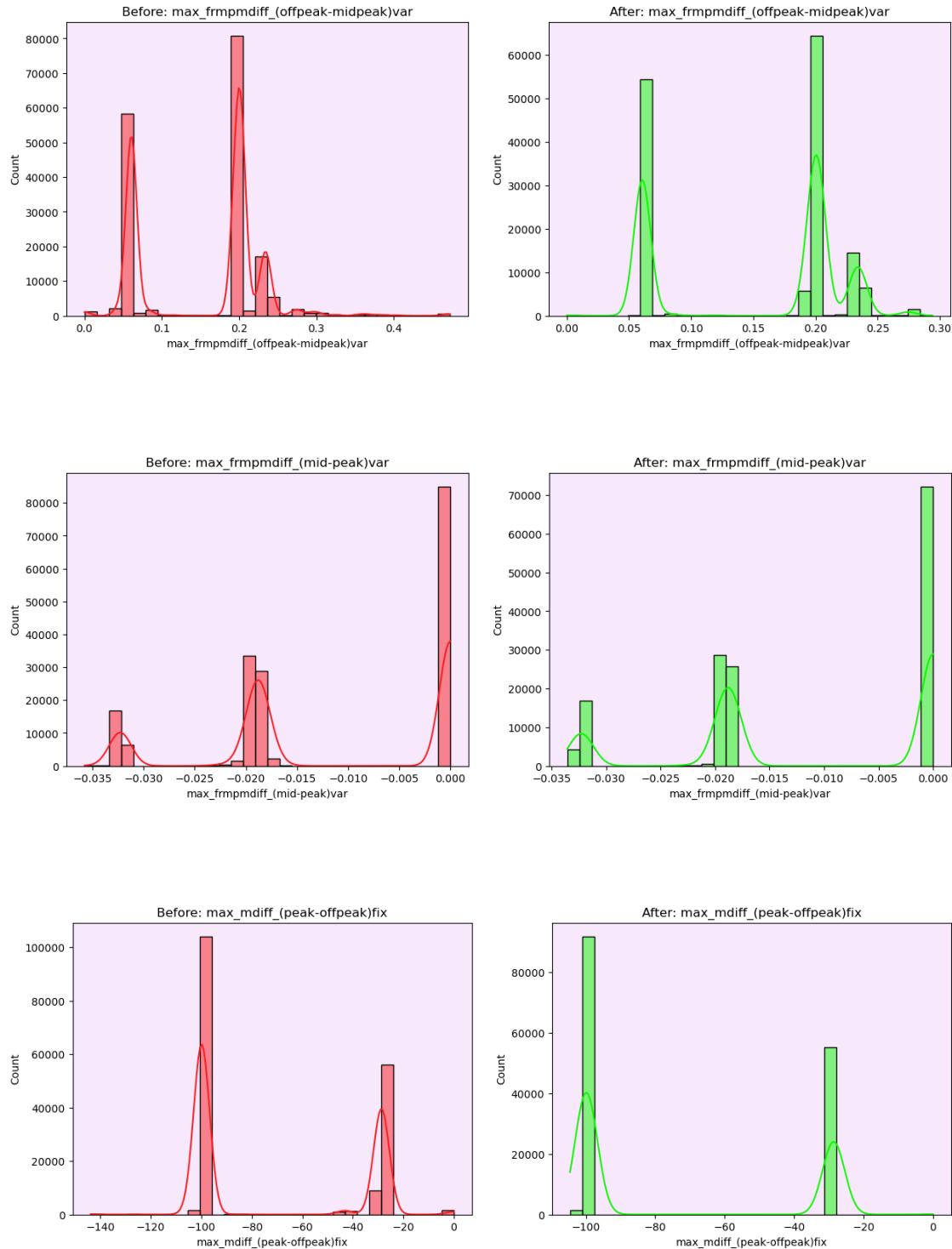


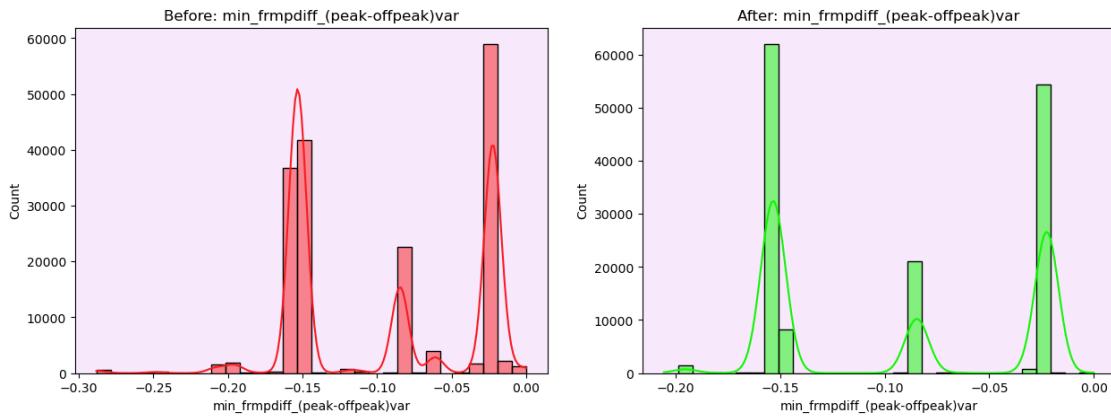
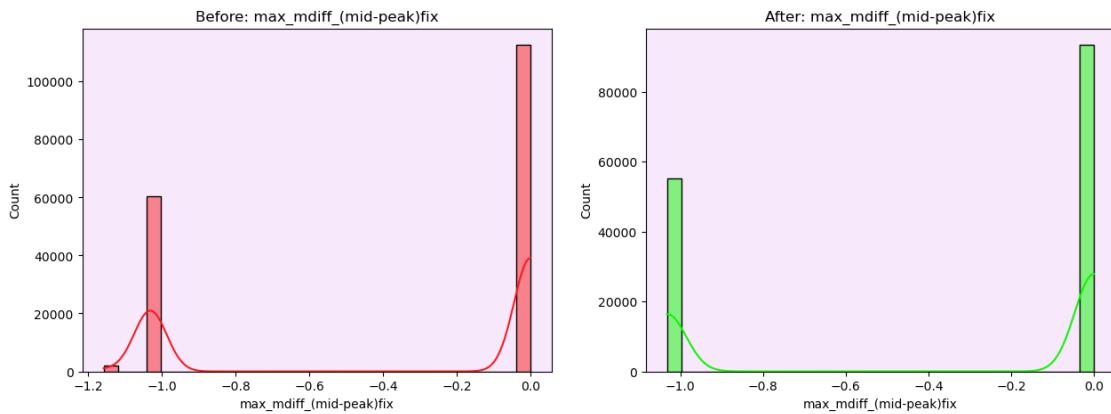
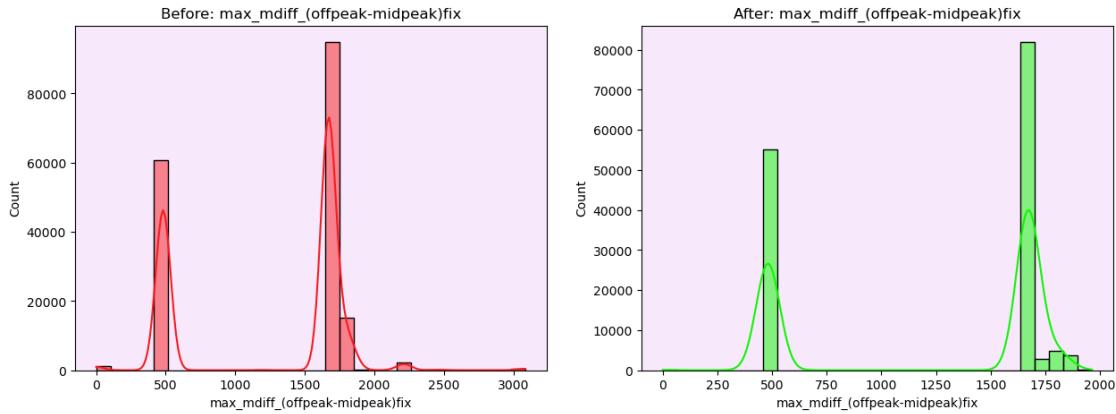


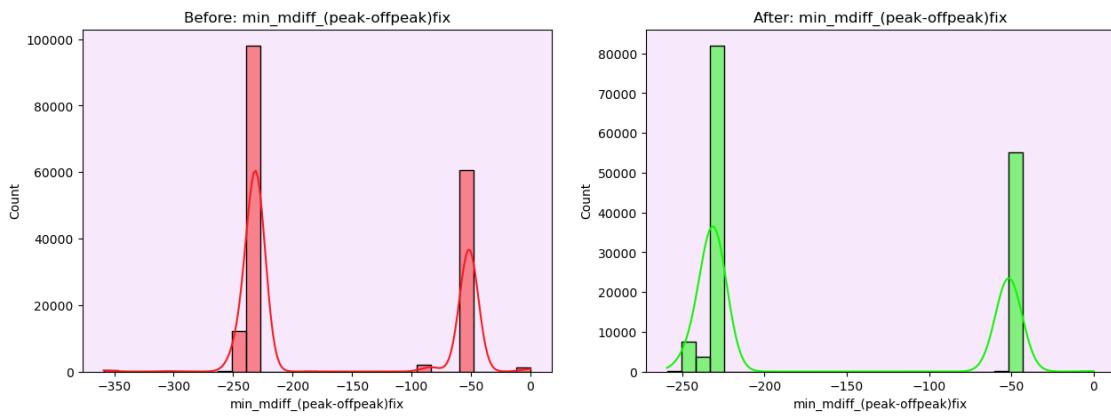
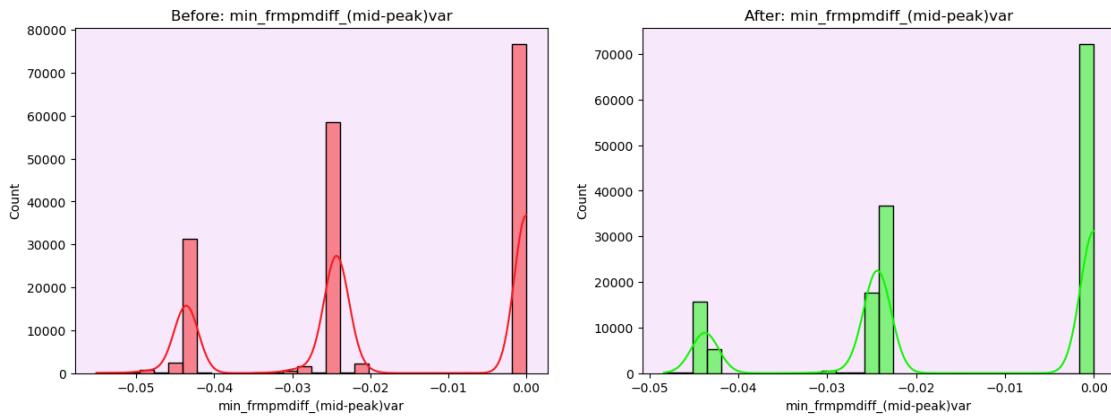
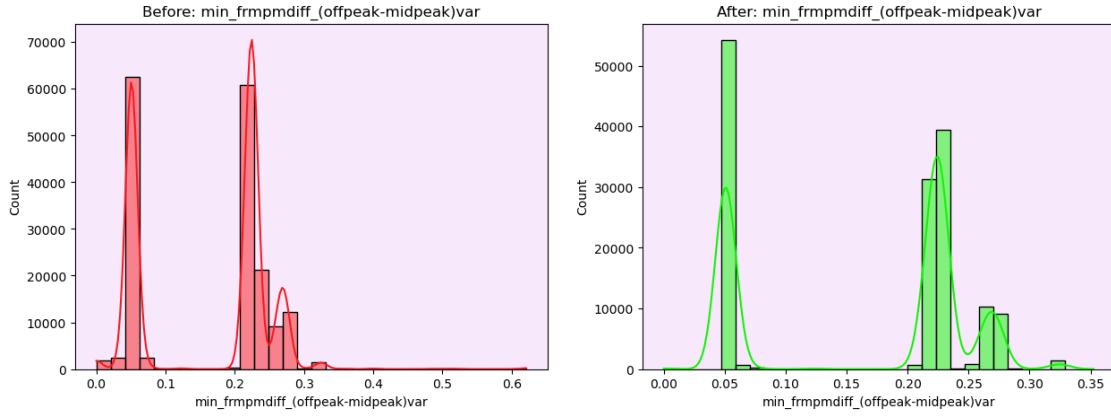


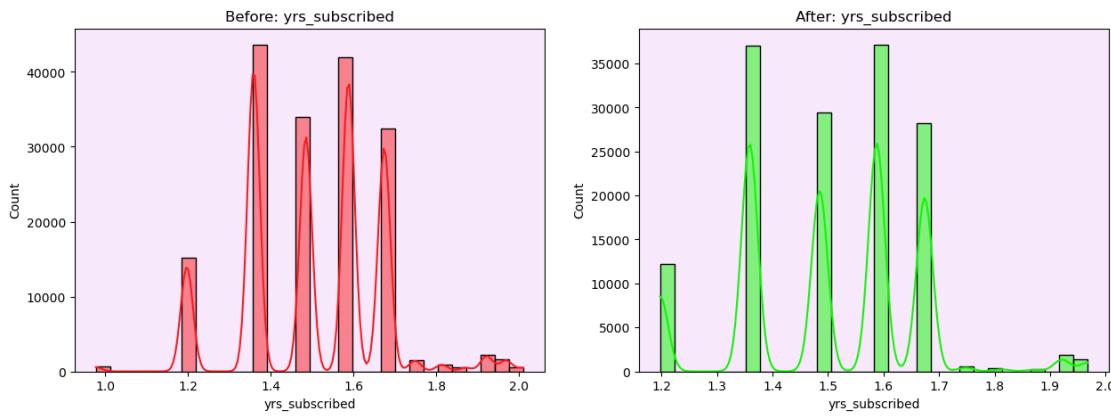
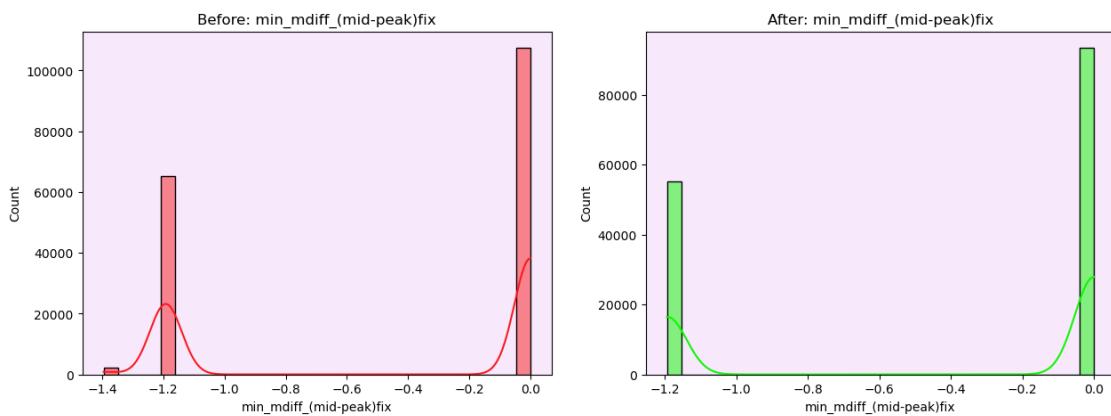
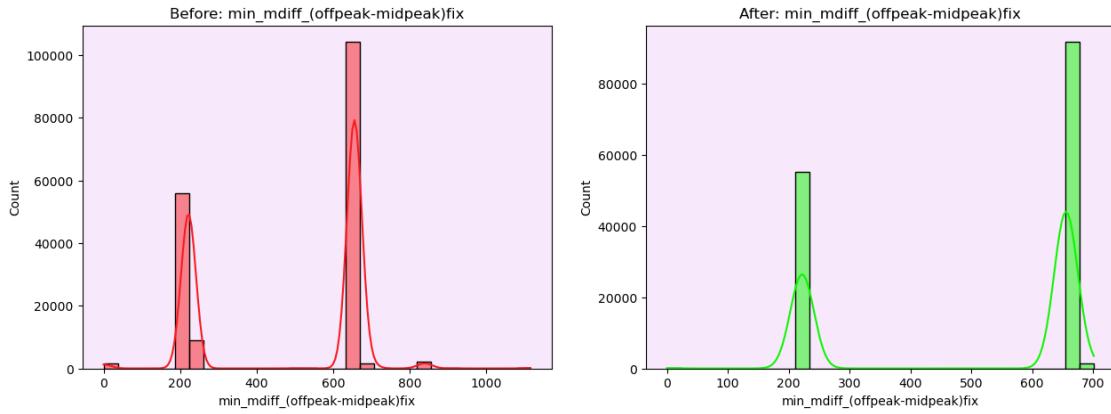


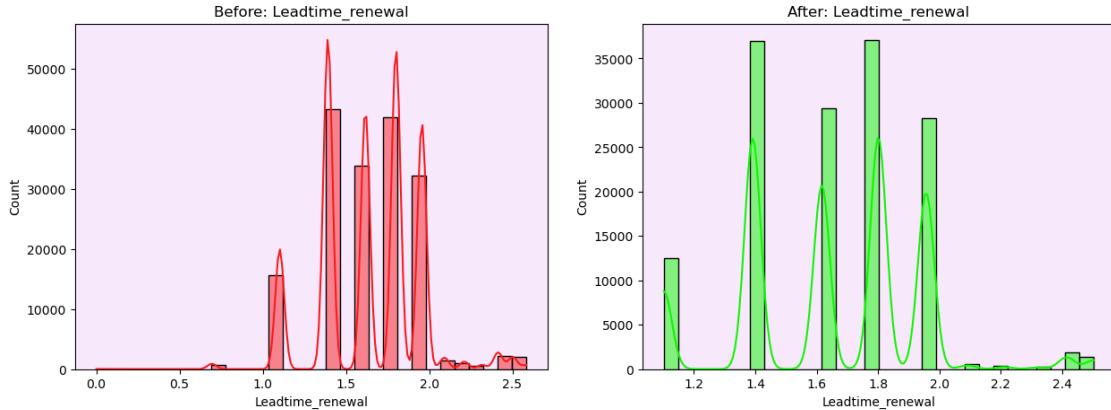












From the Yeo Johnson transformations really worked for this dataset.

```
[45]: ft_out.select_dtypes(include=['datetime64']).columns
```

```
[45]: Index(['date_activ', 'date_end', 'date_modif_prod', 'date_renewal',
       'price_date'],
       dtype='object')
```

```
[46]: # conversion of datetime features to months
for i in ft_out.select_dtypes(include=['datetime64']):
    ft_out[i]=pd.DatetimeIndex(ft_out[i]).month
```

```
/tmp/ipykernel_33/1962657511.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
ft_out[i]=pd.DatetimeIndex(ft_out[i]).month
```

```
[47]: ft_out[['date_activ', 'date_end', 'date_modif_prod', 'date_renewal',
       'price_date']].head()
```

	date_activ	date_end	date_modif_prod	date_renewal	price_date
24	4	4		4	12
25	4	4		4	12
26	4	4		4	12
27	4	4		4	12
28	4	4		4	12

```
[48]: # conversion of datetime features to months
for i in ft_out.select_dtypes(include=['object','category']):
```

```
print(f"[{i}]")
```

```
id  
channel_sales  
has_gas  
origin_up
```

```
[49]: po=ft_out.copy()  
po
```

```
[49]:
```

		id	channel_sales	\			
24		764c75f661154dac3a6c254cd082ea7d	foosdfpfkusacimwkcsothicdxkicaua				
25		764c75f661154dac3a6c254cd082ea7d	foosdfpfkusacimwkcsothicdxkicaua				
26		764c75f661154dac3a6c254cd082ea7d	foosdfpfkusacimwkcsothicdxkicaua				
27		764c75f661154dac3a6c254cd082ea7d	foosdfpfkusacimwkcsothicdxkicaua				
28		764c75f661154dac3a6c254cd082ea7d	foosdfpfkusacimwkcsothicdxkicaua				
...		...	...	...			
175144		563dde550fd624d7352f3de77c0cdfcd		MISSING			
175145		563dde550fd624d7352f3de77c0cdfcd		MISSING			
175146		563dde550fd624d7352f3de77c0cdfcd		MISSING			
175147		563dde550fd624d7352f3de77c0cdfcd		MISSING			
175148		563dde550fd624d7352f3de77c0cdfcd		MISSING			
...		...	...	...			
		cons_12m	cons_gas_12m	cons_last_month	date_activ	date_end	\
24		6.828974	-0.0	0.0	4	4	
25		6.828974	-0.0	0.0	4	4	
26		6.828974	-0.0	0.0	4	4	
27		6.828974	-0.0	0.0	4	4	
28		6.828974	-0.0	0.0	4	4	
...		...	...	...	...	...	...
175144		10.196953	-0.0	0.0	12	12	
175145		10.196953	-0.0	0.0	12	12	
175146		10.196953	-0.0	0.0	12	12	
175147		10.196953	-0.0	0.0	12	12	
175148		10.196953	-0.0	0.0	12	12	
...		...	...	...	...	...	...
		date_modif_prod	date_renewal	forecast_cons_12m	forecast_cons_year	\	
24		4	4	6.849719		0.0	
25		4	4	6.849719		0.0	
26		4	4	6.849719		0.0	
27		4	4	6.849719		0.0	
28		4	4	6.849719		0.0	
...		...	...	...	...	...	...
175144		12	12	18.345274		0.0	
175145		12	12	18.345274		0.0	
175146		12	12	18.345274		0.0	
175147		12	12	18.345274		0.0	

175148	12	12	18.345274	0.0	
	forecast_discount_energy	forecast_meter_rent_12m	\		
24	0	5.120009			
25	0	5.120009			
26	0	5.120009			
27	0	5.120009			
28	0	5.120009			
...	...	...			
175144	0	0.774527			
175145	0	0.774527			
175146	0	0.774527			
175147	0	0.774527			
175148	0	0.774527			
	forecast_price_energy_off_peak	forecast_price_energy_peak	\		
24	0.174584	0.086345			
25	0.174584	0.086345			
26	0.174584	0.086345			
27	0.174584	0.086345			
28	0.174584	0.086345			
...	...	...			
175144	0.176013	0.086880			
175145	0.176013	0.086880			
175146	0.176013	0.086880			
175147	0.176013	0.086880			
175148	0.176013	0.086880			
	forecast_price_pow_off_peak	has_gas	imp_cons	margin_gross_pow_ele	\
24	1704.515088	f	-0.0	6.053563	
25	1704.515088	f	-0.0	6.053563	
26	1704.515088	f	-0.0	6.053563	
27	1704.515088	f	-0.0	6.053563	
28	1704.515088	f	-0.0	6.053563	
...	...	...	...	...	
175144	1786.531993	f	-0.0	3.919026	
175145	1786.531993	f	-0.0	3.919026	
175146	1786.531993	f	-0.0	3.919026	
175147	1786.531993	f	-0.0	3.919026	
175148	1786.531993	f	-0.0	3.919026	
	margin_net_pow_ele	nb_prod_act	net_margin	num_years_antig	\
24	6.053064	0.117272	2.496395	1.258073	
25	6.053064	0.117272	2.496395	1.258073	
26	6.053064	0.117272	2.496395	1.258073	
27	6.053064	0.117272	2.496395	1.258073	
28	6.053064	0.117272	2.496395	1.258073	

...            ...            ...            ...            ...  
 175144        3.918791     0.117272     7.456686     1.258073  
 175145        3.918791     0.117272     7.456686     1.258073  
 175146        3.918791     0.117272     7.456686     1.258073  
 175147        3.918791     0.117272     7.456686     1.258073  
 175148        3.918791     0.117272     7.456686     1.258073

origin\_up    pow\_max    y\_price\_off\_peak\_var \

24        kamkkxfxxuwbds1kwifmmcsiusiuosws    0.746777    0.000005  
 25        kamkkxfxxuwbds1kwifmmcsiusiuosws    0.746777    0.000005  
 26        kamkkxfxxuwbds1kwifmmcsiusiuosws    0.746777    0.000005  
 27        kamkkxfxxuwbds1kwifmmcsiusiuosws    0.746777    0.000005  
 28        kamkkxfxxuwbds1kwifmmcsiusiuosws    0.746777    0.000005

...            ...            ...            ...  
 175144      ldkssxwpmemidmecebumciepifcamkci    0.737249    0.000005  
 175145      ldkssxwpmemidmecebumciepifcamkci    0.737249    0.000005  
 175146      ldkssxwpmemidmecebumciepifcamkci    0.737249    0.000005  
 175147      ldkssxwpmemidmecebumciepifcamkci    0.737249    0.000005  
 175148      ldkssxwpmemidmecebumciepifcamkci    0.737249    0.000005

y\_price\_peak\_var    y\_price\_mid\_peak\_var    y\_price\_off\_peak\_fix \

24        2.551665e-07                            -0.0        7.308431e-03  
 25        2.551665e-07                            -0.0        7.308431e-03  
 26        2.551665e-07                            -0.0        7.308431e-03  
 27        2.551665e-07                            -0.0        7.308431e-03  
 28        2.551665e-07                            -0.0        7.308431e-03

...            ...            ...            ...  
 175144      1.501067e-07                            -0.0        3.819191e-13  
 175145      1.501067e-07                            -0.0        3.819191e-13  
 175146      1.501067e-07                            -0.0        3.819191e-13  
 175147      1.501067e-07                            -0.0        3.819191e-13  
 175148      1.501067e-07                            -0.0        3.819191e-13

yv\_price \_off\_peak p1 \

24        -0.0                                    -0.0        0.007314  
 25        -0.0                                    -0.0        0.007314  
 26        -0.0                                    -0.0        0.007314  
 27        -0.0                                    -0.0        0.007314  
 28        -0.0                                    -0.0        0.007314

...            ...            ...            ...  
 175144      -0.0                                    -0.0        0.000006  
 175145      -0.0                                    -0.0        0.000006  
 175146      -0.0                                    -0.0        0.000006  
 175147      -0.0                                    -0.0        0.000006  
 175148      -0.0                                    -0.0        0.000006

yv\_price\_peak p2    yv\_price\_midpeak p3    6\_price\_off\_peak\_var \

24	2.558508e-07	-0.0	0.000005
25	2.558508e-07	-0.0	0.000005
26	2.558508e-07	-0.0	0.000005
27	2.558508e-07	-0.0	0.000005
28	2.558508e-07	-0.0	0.000005
...	...	...	...
175144	1.503432e-07	-0.0	0.000005
175145	1.503432e-07	-0.0	0.000005
175146	1.503432e-07	-0.0	0.000005
175147	1.503432e-07	-0.0	0.000005
175148	1.503432e-07	-0.0	0.000005
24	6_price_peak_var	6_price_mid_peak_var	6_price_off_peak_fix \
24	1.347807e-07	-0.0	-0.0
25	1.347807e-07	-0.0	-0.0
26	1.347807e-07	-0.0	-0.0
27	1.347807e-07	-0.0	-0.0
28	1.347807e-07	-0.0	-0.0
...	...	...	...
175144	1.347807e-07	-0.0	-0.0
175145	1.347807e-07	-0.0	-0.0
175146	1.347807e-07	-0.0	-0.0
175147	1.347807e-07	-0.0	-0.0
175148	1.347807e-07	-0.0	-0.0
24	6_price_peak_fix	6_price_mid_peak_fix	6mps_price_off_peak p1 \
24	-0.0	-0.0	0.000005
25	-0.0	-0.0	0.000005
26	-0.0	-0.0	0.000005
27	-0.0	-0.0	0.000005
28	-0.0	-0.0	0.000005
...	...	...	...
175144	-0.0	-0.0	0.000005
175145	-0.0	-0.0	0.000005
175146	-0.0	-0.0	0.000005
175147	-0.0	-0.0	0.000005
175148	-0.0	-0.0	0.000005
24	6mps_price_peak p2	6mps_price_mid_peak p3	churn price_date \
24	1.350021e-07	-0.0	0 12
25	1.350021e-07	-0.0	0 12
26	1.350021e-07	-0.0	0 12
27	1.350021e-07	-0.0	0 12
28	1.350021e-07	-0.0	0 12
...	...	...	...
175144	1.350021e-07	-0.0	0 12
175145	1.350021e-07	-0.0	0 12

175146	1.350021e-07	-0.0	0	12
175147	1.350021e-07	-0.0	0	12
175148	1.350021e-07	-0.0	0	12
	D-J_of_pk_fix	D-J_of_pk_var	mdiff_(peak-offpeak)var	\
24	0.178813	-0.004725	-0.083655	
25	0.178813	-0.004725	-0.083655	
26	0.178813	-0.004725	-0.083655	
27	0.178813	-0.004725	-0.083655	
28	0.178813	-0.004725	-0.083655	
...	...	...	...	
175144	-0.000001	-0.004682	-0.082852	
175145	-0.000001	-0.004682	-0.082852	
175146	-0.000001	-0.004682	-0.082852	
175147	-0.000001	-0.004682	-0.082852	
175148	-0.000001	-0.004682	-0.082852	
	mdiff_(offpeak-midpeak)var	mdiff_(mid-peak)var	\	
24	0.247128	-0.036089		
25	0.247128	-0.036089		
26	0.247128	-0.036089		
27	0.247128	-0.036089		
28	0.247128	-0.036089		
...	...	...	...	
175144	0.243469	-0.035959		
175145	0.243469	-0.035959		
175146	0.243469	-0.035959		
175147	0.243469	-0.035959		
175148	0.243469	-0.035959		
	mdiff_(peak-offpeak)fix	mdiff_(offpeak-midpeak)fix	\	
24	-116.555088	793.885635		
25	-116.555088	793.885635		
26	-116.555088	793.885635		
27	-116.555088	793.885635		
28	-116.555088	793.885635		
...	...	...	...	
175144	-116.150237	789.904364		
175145	-116.150237	789.904364		
175146	-116.150237	789.904364		
175147	-116.150237	789.904364		
175148	-116.150237	789.904364		
	mdiff_(mid-peak)fix	max_frmpdiff_(peak-offpeak)var	\	
24	0.0	-0.075504		
25	0.0	-0.075504		
26	0.0	-0.075504		

27	0.0	-0.075504
28	0.0	-0.075504
...	...	...
175144	0.0	-0.075203
175145	0.0	-0.075203
175146	0.0	-0.075203
175147	0.0	-0.075203
175148	0.0	-0.075203
24	max_frmpmdiff_(offpeak-midpeak)var	max_frmpmdiff_(mid-peak)var \ 0.238161 -0.032558
25		0.238161 -0.032558
26		0.238161 -0.032558
27		0.238161 -0.032558
28		0.238161 -0.032558
...	...	...
175144	0.234750	-0.032470
175145	0.234750	-0.032470
175146	0.234750	-0.032470
175147	0.234750	-0.032470
175148	0.234750	-0.032470
24	max_mdiff_(peak-offpeak)fix	max_mdiff_(offpeak-midpeak)fix \ -99.797374 1673.962443
25		-99.797374 1673.962443
26		-99.797374 1673.962443
27		-99.797374 1673.962443
28		-99.797374 1673.962443
...	...	...
175144	-99.797371	1659.943479
175145	-99.797371	1659.943479
175146	-99.797371	1659.943479
175147	-99.797371	1659.943479
175148	-99.797371	1659.943479
24	max_mdiff_(mid-peak)fix	min_frmpdiff_(peak-offpeak)var \ 0.0 -0.085300
25		0.0 -0.085300
26		0.0 -0.085300
27		0.0 -0.085300
28		0.0 -0.085300
...	...	...
175144	0.0	-0.083618
175145	0.0	-0.083618
175146	0.0	-0.083618
175147	0.0	-0.083618
175148	0.0	-0.083618

```

min_frmpmdiff_(offpeak-midpeak)var min_frmpmdiff_(mid-peak)var \
24          0.275277      -0.044072
25          0.275277      -0.044072
26          0.275277      -0.044072
27          0.275277      -0.044072
28          0.275277      -0.044072
...
175144        ...          ...
175145        0.270782      -0.043763
175146        0.270782      -0.043763
175147        0.270782      -0.043763
175148        0.270782      -0.043763

min_mdiff_(peak-offpeak)fix min_mdiff_(offpeak-midpeak)fix \
24          -231.144085     655.036192
25          -231.144085     655.036192
26          -231.144085     655.036192
27          -231.144085     655.036192
28          -231.144085     655.036192
...
175144        ...          ...
175145        -229.749069    655.036160
175146        -229.749069    655.036160
175147        -229.749069    655.036160
175148        -229.749069    655.036160

min_mdiff_(mid-peak)fix yrs_subscribed Leadtime_renewal
24          0.0            1.587724     1.800734
25          0.0            1.587724     1.800734
26          0.0            1.587724     1.800734
27          0.0            1.587724     1.800734
28          0.0            1.587724     1.800734
...
175144        ...          ...
175145        0.0            1.674061     1.956498
175146        0.0            1.674061     1.956498
175147        0.0            1.674061     1.956498
175148        0.0            1.674061     1.956498

```

[148546 rows x 67 columns]

```
[50]: def valuecounts(col):
    counts= ft_out[col].value_counts().sort_values(ascending=False)
    return counts
```

```
[51]: ft_out['channel_sales']=ft_out['channel_sales'].astype('category')
       valuecounts('channel_sales')
```

```
/tmp/ipykernel_33/3281812955.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

```
See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
       ft_out['channel_sales']=ft_out['channel_sales'].astype('category')
```

```
[51]: foosdfpfkusacimwkcsothicdxkicaua    69773
      MISSING                         36289
      lmkebamcaaclubfxadlmueccxoimlema   19386
      usilxuppasemubllopkafesmlibmsdf    13837
      ewpakwlliwiwiwduibdlfmalxowmwpci  9129
      sddiedcslfslkckwlfkdpoeailfpeds   84
      epumfxlbckeskwekxbiuasklxalciiuu  24
      fixdbufsefwooaasfcxdxadsiekoceaa  24
      Name: channel_sales, dtype: int64
```

```
[52]: ft_out=pd.
       ↪get_dummies(data=ft_out,columns=['channel_sales'],prefix='channelsale')
```

```
[53]: valuecounts('has_gas')
```

```
[53]: f    121388
      t    27158
      Name: has_gas, dtype: int64
```

```
[54]: ft_out['has_gas']=ft_out['has_gas'].map({'f':0,'t':1})
```

```
[55]: ft_out['origin_up']=ft_out['origin_up'].astype('category')
       valuecounts('origin_up')
```

```
[55]: lxiidpiddsbxsbosboudacockeimpuepw  72347
      kamkkxfxxuwbdslkwifmmcsiusiuosws  45103
      ldkssxwpmemidmecebumeipfcamkci    30533
      MISSING                           539
      ewxeelcelemmiwuafmddpobolfuxioce  12
      usapbepcfolokilkwsdiboslwaxobdp   12
      Name: origin_up, dtype: int64
```

```
[56]: ft_out=pd.get_dummies(data=ft_out,columns=['origin_up'],prefix='origin')
```

```
[57]: ft_out.drop(columns=['id'],axis=1,inplace=True)
```

```
ft_out.columns
```

```
[57]: Index(['cons_12m', 'cons_gas_12m', 'cons_last_month', 'date_activ', 'date_end',
       'date_modif_prod', 'date_renewal', 'forecast_cons_12m',
       'forecast_cons_year', 'forecast_discount_energy',
       'forecast_meter_rent_12m', 'forecast_price_energy_off_peak',
       'forecast_price_energy_peak', 'forecast_price_pow_off_peak', 'has_gas',
       'imp_cons', 'margin_gross_pow_ele', 'margin_net_pow_ele', 'nb_prod_act',
       'net_margin', 'num_years_antig', 'pow_max', 'y_price_off_peak_var',
       'y_price_peak_var', 'y_price_mid_peak_var', 'y_price_off_peak_fix',
       'y_price_peak_fix', 'y_price_mid_peak_fix', 'yv_price_off_peak p1',
       'yv_price_peak p2', 'yv_price_midpeak p3', '6_price_off_peak_var',
       '6_price_peak_var', '6_price_mid_peak_var', '6_price_off_peak_fix',
       '6_price_peak_fix', '6_price_mid_peak_fix', '6mps_price_off_peak p1',
       '6mps_price_peak p2', '6mps_price_mid_peak p3', 'churn', 'price_date',
       'D-J_of_pk_fix', 'D-J_of_pk_var', 'mdiff_(peak-offpeak)var',
       'mdiff_(offpeak-midpeak)var', 'mdiff_(mid-peak)var',
       'mdiff_(peak-offpeak)fix', 'mdiff_(offpeak-midpeak)fix',
       'mdiff_(mid-peak)fix', 'max_frmpdiff_(peak-offpeak)var',
       'max_frmpdiff_(offpeak-midpeak)var', 'max_frmpdiff_(mid-peak)var',
       'max_mdif_(peak-offpeak)fix', 'max_mdif_(offpeak-midpeak)fix',
       'max_mdif_(mid-peak)fix', 'min_frmpdiff_(peak-offpeak)var',
       'min_frmpdiff_(offpeak-midpeak)var', 'min_frmpdiff_(mid-peak)var',
       'min_mdif_(peak-offpeak)fix', 'min_mdif_(offpeak-midpeak)fix',
       'min_mdif_(mid-peak)fix', 'yrs_subscribed', 'Leadtime_renewal',
       'channelsale_MISSING', 'channelsale_epumfxlbckeskwekxbiuasklxalciiuu',
       'channelsale_ewpakwlliwiwdiubdlfmalkxowmwpcl',
       'channelsale_fixdbufsefwooaasfcxdxadsiekoceaa',
       'channelsale_foosdfpfkuscacimwkcsobcdxkicaua',
       'channelsale_lmkebamcaaclubfxadlmueccxoimlema',
       'channelsale_sddiedcslfslkckwlfpoeailfpeds',
       'channelsale_usilxuppasemubllopkafesmlibmsdf', 'origin_MISSING',
       'origin_ewxeelcelemmiwuafmddpobolfxioce',
       'origin_kamkkxfxxuwbdslkwifmmcsiusiuosws',
       'origin_ldkssxwpmemidmecebumciepifcamkci',
       'origin_lxidpiddsbxsbosboudacockeimpuepw',
       'origin_usapbepcfoloekilkwsdiboslwaxobdp'],
      dtype='object')
```

```
[58]: ft_out['Churn']=ft_out['churn']
```

```
[59]: ft_out.drop(columns=['churn'],axis=1,inplace=True)
```

```
[60]: ft_out.info()
```

#	Column	Non-Null Count	Dtype
---	-----	-----	-----

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 148546 entries, 24 to 175148
Data columns (total 78 columns):
 #   Column
 ---  -----

```

0	cons_12m	148546	non-null	float64
1	cons_gas_12m	148546	non-null	float64
2	cons_last_month	148546	non-null	float64
3	date_activ	148546	non-null	int64
4	date_end	148546	non-null	int64
5	date_modif_prod	148546	non-null	int64
6	date_renewal	148546	non-null	int64
7	forecast_cons_12m	148546	non-null	float64
8	forecast_cons_year	148546	non-null	float64
9	forecast_discount_energy	148546	non-null	int64
10	forecast_meter_rent_12m	148546	non-null	float64
11	forecast_price_energy_off_peak	148546	non-null	float64
12	forecast_price_energy_peak	148546	non-null	float64
13	forecast_price_pow_off_peak	148546	non-null	float64
14	has_gas	148546	non-null	int64
15	imp_cons	148546	non-null	float64
16	margin_gross_pow_ele	148546	non-null	float64
17	margin_net_pow_ele	148546	non-null	float64
18	nb_prod_act	148546	non-null	float64
19	net_margin	148546	non-null	float64
20	num_years_antig	148546	non-null	float64
21	pow_max	148546	non-null	float64
22	y_price_off_peak_var	148546	non-null	float64
23	y_price_peak_var	148546	non-null	float64
24	y_price_mid_peak_var	148546	non-null	float64
25	y_price_off_peak_fix	148546	non-null	float64
26	y_price_peak_fix	148546	non-null	float64
27	y_price_mid_peak_fix	148546	non-null	float64
28	yv_price_off_peak p1	148546	non-null	float64
29	yv_price_peak p2	148546	non-null	float64
30	yv_price_midpeak p3	148546	non-null	float64
31	6_price_off_peak_var	148546	non-null	float64
32	6_price_peak_var	148546	non-null	float64
33	6_price_mid_peak_var	148546	non-null	float64
34	6_price_off_peak_fix	148546	non-null	float64
35	6_price_peak_fix	148546	non-null	float64
36	6_price_mid_peak_fix	148546	non-null	float64
37	6mps_price_off_peak p1	148546	non-null	float64
38	6mps_price_peak p2	148546	non-null	float64
39	6mps_price_mid_peak p3	148546	non-null	float64
40	price_date	148546	non-null	int64
41	D-J_of_pk_fix	148546	non-null	float64
42	D-J_of_pk_var	148546	non-null	float64
43	mdiff_(peak-offpeak)var	148546	non-null	float64
44	mdiff_(offpeak-midpeak)var	148546	non-null	float64
45	mdiff_(mid-peak)var	148546	non-null	float64
46	mdiff_(peak-offpeak)fix	148546	non-null	float64
47	mdiff_(offpeak-midpeak)fix	148546	non-null	float64

```

48 mdiff_(mid-peak)fix           148546 non-null float64
49 max_frmpdiff_(peak-offpeak)var 148546 non-null float64
50 max_frmpmdiff_(offpeak-midpeak)var 148546 non-null float64
51 max_frmpmdiff_(mid-peak)var    148546 non-null float64
52 max_mdiff_(peak-offpeak)fix   148546 non-null float64
53 max_mdiff_(offpeak-midpeak)fix 148546 non-null float64
54 max_mdiff_(mid-peak)fix      148546 non-null float64
55 min_frmpdiff_(peak-offpeak)var 148546 non-null float64
56 min_frmpmdiff_(offpeak-midpeak)var 148546 non-null float64
57 min_frmpmdiff_(mid-peak)var    148546 non-null float64
58 min_mdiff_(peak-offpeak)fix   148546 non-null float64
59 min_mdiff_(offpeak-midpeak)fix 148546 non-null float64
60 min_mdiff_(mid-peak)fix      148546 non-null float64
61 yrs_subscribed                148546 non-null float64
62 Leadtime_renewal              148546 non-null float64
63 channelsale_MISSING            148546 non-null uint8
64 channelsale_epumfxlbckeskwekxbiuasklxalciiuu 148546 non-null uint8
65 channelsale_ewpakwlliwiwdiubdlfmalkxowmwpcl 148546 non-null uint8
66 channelsale_fixdbufsefwooaasfcxdxadsiekoceaa 148546 non-null uint8
67 channelsale_foosdfpfkusacimwkcsosbicdxkicaua 148546 non-null uint8
68 channelsale_lmkebamacaclubfxadlmueccxoimlema 148546 non-null uint8
69 channelsale_sddiedcslfslkckwlfdpoeeailfpeds 148546 non-null uint8
70 channelsale_usilxuppasemubllopkafesmlibmsdf 148546 non-null uint8
71 origin_MISSING                 148546 non-null uint8
72 origin_ewxeelcelemmiwuafmddpobolfxioce     148546 non-null uint8
73 origin_kamkkxfxxuwbdslkwifmmcsiusiuosws 148546 non-null uint8
74 origin_ldkssxwpmemidmecebumeipfcamkci     148546 non-null uint8
75 origin_lxidpiddsbxsbosboudacockeimpuepw 148546 non-null uint8
76 origin_usapbepcfoloekilkwsdiboslwaxobdp 148546 non-null uint8
77 Churn                           148546 non-null int64

dtypes: float64(56), int64(8), uint8(14)
memory usage: 75.6 MB

```

## 8 BALANCING TARGET VARIABLE CLASSES-SMOTE(Synthetic Minority Over-Sampling Technique) AND TRAIN/TEST SPLIT

```
[61]: from sklearn.model_selection import train_test_split
from sklearn.feature_selection import mutual_info_classif,SelectKBest
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler,MinMaxScaler
pd.set_option('display.max_rows',80)
from imblearn.over_sampling import SMOTE
```

```
[62]: x=ft_out.iloc[:, :-1]
y=ft_out.iloc[:, -1]
```

```
[63]: cols=x.columns.to_list()
```

```
[64]: xtrain,xtest,ytrain,ytest=train_test_split(x,y,test_size=.
      ↪25,random_state=234,shuffle=True,)

before=ytrain.value_counts().reset_index()
print(f"xtrain: {xtrain.shape} and ytrain: {ytrain.shape}")
print(f"xtest: {xtest.shape} and ytest: {ytest.shape}")

xtrain: (111409, 77) and ytrain: (111409,)
xtest: (37137, 77) and ytest: (37137,)
```

```
[65]: before
```

```
[65]:   index    Churn
0         0  100964
1         1  10445
```

```
[66]: smote=SMOTE(random_state=47854,)

xtrain,ytrain=smote.fit_resample(xtrain,ytrain)
after=ytrain.value_counts().reset_index()
print(f"xtrain: {xtrain.shape} and ytrain: {ytrain.shape}")

xtrain: (201928, 77) and ytrain: (201928,)
```

```
[67]: traincols=xtrain.columns
```

```
[68]: after
```

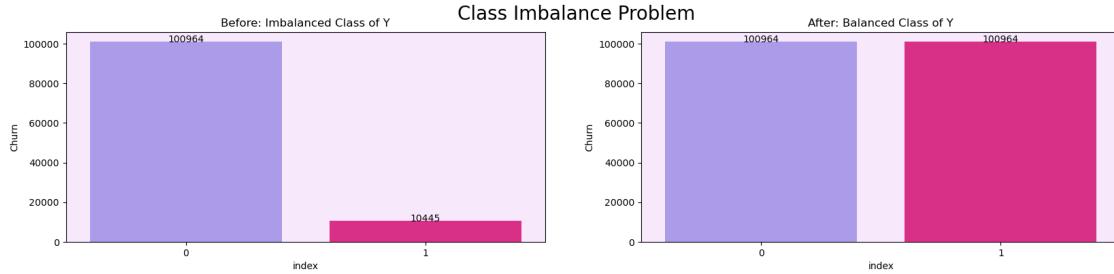
```
[68]:   index    Churn
0         0  100964
1         1  100964
```

```
[69]: fig,axs=plt.subplots(1,2,figsize=(20,4))# va    'top', 'bottom', 'center', ↴
      ↪'baseline', 'center_baseline'

a=sns.
  ↪barplot(data=before,y='Churn',x='index',ax=axs[0],palette=('#a48ef5','#f51488')) ↪
  ↪ ha  'center', 'right', 'left'
for i,v in enumerate(before['Churn']):
    axs[0].text(i,v,f"{v}",ha='center',va='baseline')

b=sns.
  ↪barplot(data=after,y='Churn',x='index',ax=axs[1],palette=('#a48ef5','#f51488')) ↪
for ind,val in enumerate(after['Churn']):
    axs[1].text(ind,val,f"{val}",ha='center',va='baseline')
a.set_title('Before: Imbalanced Class of Y')
b.set_title('After: Balanced Class of Y')
fig.suptitle(f"Class Imbalance Problem",fontsize=20)
```

```
[69]: Text(0.5, 0.98, 'Class Imbalance Problem')
```



## 9 MUTUAL INFORMATION

```
[94]: mi=mutual_info_classif(xtrain,ytrain)
midf=pd.DataFrame({'features': cols, 'Scores':mi})
```

```
[95]: c=midf.sort_values(by='Scores',ascending=False)
plt.figure(figsize=(30,70))
sns.barplot(data=c,y='features',x='Scores',palette='brg')
plt.xticks(rotation=30,fontsize=25)
plt.yticks(fontsize=25)
```

```
[95]: (array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
       17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
       34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50,
       51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67,
       68, 69, 70, 71, 72, 73, 74, 75, 76]),

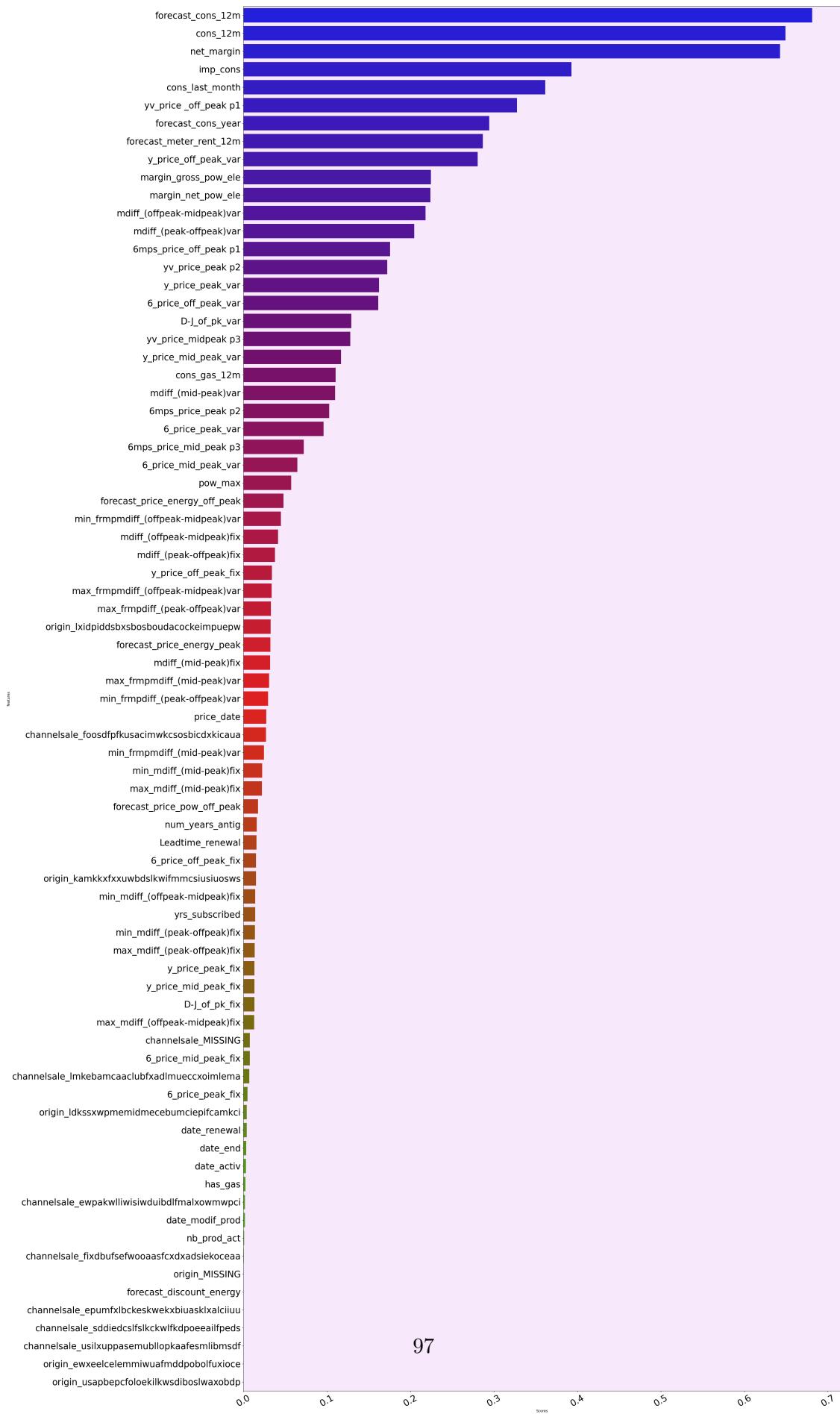
[Text(0, 0, 'forecast_cons_12m'),
 Text(0, 1, 'cons_12m'),
 Text(0, 2, 'net_margin'),
 Text(0, 3, 'imp_cons'),
 Text(0, 4, 'cons_last_month'),
 Text(0, 5, 'yv_price_off_peak p1'),
 Text(0, 6, 'forecast_cons_year'),
 Text(0, 7, 'forecast_meter_rent_12m'),
 Text(0, 8, 'y_price_off_peak_var'),
 Text(0, 9, 'margin_gross_pow_ele'),
 Text(0, 10, 'margin_net_pow_ele'),
 Text(0, 11, 'mdiff_(offpeak-midpeak)var'),
 Text(0, 12, 'mdiff_(peak-offpeak)var'),
 Text(0, 13, '6mps_price_off_peak p1'),
 Text(0, 14, 'yv_price_peak p2'),
 Text(0, 15, 'y_price_peak_var'),
 Text(0, 16, '6_price_off_peak_var'),
 Text(0, 17, 'D-J_of_pk_var'),
 Text(0, 18, 'yv_price_midpeak p3'),
```

```

Text(0, 19, 'y_price_mid_peak_var'),
Text(0, 20, 'cons_gas_12m'),
Text(0, 21, 'mdiff_(mid-peak)var'),
Text(0, 22, '6mps_price_peak p2'),
Text(0, 23, '6_price_peak_var'),
Text(0, 24, '6mps_price_mid_peak p3'),
Text(0, 25, '6_price_mid_peak_var'),
Text(0, 26, 'pow_max'),
Text(0, 27, 'forecast_price_energy_off_peak'),
Text(0, 28, 'min_frmpmdiff_(offpeak-midpeak)var'),
Text(0, 29, 'mdiff_(offpeak-midpeak)fix'),
Text(0, 30, 'mdiff_(peak-offpeak)fix'),
Text(0, 31, 'y_price_off_peak_fix'),
Text(0, 32, 'max_frmpmdiff_(offpeak-midpeak)var'),
Text(0, 33, 'max_frmpdiff_(peak-offpeak)var'),
Text(0, 34, 'origin_lxidpiddsbxsbosboudacockeimpuepw'),
Text(0, 35, 'forecast_price_energy_peak'),
Text(0, 36, 'mdiff_(mid-peak)fix'),
Text(0, 37, 'max_frmpmdiff_(mid-peak)var'),
Text(0, 38, 'min_frmpdiff_(peak-offpeak)var'),
Text(0, 39, 'price_date'),
Text(0, 40, 'channelsale_foosdfpkusacimwkcacosbickxkicaua'),
Text(0, 41, 'min_frmpmdiff_(mid-peak)var'),
Text(0, 42, 'min_mdiff_(mid-peak)fix'),
Text(0, 43, 'max_mdiff_(mid-peak)fix'),
Text(0, 44, 'forecast_price_pow_off_peak'),
Text(0, 45, 'num_years_antig'),
Text(0, 46, 'Leadtime_renewal'),
Text(0, 47, '6_price_off_peak_fix'),
Text(0, 48, 'origin_kamkxxfxuwbdslkwifmmcsiusiuosws'),
Text(0, 49, 'min_mdiff_(offpeak-midpeak)fix'),
Text(0, 50, 'yrs_subscribed'),
Text(0, 51, 'min_mdiff_(peak-offpeak)fix'),
Text(0, 52, 'max_mdiff_(peak-offpeak)fix'),
Text(0, 53, 'y_price_peak_fix'),
Text(0, 54, 'y_price_mid_peak_fix'),
Text(0, 55, 'D-J_of_pk_fix'),
Text(0, 56, 'max_mdiff_(offpeak-midpeak)fix'),
Text(0, 57, 'channelsale_MISSING'),
Text(0, 58, '6_price_mid_peak_fix'),
Text(0, 59, 'channelsale_lmkebamcaclubfxadlmueccxoimlema'),
Text(0, 60, '6_price_peak_fix'),
Text(0, 61, 'origin_ldkssxwpmemidmecebumciepifcamkci'),
Text(0, 62, 'date_renewal'),
Text(0, 63, 'date_end'),
Text(0, 64, 'date_activ'),
Text(0, 65, 'has_gas'),

```

```
Text(0, 66, 'channelsale_ewpakwlliwisiwduibdlfmalxowmwpc'),  
Text(0, 67, 'date_modif_prod'),  
Text(0, 68, 'nb_prod_act'),  
Text(0, 69, 'channelsale_fixdbufsefwooaasfcxdxadsiekoceaa'),  
Text(0, 70, 'origin_MISSING'),  
Text(0, 71, 'forecast_discount_energy'),  
Text(0, 72, 'channelsale_epumfxlbckeskwekxbiuasklxalciiuu'),  
Text(0, 73, 'channelsale_sddiedcslfslkckwlfkdpoeailfpeds'),  
Text(0, 74, 'channelsale_usilxuppasemubllopkaaafesmlibmsdf'),  
Text(0, 75, 'origin_ewxeelcelemmiwuafmddpobolfxioce'),  
Text(0, 76, 'origin_usapbepcfoloekilkwsdiboslwaxobdp'))
```



## 10 SCALING TRANSFORMATION-MINMAXSCALING

```
[70]: scaler=MinMaxScaler()  
xtrain=scaler.fit_transform(xtrain)  
xtest=scaler.fit_transform(xtest)
```

## 11 PREDICTIVE MODELLING

```
[71]: from sklearn.linear_model import LogisticRegression  
from imblearn.ensemble import BalancedRandomForestClassifier  
from sklearn.ensemble import AdaBoostClassifier, BaggingClassifier, GradientBoostingClassifier  
from sklearn.neighbors import KNeighborsClassifier  
from sklearn.model_selection import KFold, cross_val_score  
from sklearn.metrics import confusion_matrix, classification_report  
from sklearn.tree import DecisionTreeClassifier  
from sklearn.linear_model import SGDClassifier  
from sklearn import tree  
import warnings  
warnings.filterwarnings("ignore")
```

```
[72]: def kfolds(model):  
  
    #using k-folds cross validation method  
  
    k=KFold(n_splits=10,shuffle=True,random_state=3457)  
    score=cross_val_score(model,xtrain,ytrain,cv=k,)  
    print(f"\n{model}:\n CROSS VAL Scores: {score}\n Score' (mean,std)':\n {score.mean(),score.std()}" )  
  
    #model fitting for the train dataset  
  
    mf=model.fit(xtrain,ytrain)  
  
    #Predictions using model  
  
    ypred=model.predict(xtest)  
  
    #Classification Reports and Confusion Matrix for the Predictions and  
    #Actual Class  
  
    clreport=classification_report(ytest,ypred)
```

```

        print("")                                ")
print(f'CLASSIFICATION REPORT : {model}')")
print(clreport)
print("")                                ")
print(f'CONFUSION MARTIX : {model}')")
v=confusion_matrix(ytest,ypred)

#Unravelling the Confusion Matrix for to get the TP,FP, FN, TN

TP,FP,FN,TN=v.ravel()
print(f" True Positive: {TP}\n False Positive: {FP}\n False Negative:{FN}\n True Negative: {TN}")


#HEATMAP FOR CONFUSION MATRIX

sns.heatmap(v,square=True,annot=True,fmt='.'
˓if',xticklabels=['Retained','Churned'],yticklabels=['Retained','Churned'],cmap='Oranges')
plt.ylabel('Predicted')
plt.xlabel('Actual')
plt.show()

```

## BALANCED-RANDOMFOREST-CLASSIFIER

```
[80]: brf1 = BalancedRandomForestClassifier(n_estimators=55,sampling_strategy="all",
˓replacement=True,
˓max_depth=18,oob_score=True,min_samples_split=20,class_weight={0:1,1:
˓2}) #max_depth=18
kfolds(brf1)
```

```
BalancedRandomForestClassifier(class_weight={0: 1, 1: 2}, max_depth=18,
                               min_samples_split=20, n_estimators=55,
                               oob_score=True, replacement=True,
                               sampling_strategy='all'):
CROSS VAL Scores: [0.97568464 0.97934928 0.97835884 0.97781409 0.97885406
0.97825979
0.97875501 0.98009211 0.97711965 0.97558439]
Score'(mean,std)': (0.9779871871640727, 0.0014034666056468043)
```

```
CLASSIFICATION REPORT : BalancedRandomForestClassifier(class_weight={0: 1, 1:
2}, max_depth=18,
                               min_samples_split=20, n_estimators=55,
                               oob_score=True, replacement=True,
                               sampling_strategy='all')
precision    recall   f1-score   support
      0          1.00      0.96      0.98      33594
```

1	0.71	1.00	0.83	3543
accuracy			0.96	37137
macro avg	0.85	0.98	0.90	37137
weighted avg	0.97	0.96	0.96	37137

CONFUSION MATRIX : BalancedRandomForestClassifier(class\_weight={0: 1, 1: 2}, max\_depth=18,

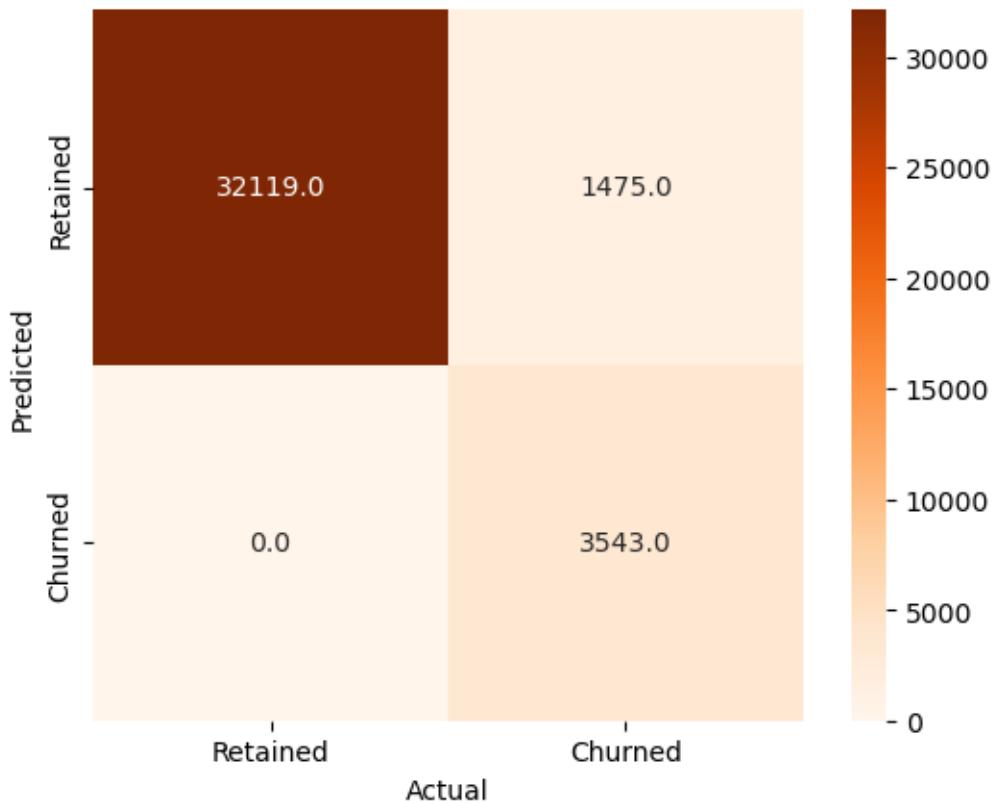
```
min_samples_split=20, n_estimators=55,
oob_score=True, replacement=True,
sampling_strategy='all')
```

True Positive: 32119

False Positive: 1475

False Negative: 0

True Negative: 3543



[81]: `print(brf1.oob_score_)`

0.9737480686185175

```
[77]: b=brf1.feature_importances_
feature_names=cols
plt.figure(figsize=(10,19))
sns.barplot(x=b,y=cols)
```

[77]: <Axes: >



```
[86]: import pickle
```

```
[87]: pickle.dump(brf1,open('brf1.pkl','wb'))
```

## GAUSSIAN NAIVE BAYES

```
[161]: from sklearn.naive_bayes import GaussianNB
```

```
gnb=GaussianNB(var_smoothing=1e-3)
kfolds(gnb)
```

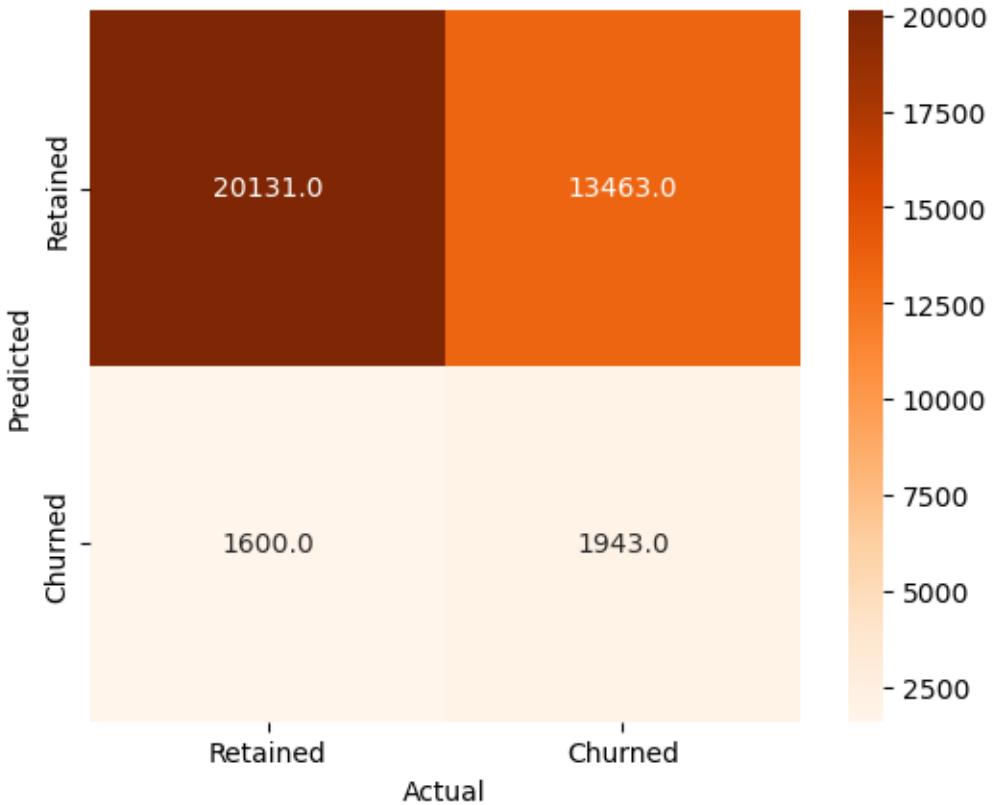
```
GaussianNB(var_smoothing=0.001):
    CROSS VAL Scores: [0.56876145 0.57574407 0.56583965 0.57703164 0.57430793
0.56772149
0.57445649 0.55628188 0.57512876 0.56878962]
Score'(mean,std)': (0.5704062988084162, 0.005993299450700824)
```

```
CLASSIFICATION REPORT : GaussianNB(var_smoothing=0.001)
```

	precision	recall	f1-score	support
0	0.93	0.60	0.73	33594
1	0.13	0.55	0.21	3543
accuracy			0.59	37137
macro avg	0.53	0.57	0.47	37137
weighted avg	0.85	0.59	0.68	37137

```
CONFUSION MARTIX : GaussianNB(var_smoothing=0.001)
```

```
True Positive: 20131
False Positive: 13463
False Negative: 1600
True Negative: 1943
```



## KNEIGHBORS CLASSIFIER

```
[89]: knc=KNeighborsClassifier(n_neighbors=4, weights='uniform' , algorithm="auto",
    ↪leaf_size=5, p=1, metric="minkowski",
    ↪metric_params=None)
#Power parameter for the Minkowski metric. When p = 1, this is equivalent to
    ↪using manhattan_distance (l1),
#and euclidean_distance (l2) for p = 2. For arbitrary p, minkowski_distance
    ↪(l_p) is used
#Algorithm used to compute the nearest neighbors:

# * 'ball_tree' will use BallTree

# * 'kd_tree' will use KDTree

# * 'brute' will use a brute-force search.

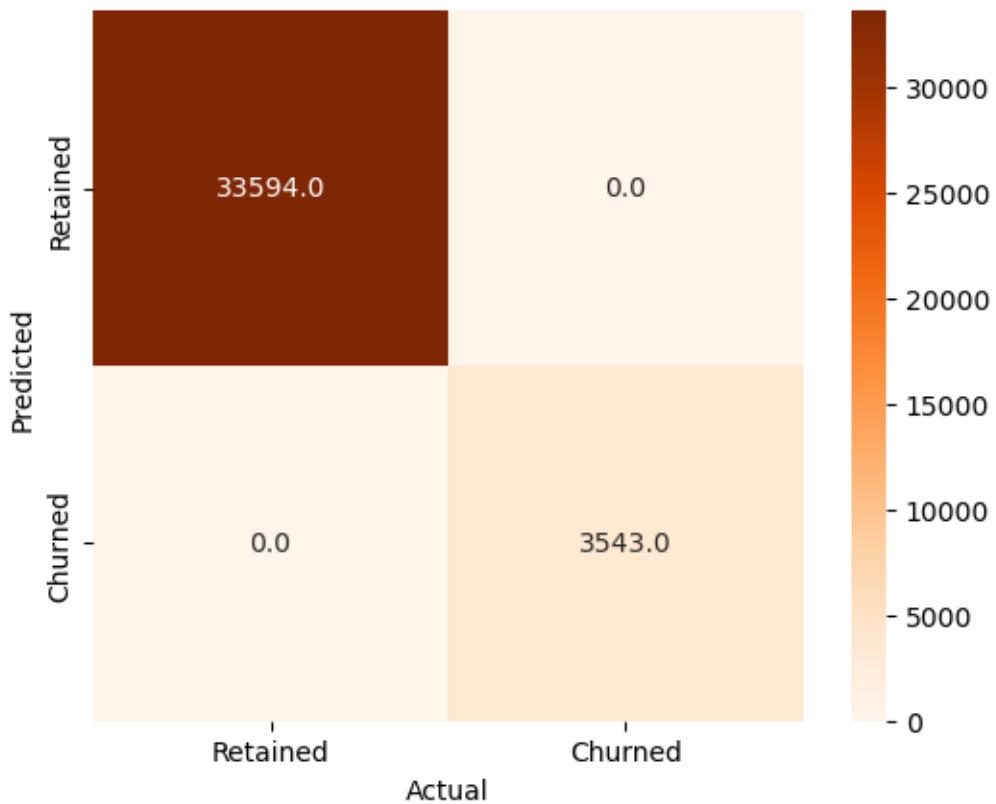
# * 'auto' will attempt to decide the most appropriate algorithm based on the
    ↪values passed to fit method.

kfolds(knc)
```

```
KNeighborsClassifier(leaf_size=5, n_neighbors=4, p=1):  
CROSS VAL Scores: [1. 1. 1. 1. 1. 1. 1. 1. 1.]  
Score'(mean,std)': (1.0, 0.0)
```

```
CLASSIFICATION REPORT : KNeighborsClassifier(leaf_size=5, n_neighbors=4, p=1)  
precision recall f1-score support  
  
0 1.00 1.00 1.00 33594  
1 1.00 1.00 1.00 3543  
  
accuracy 1.00 37137  
macro avg 1.00 1.00 1.00 37137  
weighted avg 1.00 1.00 1.00 37137
```

```
CONFUSION MATRIX : KNeighborsClassifier(leaf_size=5, n_neighbors=4, p=1)  
True Positive: 33594  
False Positive: 0  
False Negative: 0  
True Negative: 3543
```



```
[90]: pickle.dump(knc,open('knearestneighbors clf.pkl','wb'))
```

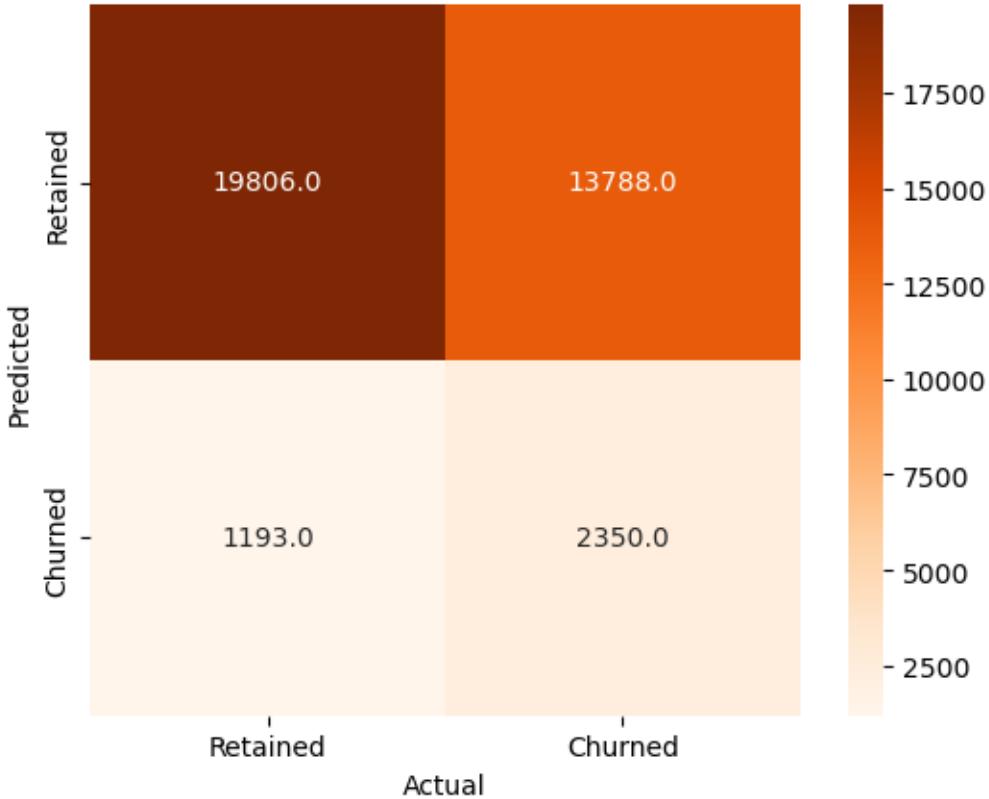
## SGD CLASSIFIER

```
[119]: sdg=SGDClassifier(loss='log_loss', penalty='elasticnet', alpha=0.001,
    ↵l1_ratio=0.15, fit_intercept=True, max_iter=10000,
    ↵          #log_loss--> gives logistic regression, a probabilistic
    ↵classifier.
    ↵tol=0.001, shuffle=True, epsilon=0.1, learning_rate='optimal',
    ↵eta0=0.0, power_t=0.5,
    ↵validation_fraction=0.2, n_iter_no_change=5, )
kfolds(sdg)
```

```
SGDClassifier(alpha=0.001, loss='log_loss', max_iter=10000,
    ↵penalty='elasticnet', validation_fraction=0.2):
CROSS VAL Scores: [0.62645471 0.62982222 0.61659981 0.63026791 0.62417669
0.62353291
0.61417323 0.62630615 0.61920563 0.62727813]
Score'(mean,std)': (0.623781738640895, 0.005190012218755415)
```

```
CLASSIFICATION REPORT : SGDClassifier(alpha=0.001, loss='log_loss',
max_iter=10000,
    ↵penalty='elasticnet', validation_fraction=0.2)
precision      recall   f1-score   support
0            0.94     0.59      0.73     33594
1            0.15     0.66      0.24     3543
accuracy                          0.60     37137
macro avg                  0.54     0.63      0.48     37137
weighted avg                 0.87     0.60      0.68     37137
```

```
CONFUSION MARTIX : SGDClassifier(alpha=0.001, loss='log_loss', max_iter=10000,
    ↵penalty='elasticnet', validation_fraction=0.2)
True Positive: 19806
False Positive: 13788
False Negative: 1193
True Negative: 2350
```



## GRADINET BOOSTING CLASSIFIER

```
[91]: gbc=GradientBoostingClassifier(loss="log_loss", learning_rate=0.01,
                                     n_estimators=50, subsample=1.0,
                                     criterion="friedman_mse",
                                     min_samples_split=2,max_features='sqrt',
                                     min_samples_leaf=15, max_depth=15,
                                     validation_fraction=0.2, tol=1e-4, ccp_alpha=0.0)
kfolds(gbc)
```

```
GradientBoostingClassifier(learning_rate=0.01, max_depth=15,
                          max_features='sqrt', min_samples_leaf=15,
                          n_estimators=50, validation_fraction=0.2):
CROSS VAL Scores: [0.99281929 0.99286882 0.99301738 0.99172981 0.99262121
0.99093745
0.99331451 0.99385926 0.99316561 0.99182845]
Score'(mean,std)': (0.9926161788444363, 0.0008261452296556004)
```

```
CLASSIFICATION REPORT : GradientBoostingClassifier(learning_rate=0.01,
max_depth=15,
max_features='sqrt', min_samples_leaf=15,
n_estimators=50, validation_fraction=0.2)
```

	precision	recall	f1-score	support
0	1.00	0.99	0.99	33594
1	0.88	0.99	0.93	3543
accuracy			0.99	37137
macro avg	0.94	0.99	0.96	37137
weighted avg	0.99	0.99	0.99	37137

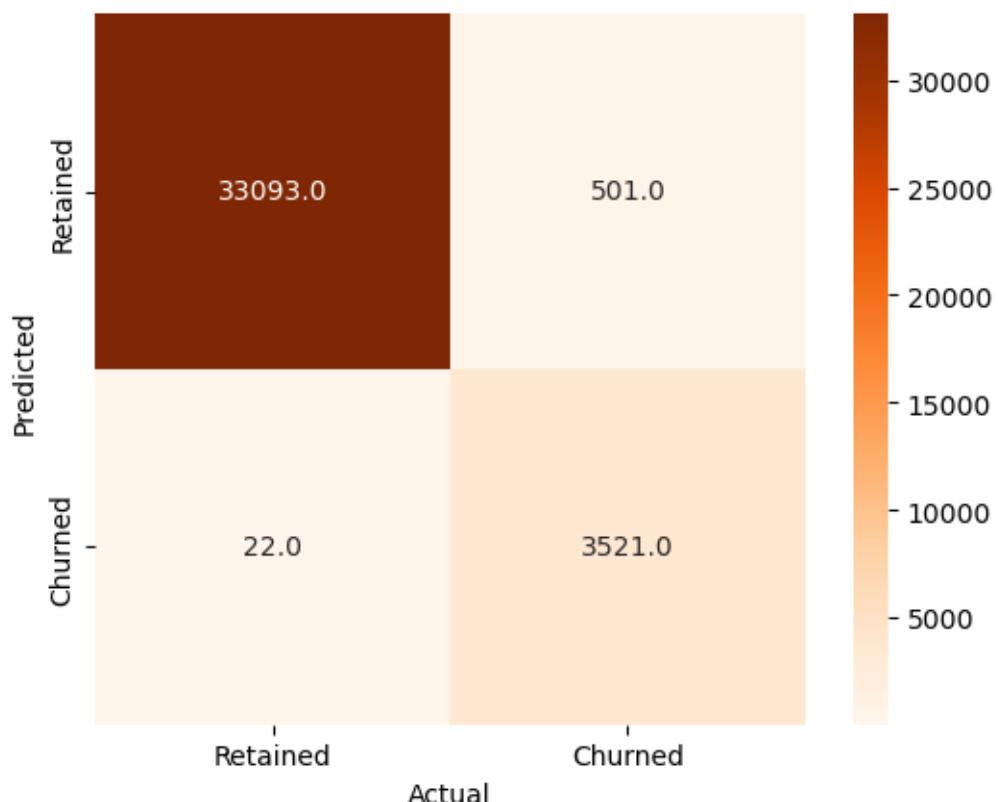
```
CONFUSION MARTIX : GradientBoostingClassifier(learning_rate=0.01, max_depth=15,
                                             max_features='sqrt', min_samples_leaf=15,
                                             n_estimators=50, validation_fraction=0.2)
```

True Positive: 33093

False Positive: 501

False Negative: 22

True Negative: 3521



```
[92]: pickle.dump(gbc,open('GradientBoost clf.pkl','wb'))
```

## LINEAR SUPPORT VECTOR CLASSIFICATION

```
[176]: from sklearn.svm import LinearSVC

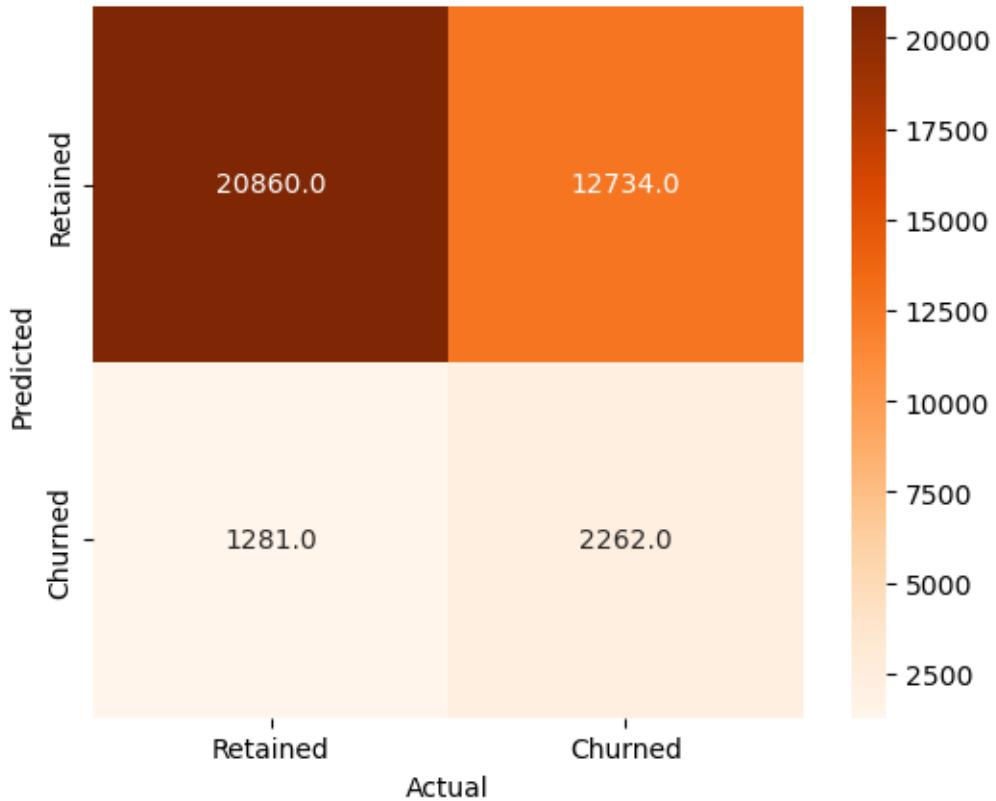
svc=LinearSVC(penalty="l2", loss="squared_hinge", dual=True, tol=1e-2, C=0.01, u
    ↪multi_class="ovr",
           fit_intercept=True, intercept_scaling=1, max_iter=1000)
kfolds(svc)
```

```
LinearSVC(C=0.01, tol=0.01):
CROSS VAL Scores: [0.62630615 0.62917843 0.6176893 0.62853464 0.63031744
0.62655376
0.62075967 0.63140692 0.62717908 0.62579239]
Score'(mean,std)': (0.6263717772073185, 0.004017240822204915)
```

```
CLASSIFICATION REPORT : LinearSVC(C=0.01, tol=0.01)
precision      recall   f1-score   support
0            0.94      0.62      0.75     33594
1            0.15      0.64      0.24      3543

accuracy                  0.62     37137
macro avg                 0.55      0.63      0.50     37137
weighted avg                0.87      0.62      0.70     37137
```

```
CONFUSION MARTIX : LinearSVC(C=0.01, tol=0.01)
True Positive: 20860
False Positive: 12734
False Negative: 1281
True Negative: 2262
```



## EXTREME GRADIENT BOOSTING CLASSIFIER

```
[95]: import xgboost as xgb
xg=xgb.XGBClassifier(n_estimators=55,learning_rate=0.
 ↪01,max_depth=20,max_leaves=3)
kfolds(xg)

XGBClassifier(base_score=None, booster=None, callbacks=None,
              colsample_bylevel=None, colsample_bynode=None,
              colsample_bytree=None, early_stopping_rounds=None,
              enable_categorical=False, eval_metric=None, feature_types=None,
              gamma=None, gpu_id=None, grow_policy=None, importance_type=None,
              interaction_constraints=None, learning_rate=0.01, max_bin=None,
              max_cat_threshold=None, max_cat_to_onehot=None,
              max_delta_step=None, max_depth=20, max_leaves=3,
              min_child_weight=None, missing=nan, monotone_constraints=None,
              n_estimators=55, n_jobs=None, num_parallel_tree=None,
              predictor=None, random_state=None, ...):
CROSS VAL Scores: [0.98940227 0.9901451 0.99083841 0.99029367 0.99366117
0.98667855
0.99301738 0.99257168 0.99187797 0.98776743]
Score'(mean,std)': (0.9906253634695394, 0.0021429460487221866)
```

```

CLASSIFICATION REPORT : XGBClassifier(base_score=None, booster=None,
callbacks=None,
    colsample_bylevel=None, colsample_bynode=None,
    colsample_bytree=None, early_stopping_rounds=None,
    enable_categorical=False, eval_metric=None, feature_types=None,
    gamma=None, gpu_id=None, grow_policy=None, importance_type=None,
    interaction_constraints=None, learning_rate=0.01, max_bin=None,
    max_cat_threshold=None, max_cat_to_onehot=None,
    max_delta_step=None, max_depth=20, max_leaves=3,
    min_child_weight=None, missing=nan, monotone_constraints=None,
    n_estimators=55, n_jobs=None, num_parallel_tree=None,
    predictor=None, random_state=None, ...)
precision      recall   f1-score   support

0            1.00     0.99     0.99    33594
1            0.87     0.97     0.92    3543

accuracy                      0.98    37137
macro avg          0.93     0.98     0.95    37137
weighted avg         0.98     0.98     0.98    37137

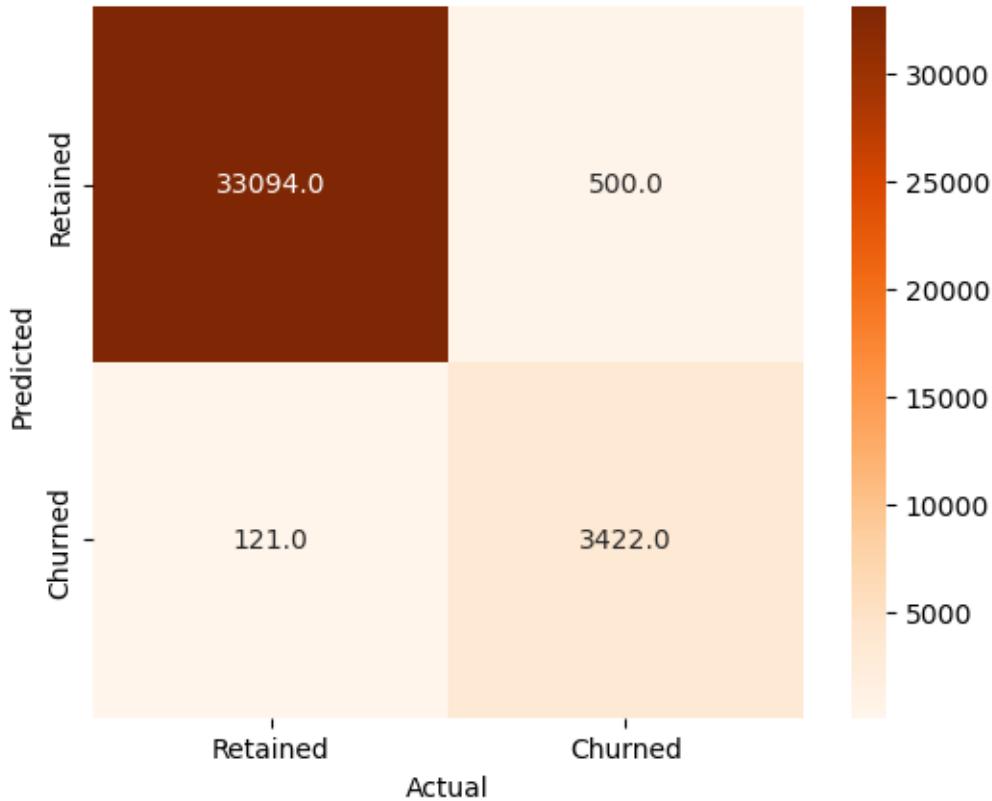
```

```

CONFUSION MARTIX : XGBClassifier(base_score=None, booster=None, callbacks=None,
    colsample_bylevel=None, colsample_bynode=None,
    colsample_bytree=None, early_stopping_rounds=None,
    enable_categorical=False, eval_metric=None, feature_types=None,
    gamma=None, gpu_id=None, grow_policy=None, importance_type=None,
    interaction_constraints=None, learning_rate=0.01, max_bin=None,
    max_cat_threshold=None, max_cat_to_onehot=None,
    max_delta_step=None, max_depth=20, max_leaves=3,
    min_child_weight=None, missing=nan, monotone_constraints=None,
    n_estimators=55, n_jobs=None, num_parallel_tree=None,
    predictor=None, random_state=None, ...)

True Positive: 33094
False Positive: 500
False Negative: 121
True Negative: 3422

```



```
[96]: pickle.dump(xg,open('EXTREMEGRADIENTBOOST CLG.pkl','wb'))
```

## CATEGORICAL BOOST CLASSIFIER

```
cat=CatBoostClassifier( learning_rate=0.01, depth=15, n_estimators=50, loss_function = "CrossEntropy") kfolds(cat)
```

```
[98]: from catboost import CatBoostClassifier
cat=CatBoostClassifier( learning_rate=0.01, depth=16, n_estimators=50, loss_function = "CrossEntropy")
kfolds(cat)# HERE MAXIMUM TREE DEPTH IS 16
```

```
0:    learn: 0.6818199      total: 6.21s      remaining: 5m 4s
1:    learn: 0.6712011      total: 12.3s      remaining: 4m 54s
2:    learn: 0.6613981      total: 18.4s      remaining: 4m 48s
3:    learn: 0.6498606      total: 24.5s      remaining: 4m 41s
4:    learn: 0.6381773      total: 30.6s      remaining: 4m 35s
5:    learn: 0.6285580      total: 36.8s      remaining: 4m 29s
6:    learn: 0.6193600      total: 42.8s      remaining: 4m 23s
7:    learn: 0.6082058      total: 49.2s      remaining: 4m 18s
8:    learn: 0.5976458      total: 55.3s      remaining: 4m 12s
9:    learn: 0.5886970      total: 1m 1s      remaining: 4m 5s
```

10:	learn: 0.5812234	total: 1m 7s	remaining: 3m 59s
11:	learn: 0.5715231	total: 1m 13s	remaining: 3m 52s
12:	learn: 0.5636522	total: 1m 19s	remaining: 3m 46s
13:	learn: 0.5564486	total: 1m 25s	remaining: 3m 39s
14:	learn: 0.5484460	total: 1m 31s	remaining: 3m 33s
15:	learn: 0.5410213	total: 1m 37s	remaining: 3m 27s
16:	learn: 0.5323220	total: 1m 43s	remaining: 3m 21s
17:	learn: 0.5236552	total: 1m 49s	remaining: 3m 15s
18:	learn: 0.5153730	total: 1m 56s	remaining: 3m 9s
19:	learn: 0.5088242	total: 2m 2s	remaining: 3m 3s
20:	learn: 0.5027914	total: 2m 8s	remaining: 2m 57s
21:	learn: 0.4967506	total: 2m 14s	remaining: 2m 51s
22:	learn: 0.4899990	total: 2m 20s	remaining: 2m 44s
23:	learn: 0.4846651	total: 2m 26s	remaining: 2m 38s
24:	learn: 0.4769079	total: 2m 32s	remaining: 2m 32s
25:	learn: 0.4698849	total: 2m 38s	remaining: 2m 26s
26:	learn: 0.4642332	total: 2m 44s	remaining: 2m 20s
27:	learn: 0.4586937	total: 2m 50s	remaining: 2m 14s
28:	learn: 0.4519820	total: 2m 56s	remaining: 2m 8s
29:	learn: 0.4450897	total: 3m 3s	remaining: 2m 2s
30:	learn: 0.4392840	total: 3m 9s	remaining: 1m 55s
31:	learn: 0.4334411	total: 3m 15s	remaining: 1m 49s
32:	learn: 0.4293811	total: 3m 21s	remaining: 1m 43s
33:	learn: 0.4242632	total: 3m 27s	remaining: 1m 37s
34:	learn: 0.4195727	total: 3m 33s	remaining: 1m 31s
35:	learn: 0.4128433	total: 3m 39s	remaining: 1m 25s
36:	learn: 0.4078785	total: 3m 45s	remaining: 1m 19s
37:	learn: 0.4032803	total: 3m 51s	remaining: 1m 13s
38:	learn: 0.3981256	total: 3m 58s	remaining: 1m 7s
39:	learn: 0.3938377	total: 4m 4s	remaining: 1m 1s
40:	learn: 0.3890707	total: 4m 10s	remaining: 55s
41:	learn: 0.3839864	total: 4m 16s	remaining: 48.9s
42:	learn: 0.3787943	total: 4m 22s	remaining: 42.8s
43:	learn: 0.3751745	total: 4m 29s	remaining: 36.7s
44:	learn: 0.3708402	total: 4m 35s	remaining: 30.6s
45:	learn: 0.3671644	total: 4m 41s	remaining: 24.5s
46:	learn: 0.3629840	total: 4m 47s	remaining: 18.4s
47:	learn: 0.3599079	total: 4m 53s	remaining: 12.2s
48:	learn: 0.3554302	total: 5m	remaining: 6.13s
49:	learn: 0.3505795	total: 5m 6s	remaining: 0us
0:	learn: 0.6821916	total: 6.27s	remaining: 5m 7s
1:	learn: 0.6721180	total: 12.4s	remaining: 4m 58s
2:	learn: 0.6616619	total: 18.7s	remaining: 4m 52s
3:	learn: 0.6499725	total: 25s	remaining: 4m 47s
4:	learn: 0.6399369	total: 31.2s	remaining: 4m 41s
5:	learn: 0.6295798	total: 37.6s	remaining: 4m 36s
6:	learn: 0.6198512	total: 43.9s	remaining: 4m 29s
7:	learn: 0.6104173	total: 50.1s	remaining: 4m 23s

8:	learn: 0.5997155	total: 56.4s	remaining: 4m 16s
9:	learn: 0.5902401	total: 1m 2s	remaining: 4m 9s
10:	learn: 0.5812356	total: 1m 8s	remaining: 4m 3s
11:	learn: 0.5732193	total: 1m 14s	remaining: 3m 56s
12:	learn: 0.5651329	total: 1m 21s	remaining: 3m 50s
13:	learn: 0.5575470	total: 1m 27s	remaining: 3m 44s
14:	learn: 0.5479834	total: 1m 33s	remaining: 3m 37s
15:	learn: 0.5404846	total: 1m 39s	remaining: 3m 31s
16:	learn: 0.5317349	total: 1m 45s	remaining: 3m 25s
17:	learn: 0.5235039	total: 1m 51s	remaining: 3m 19s
18:	learn: 0.5159340	total: 1m 58s	remaining: 3m 12s
19:	learn: 0.5082963	total: 2m 4s	remaining: 3m 6s
20:	learn: 0.5009117	total: 2m 10s	remaining: 3m
21:	learn: 0.4955811	total: 2m 16s	remaining: 2m 54s
22:	learn: 0.4883078	total: 2m 23s	remaining: 2m 48s
23:	learn: 0.4829799	total: 2m 29s	remaining: 2m 41s
24:	learn: 0.4763559	total: 2m 35s	remaining: 2m 35s
25:	learn: 0.4693716	total: 2m 41s	remaining: 2m 29s
26:	learn: 0.4632022	total: 2m 48s	remaining: 2m 23s
27:	learn: 0.4574993	total: 2m 54s	remaining: 2m 16s
28:	learn: 0.4516600	total: 3m	remaining: 2m 10s
29:	learn: 0.4449828	total: 3m 6s	remaining: 2m 4s
30:	learn: 0.4387790	total: 3m 12s	remaining: 1m 57s
31:	learn: 0.4329256	total: 3m 18s	remaining: 1m 51s
32:	learn: 0.4284456	total: 3m 24s	remaining: 1m 45s
33:	learn: 0.4239983	total: 3m 30s	remaining: 1m 39s
34:	learn: 0.4180046	total: 3m 37s	remaining: 1m 33s
35:	learn: 0.4132428	total: 3m 43s	remaining: 1m 26s
36:	learn: 0.4078550	total: 3m 49s	remaining: 1m 20s
37:	learn: 0.4034283	total: 3m 55s	remaining: 1m 14s
38:	learn: 0.3978145	total: 4m 1s	remaining: 1m 8s
39:	learn: 0.3922262	total: 4m 7s	remaining: 1m 1s
40:	learn: 0.3871898	total: 4m 13s	remaining: 55.7s
41:	learn: 0.3825669	total: 4m 20s	remaining: 49.5s
42:	learn: 0.3777783	total: 4m 26s	remaining: 43.3s
43:	learn: 0.3741863	total: 4m 32s	remaining: 37.2s
44:	learn: 0.3704923	total: 4m 38s	remaining: 31s
45:	learn: 0.3660292	total: 4m 44s	remaining: 24.8s
46:	learn: 0.3609482	total: 4m 51s	remaining: 18.6s
47:	learn: 0.3563182	total: 4m 57s	remaining: 12.4s
48:	learn: 0.3523280	total: 5m 3s	remaining: 6.19s
49:	learn: 0.3490284	total: 5m 9s	remaining: 0us
0:	learn: 0.6820190	total: 6.16s	remaining: 5m 1s
1:	learn: 0.6724004	total: 12.2s	remaining: 4m 52s
2:	learn: 0.6620346	total: 18.6s	remaining: 4m 50s
3:	learn: 0.6505677	total: 24.8s	remaining: 4m 44s
4:	learn: 0.6403357	total: 31s	remaining: 4m 39s
5:	learn: 0.6299431	total: 37.2s	remaining: 4m 32s

6:	learn: 0.6213237	total: 43.1s	remaining: 4m 25s
7:	learn: 0.6101571	total: 49.3s	remaining: 4m 18s
8:	learn: 0.5995252	total: 55.5s	remaining: 4m 12s
9:	learn: 0.5903491	total: 1m 1s	remaining: 4m 6s
10:	learn: 0.5808631	total: 1m 8s	remaining: 4m 1s
11:	learn: 0.5713391	total: 1m 14s	remaining: 3m 55s
12:	learn: 0.5634478	total: 1m 20s	remaining: 3m 49s
13:	learn: 0.5558706	total: 1m 26s	remaining: 3m 43s
14:	learn: 0.5472504	total: 1m 33s	remaining: 3m 37s
15:	learn: 0.5395527	total: 1m 39s	remaining: 3m 30s
16:	learn: 0.5307871	total: 1m 45s	remaining: 3m 24s
17:	learn: 0.5222565	total: 1m 51s	remaining: 3m 18s
18:	learn: 0.5142174	total: 1m 57s	remaining: 3m 12s
19:	learn: 0.5076261	total: 2m 3s	remaining: 3m 5s
20:	learn: 0.5013569	total: 2m 10s	remaining: 2m 59s
21:	learn: 0.4940933	total: 2m 16s	remaining: 2m 53s
22:	learn: 0.4870146	total: 2m 22s	remaining: 2m 47s
23:	learn: 0.4791065	total: 2m 28s	remaining: 2m 41s
24:	learn: 0.4718424	total: 2m 34s	remaining: 2m 34s
25:	learn: 0.4648462	total: 2m 40s	remaining: 2m 28s
26:	learn: 0.4592891	total: 2m 47s	remaining: 2m 22s
27:	learn: 0.4521998	total: 2m 53s	remaining: 2m 16s
28:	learn: 0.4464584	total: 2m 59s	remaining: 2m 9s
29:	learn: 0.4401136	total: 3m 5s	remaining: 2m 3s
30:	learn: 0.4345316	total: 3m 11s	remaining: 1m 57s
31:	learn: 0.4280016	total: 3m 17s	remaining: 1m 51s
32:	learn: 0.4213697	total: 3m 23s	remaining: 1m 44s
33:	learn: 0.4166494	total: 3m 29s	remaining: 1m 38s
34:	learn: 0.4123580	total: 3m 35s	remaining: 1m 32s
35:	learn: 0.4061282	total: 3m 41s	remaining: 1m 26s
36:	learn: 0.4015898	total: 3m 48s	remaining: 1m 20s
37:	learn: 0.3973616	total: 3m 54s	remaining: 1m 13s
38:	learn: 0.3919291	total: 4m	remaining: 1m 7s
39:	learn: 0.3875964	total: 4m 6s	remaining: 1m 1s
40:	learn: 0.3830754	total: 4m 12s	remaining: 55.5s
41:	learn: 0.3776925	total: 4m 18s	remaining: 49.3s
42:	learn: 0.3725265	total: 4m 24s	remaining: 43.1s
43:	learn: 0.3685566	total: 4m 31s	remaining: 37s
44:	learn: 0.3645662	total: 4m 37s	remaining: 30.8s
45:	learn: 0.3597957	total: 4m 43s	remaining: 24.6s
46:	learn: 0.3561889	total: 4m 49s	remaining: 18.5s
47:	learn: 0.3528184	total: 4m 55s	remaining: 12.3s
48:	learn: 0.3491072	total: 5m 1s	remaining: 6.16s
49:	learn: 0.3456436	total: 5m 7s	remaining: 0us
0:	learn: 0.6829268	total: 6.11s	remaining: 4m 59s
1:	learn: 0.6726755	total: 12.3s	remaining: 4m 55s
2:	learn: 0.6622696	total: 18.8s	remaining: 4m 54s
3:	learn: 0.6501447	total: 25.1s	remaining: 4m 48s

4:	learn: 0.6404610	total: 31.2s	remaining: 4m 40s
5:	learn: 0.6301463	total: 37.5s	remaining: 4m 34s
6:	learn: 0.6203678	total: 43.9s	remaining: 4m 29s
7:	learn: 0.6092629	total: 50.5s	remaining: 4m 25s
8:	learn: 0.5991991	total: 56.8s	remaining: 4m 18s
9:	learn: 0.5902545	total: 1m 3s	remaining: 4m 13s
10:	learn: 0.5813654	total: 1m 9s	remaining: 4m 7s
11:	learn: 0.5714369	total: 1m 15s	remaining: 4m
12:	learn: 0.5640408	total: 1m 22s	remaining: 3m 54s
13:	learn: 0.5554966	total: 1m 28s	remaining: 3m 48s
14:	learn: 0.5478864	total: 1m 35s	remaining: 3m 41s
15:	learn: 0.5393994	total: 1m 41s	remaining: 3m 35s
16:	learn: 0.5307878	total: 1m 47s	remaining: 3m 28s
17:	learn: 0.5229420	total: 1m 53s	remaining: 3m 22s
18:	learn: 0.5151624	total: 2m	remaining: 3m 16s
19:	learn: 0.5077397	total: 2m 6s	remaining: 3m 9s
20:	learn: 0.5000790	total: 2m 12s	remaining: 3m 3s
21:	learn: 0.4936643	total: 2m 18s	remaining: 2m 56s
22:	learn: 0.4873359	total: 2m 25s	remaining: 2m 50s
23:	learn: 0.4802959	total: 2m 31s	remaining: 2m 44s
24:	learn: 0.4745653	total: 2m 38s	remaining: 2m 38s
25:	learn: 0.4671469	total: 2m 44s	remaining: 2m 31s
26:	learn: 0.4610586	total: 2m 50s	remaining: 2m 25s
27:	learn: 0.4545559	total: 2m 57s	remaining: 2m 19s
28:	learn: 0.4497023	total: 3m 3s	remaining: 2m 12s
29:	learn: 0.4429568	total: 3m 9s	remaining: 2m 6s
30:	learn: 0.4371976	total: 3m 15s	remaining: 2m
31:	learn: 0.4306260	total: 3m 22s	remaining: 1m 53s
32:	learn: 0.4256432	total: 3m 28s	remaining: 1m 47s
33:	learn: 0.4211475	total: 3m 35s	remaining: 1m 41s
34:	learn: 0.4163468	total: 3m 41s	remaining: 1m 34s
35:	learn: 0.4099982	total: 3m 47s	remaining: 1m 28s
36:	learn: 0.4048509	total: 3m 54s	remaining: 1m 22s
37:	learn: 0.3998434	total: 4m	remaining: 1m 16s
38:	learn: 0.3955958	total: 4m 7s	remaining: 1m 9s
39:	learn: 0.3907379	total: 4m 13s	remaining: 1m 3s
40:	learn: 0.3854493	total: 4m 19s	remaining: 57.1s
41:	learn: 0.3807538	total: 4m 26s	remaining: 50.7s
42:	learn: 0.3764228	total: 4m 32s	remaining: 44.4s
43:	learn: 0.3719924	total: 4m 39s	remaining: 38s
44:	learn: 0.3680204	total: 4m 45s	remaining: 31.7s
45:	learn: 0.3640204	total: 4m 51s	remaining: 25.4s
46:	learn: 0.3596577	total: 4m 58s	remaining: 19s
47:	learn: 0.3557419	total: 5m 4s	remaining: 12.7s
48:	learn: 0.3517784	total: 5m 11s	remaining: 6.35s
49:	learn: 0.3486230	total: 5m 17s	remaining: 0us
0:	learn: 0.6821542	total: 6.27s	remaining: 5m 7s
1:	learn: 0.6729668	total: 12.6s	remaining: 5m 3s

2:	learn: 0.6616430	total: 18.7s	remaining: 4m 52s
3:	learn: 0.6495601	total: 25s	remaining: 4m 47s
4:	learn: 0.6379955	total: 31.3s	remaining: 4m 42s
5:	learn: 0.6267998	total: 37.7s	remaining: 4m 36s
6:	learn: 0.6169104	total: 44.3s	remaining: 4m 32s
7:	learn: 0.6060453	total: 50.7s	remaining: 4m 25s
8:	learn: 0.5959603	total: 56.9s	remaining: 4m 19s
9:	learn: 0.5873970	total: 1m 3s	remaining: 4m 12s
10:	learn: 0.5782895	total: 1m 9s	remaining: 4m 5s
11:	learn: 0.5686302	total: 1m 15s	remaining: 3m 59s
12:	learn: 0.5604164	total: 1m 22s	remaining: 3m 53s
13:	learn: 0.5532598	total: 1m 28s	remaining: 3m 47s
14:	learn: 0.5441626	total: 1m 34s	remaining: 3m 41s
15:	learn: 0.5363319	total: 1m 41s	remaining: 3m 34s
16:	learn: 0.5276564	total: 1m 47s	remaining: 3m 28s
17:	learn: 0.5205621	total: 1m 53s	remaining: 3m 22s
18:	learn: 0.5130065	total: 2m	remaining: 3m 16s
19:	learn: 0.5062501	total: 2m 6s	remaining: 3m 9s
20:	learn: 0.5001326	total: 2m 12s	remaining: 3m 3s
21:	learn: 0.4941314	total: 2m 19s	remaining: 2m 56s
22:	learn: 0.4875398	total: 2m 25s	remaining: 2m 50s
23:	learn: 0.4804905	total: 2m 31s	remaining: 2m 44s
24:	learn: 0.4729366	total: 2m 37s	remaining: 2m 37s
25:	learn: 0.4650891	total: 2m 44s	remaining: 2m 31s
26:	learn: 0.4594356	total: 2m 50s	remaining: 2m 25s
27:	learn: 0.4533724	total: 2m 56s	remaining: 2m 18s
28:	learn: 0.4467240	total: 3m 3s	remaining: 2m 12s
29:	learn: 0.4394005	total: 3m 9s	remaining: 2m 6s
30:	learn: 0.4339785	total: 3m 15s	remaining: 1m 59s
31:	learn: 0.4273462	total: 3m 21s	remaining: 1m 53s
32:	learn: 0.4216972	total: 3m 27s	remaining: 1m 46s
33:	learn: 0.4165592	total: 3m 33s	remaining: 1m 40s
34:	learn: 0.4110304	total: 3m 40s	remaining: 1m 34s
35:	learn: 0.4055216	total: 3m 46s	remaining: 1m 28s
36:	learn: 0.4004959	total: 3m 52s	remaining: 1m 21s
37:	learn: 0.3962806	total: 3m 59s	remaining: 1m 15s
38:	learn: 0.3925136	total: 4m 5s	remaining: 1m 9s
39:	learn: 0.3878809	total: 4m 12s	remaining: 1m 3s
40:	learn: 0.3827152	total: 4m 18s	remaining: 56.8s
41:	learn: 0.3772215	total: 4m 24s	remaining: 50.5s
42:	learn: 0.3729839	total: 4m 31s	remaining: 44.2s
43:	learn: 0.3683662	total: 4m 37s	remaining: 37.9s
44:	learn: 0.3642582	total: 4m 43s	remaining: 31.5s
45:	learn: 0.3594967	total: 4m 50s	remaining: 25.3s
46:	learn: 0.3549999	total: 4m 56s	remaining: 18.9s
47:	learn: 0.3515596	total: 5m 3s	remaining: 12.6s
48:	learn: 0.3471065	total: 5m 9s	remaining: 6.31s
49:	learn: 0.3438273	total: 5m 15s	remaining: 0us

0:	learn: 0.6824154	total: 6.12s	remaining: 4m 59s
1:	learn: 0.6715066	total: 12.3s	remaining: 4m 55s
2:	learn: 0.6606229	total: 18.5s	remaining: 4m 50s
3:	learn: 0.6489092	total: 24.8s	remaining: 4m 44s
4:	learn: 0.6387222	total: 31.1s	remaining: 4m 40s
5:	learn: 0.6289352	total: 37.4s	remaining: 4m 34s
6:	learn: 0.6192140	total: 43.6s	remaining: 4m 28s
7:	learn: 0.6081146	total: 49.9s	remaining: 4m 22s
8:	learn: 0.5991267	total: 56.2s	remaining: 4m 15s
9:	learn: 0.5886971	total: 1m 2s	remaining: 4m 10s
10:	learn: 0.5807725	total: 1m 8s	remaining: 4m 4s
11:	learn: 0.5713224	total: 1m 15s	remaining: 3m 58s
12:	learn: 0.5628152	total: 1m 21s	remaining: 3m 52s
13:	learn: 0.5537396	total: 1m 28s	remaining: 3m 47s
14:	learn: 0.5457586	total: 1m 34s	remaining: 3m 41s
15:	learn: 0.5376851	total: 1m 41s	remaining: 3m 35s
16:	learn: 0.5300302	total: 1m 47s	remaining: 3m 29s
17:	learn: 0.5215239	total: 1m 54s	remaining: 3m 22s
18:	learn: 0.5135123	total: 2m	remaining: 3m 16s
19:	learn: 0.5072053	total: 2m 7s	remaining: 3m 10s
20:	learn: 0.5006481	total: 2m 13s	remaining: 3m 4s
21:	learn: 0.4944190	total: 2m 20s	remaining: 2m 58s
22:	learn: 0.4869426	total: 2m 26s	remaining: 2m 51s
23:	learn: 0.4807349	total: 2m 32s	remaining: 2m 45s
24:	learn: 0.4749246	total: 2m 39s	remaining: 2m 39s
25:	learn: 0.4680998	total: 2m 45s	remaining: 2m 32s
26:	learn: 0.4616890	total: 2m 51s	remaining: 2m 26s
27:	learn: 0.4560619	total: 2m 57s	remaining: 2m 19s
28:	learn: 0.4491938	total: 3m 4s	remaining: 2m 13s
29:	learn: 0.4439734	total: 3m 10s	remaining: 2m 6s
30:	learn: 0.4392526	total: 3m 16s	remaining: 2m
31:	learn: 0.4339170	total: 3m 22s	remaining: 1m 54s
32:	learn: 0.4283654	total: 3m 29s	remaining: 1m 47s
33:	learn: 0.4230484	total: 3m 35s	remaining: 1m 41s
34:	learn: 0.4185932	total: 3m 41s	remaining: 1m 35s
35:	learn: 0.4119146	total: 3m 48s	remaining: 1m 28s
36:	learn: 0.4061845	total: 3m 54s	remaining: 1m 22s
37:	learn: 0.4018641	total: 4m	remaining: 1m 15s
38:	learn: 0.3974780	total: 4m 6s	remaining: 1m 9s
39:	learn: 0.3928356	total: 4m 13s	remaining: 1m 3s
40:	learn: 0.3890938	total: 4m 19s	remaining: 57s
41:	learn: 0.3832318	total: 4m 25s	remaining: 50.6s
42:	learn: 0.3781490	total: 4m 32s	remaining: 44.3s
43:	learn: 0.3746409	total: 4m 38s	remaining: 38s
44:	learn: 0.3704320	total: 4m 45s	remaining: 31.7s
45:	learn: 0.3666797	total: 4m 51s	remaining: 25.3s
46:	learn: 0.3620596	total: 4m 57s	remaining: 19s
47:	learn: 0.3584769	total: 5m 3s	remaining: 12.7s

48:	learn: 0.3550112	total: 5m 10s	remaining: 6.33s
49:	learn: 0.3514551	total: 5m 16s	remaining: 0us
0:	learn: 0.6823562	total: 6.15s	remaining: 5m 1s
1:	learn: 0.6713425	total: 12.5s	remaining: 5m
2:	learn: 0.6602592	total: 18.9s	remaining: 4m 55s
3:	learn: 0.6482029	total: 25.4s	remaining: 4m 52s
4:	learn: 0.6369937	total: 31.6s	remaining: 4m 44s
5:	learn: 0.6265531	total: 37.8s	remaining: 4m 37s
6:	learn: 0.6174270	total: 44.3s	remaining: 4m 31s
7:	learn: 0.6072963	total: 50.7s	remaining: 4m 26s
8:	learn: 0.5964610	total: 57.2s	remaining: 4m 20s
9:	learn: 0.5875313	total: 1m 3s	remaining: 4m 13s
10:	learn: 0.5792694	total: 1m 9s	remaining: 4m 7s
11:	learn: 0.5701128	total: 1m 16s	remaining: 4m 1s
12:	learn: 0.5620972	total: 1m 22s	remaining: 3m 54s
13:	learn: 0.5531804	total: 1m 29s	remaining: 3m 48s
14:	learn: 0.5452829	total: 1m 35s	remaining: 3m 43s
15:	learn: 0.5378235	total: 1m 42s	remaining: 3m 37s
16:	learn: 0.5298731	total: 1m 49s	remaining: 3m 31s
17:	learn: 0.5212755	total: 1m 55s	remaining: 3m 25s
18:	learn: 0.5137338	total: 2m 2s	remaining: 3m 19s
19:	learn: 0.5065493	total: 2m 8s	remaining: 3m 12s
20:	learn: 0.5005573	total: 2m 14s	remaining: 3m 6s
21:	learn: 0.4932797	total: 2m 21s	remaining: 2m 59s
22:	learn: 0.4858127	total: 2m 27s	remaining: 2m 53s
23:	learn: 0.4789768	total: 2m 33s	remaining: 2m 46s
24:	learn: 0.4729366	total: 2m 39s	remaining: 2m 39s
25:	learn: 0.4656922	total: 2m 46s	remaining: 2m 33s
26:	learn: 0.4602725	total: 2m 52s	remaining: 2m 26s
27:	learn: 0.4550913	total: 2m 58s	remaining: 2m 20s
28:	learn: 0.4490080	total: 3m 5s	remaining: 2m 14s
29:	learn: 0.4431934	total: 3m 11s	remaining: 2m 7s
30:	learn: 0.4364060	total: 3m 17s	remaining: 2m 1s
31:	learn: 0.4308669	total: 3m 24s	remaining: 1m 54s
32:	learn: 0.4265062	total: 3m 30s	remaining: 1m 48s
33:	learn: 0.4218306	total: 3m 36s	remaining: 1m 41s
34:	learn: 0.4168006	total: 3m 42s	remaining: 1m 35s
35:	learn: 0.4099461	total: 3m 49s	remaining: 1m 29s
36:	learn: 0.4045570	total: 3m 55s	remaining: 1m 22s
37:	learn: 0.3995564	total: 4m 1s	remaining: 1m 16s
38:	learn: 0.3936354	total: 4m 8s	remaining: 1m 9s
39:	learn: 0.3884234	total: 4m 14s	remaining: 1m 3s
40:	learn: 0.3848444	total: 4m 20s	remaining: 57.1s
41:	learn: 0.3796490	total: 4m 26s	remaining: 50.7s
42:	learn: 0.3746280	total: 4m 32s	remaining: 44.4s
43:	learn: 0.3705624	total: 4m 39s	remaining: 38s
44:	learn: 0.3671124	total: 4m 45s	remaining: 31.7s
45:	learn: 0.3624869	total: 4m 51s	remaining: 25.4s

46:	learn: 0.3588521	total: 4m 57s	remaining: 19s
47:	learn: 0.3547275	total: 5m 4s	remaining: 12.7s
48:	learn: 0.3506664	total: 5m 10s	remaining: 6.34s
49:	learn: 0.3475771	total: 5m 16s	remaining: 0us
0:	learn: 0.6821467	total: 6.29s	remaining: 5m 7s
1:	learn: 0.6710348	total: 12.7s	remaining: 5m 3s
2:	learn: 0.6607421	total: 18.9s	remaining: 4m 56s
3:	learn: 0.6491377	total: 25.2s	remaining: 4m 49s
4:	learn: 0.6390033	total: 31.5s	remaining: 4m 43s
5:	learn: 0.6289471	total: 37.8s	remaining: 4m 36s
6:	learn: 0.6192053	total: 44.1s	remaining: 4m 31s
7:	learn: 0.6085793	total: 50.3s	remaining: 4m 24s
8:	learn: 0.5988457	total: 56.6s	remaining: 4m 17s
9:	learn: 0.5879411	total: 1m 2s	remaining: 4m 11s
10:	learn: 0.5786219	total: 1m 9s	remaining: 4m 4s
11:	learn: 0.5698657	total: 1m 15s	remaining: 3m 59s
12:	learn: 0.5618202	total: 1m 22s	remaining: 3m 53s
13:	learn: 0.5538130	total: 1m 28s	remaining: 3m 47s
14:	learn: 0.5454489	total: 1m 34s	remaining: 3m 40s
15:	learn: 0.5370131	total: 1m 41s	remaining: 3m 34s
16:	learn: 0.5289737	total: 1m 47s	remaining: 3m 28s
17:	learn: 0.5209192	total: 1m 53s	remaining: 3m 22s
18:	learn: 0.5129195	total: 2m	remaining: 3m 16s
19:	learn: 0.5064439	total: 2m 6s	remaining: 3m 10s
20:	learn: 0.4996266	total: 2m 13s	remaining: 3m 4s
21:	learn: 0.4943816	total: 2m 20s	remaining: 2m 58s
22:	learn: 0.4870625	total: 2m 26s	remaining: 2m 52s
23:	learn: 0.4802609	total: 2m 33s	remaining: 2m 45s
24:	learn: 0.4727608	total: 2m 39s	remaining: 2m 39s
25:	learn: 0.4660177	total: 2m 46s	remaining: 2m 33s
26:	learn: 0.4599535	total: 2m 53s	remaining: 2m 27s
27:	learn: 0.4525829	total: 2m 59s	remaining: 2m 21s
28:	learn: 0.4472870	total: 3m 6s	remaining: 2m 15s
29:	learn: 0.4407548	total: 3m 13s	remaining: 2m 9s
30:	learn: 0.4349962	total: 3m 20s	remaining: 2m 2s
31:	learn: 0.4283151	total: 3m 27s	remaining: 1m 56s
32:	learn: 0.4218098	total: 3m 34s	remaining: 1m 50s
33:	learn: 0.4172195	total: 3m 41s	remaining: 1m 44s
34:	learn: 0.4123722	total: 3m 48s	remaining: 1m 37s
35:	learn: 0.4082760	total: 3m 55s	remaining: 1m 31s
36:	learn: 0.4031478	total: 4m 2s	remaining: 1m 25s
37:	learn: 0.3987115	total: 4m 9s	remaining: 1m 18s
38:	learn: 0.3938375	total: 4m 15s	remaining: 1m 12s
39:	learn: 0.3897461	total: 4m 22s	remaining: 1m 5s
40:	learn: 0.3865191	total: 4m 28s	remaining: 59s
41:	learn: 0.3811761	total: 4m 35s	remaining: 52.5s
42:	learn: 0.3776367	total: 4m 41s	remaining: 45.9s
43:	learn: 0.3729128	total: 4m 48s	remaining: 39.3s

44:	learn: 0.3684969	total: 4m 54s	remaining: 32.7s
45:	learn: 0.3638363	total: 5m	remaining: 26.2s
46:	learn: 0.3587760	total: 5m 7s	remaining: 19.6s
47:	learn: 0.3544233	total: 5m 13s	remaining: 13.1s
48:	learn: 0.3498643	total: 5m 20s	remaining: 6.54s
49:	learn: 0.3463332	total: 5m 26s	remaining: 0us
0:	learn: 0.6821537	total: 6.4s	remaining: 5m 13s
1:	learn: 0.6703938	total: 12.8s	remaining: 5m 7s
2:	learn: 0.6604888	total: 19.2s	remaining: 5m 1s
3:	learn: 0.6497046	total: 26s	remaining: 4m 58s
4:	learn: 0.6380820	total: 32.6s	remaining: 4m 53s
5:	learn: 0.6278445	total: 39.4s	remaining: 4m 48s
6:	learn: 0.6178954	total: 46s	remaining: 4m 42s
7:	learn: 0.6071832	total: 52.6s	remaining: 4m 36s
8:	learn: 0.5966474	total: 59.4s	remaining: 4m 30s
9:	learn: 0.5877277	total: 1m 6s	remaining: 4m 24s
10:	learn: 0.5788090	total: 1m 12s	remaining: 4m 17s
11:	learn: 0.5695249	total: 1m 19s	remaining: 4m 10s
12:	learn: 0.5615448	total: 1m 25s	remaining: 4m 3s
13:	learn: 0.5525978	total: 1m 32s	remaining: 3m 57s
14:	learn: 0.5461737	total: 1m 38s	remaining: 3m 50s
15:	learn: 0.5382542	total: 1m 45s	remaining: 3m 44s
16:	learn: 0.5299420	total: 1m 52s	remaining: 3m 38s
17:	learn: 0.5216704	total: 1m 58s	remaining: 3m 31s
18:	learn: 0.5137588	total: 2m 5s	remaining: 3m 24s
19:	learn: 0.5064492	total: 2m 12s	remaining: 3m 18s
20:	learn: 0.4996117	total: 2m 18s	remaining: 3m 11s
21:	learn: 0.4920019	total: 2m 25s	remaining: 3m 4s
22:	learn: 0.4852151	total: 2m 32s	remaining: 2m 58s
23:	learn: 0.4781119	total: 2m 38s	remaining: 2m 52s
24:	learn: 0.4716736	total: 2m 45s	remaining: 2m 45s
25:	learn: 0.4657778	total: 2m 52s	remaining: 2m 38s
26:	learn: 0.4604170	total: 2m 58s	remaining: 2m 31s
27:	learn: 0.4543728	total: 3m 4s	remaining: 2m 25s
28:	learn: 0.4489284	total: 3m 11s	remaining: 2m 18s
29:	learn: 0.4426352	total: 3m 17s	remaining: 2m 11s
30:	learn: 0.4358402	total: 3m 23s	remaining: 2m 4s
31:	learn: 0.4299508	total: 3m 30s	remaining: 1m 58s
32:	learn: 0.4249207	total: 3m 36s	remaining: 1m 51s
33:	learn: 0.4200868	total: 3m 43s	remaining: 1m 44s
34:	learn: 0.4150649	total: 3m 49s	remaining: 1m 38s
35:	learn: 0.4081774	total: 3m 55s	remaining: 1m 31s
36:	learn: 0.4037173	total: 4m 2s	remaining: 1m 25s
37:	learn: 0.3995148	total: 4m 8s	remaining: 1m 18s
38:	learn: 0.3944312	total: 4m 15s	remaining: 1m 12s
39:	learn: 0.3894426	total: 4m 22s	remaining: 1m 5s
40:	learn: 0.3852817	total: 4m 28s	remaining: 59s
41:	learn: 0.3796710	total: 4m 35s	remaining: 52.5s

42:	learn: 0.3751131	total: 4m 41s	remaining: 45.9s
43:	learn: 0.3714101	total: 4m 48s	remaining: 39.3s
44:	learn: 0.3678924	total: 4m 55s	remaining: 32.8s
45:	learn: 0.3646204	total: 5m 1s	remaining: 26.2s
46:	learn: 0.3601537	total: 5m 8s	remaining: 19.7s
47:	learn: 0.3555847	total: 5m 14s	remaining: 13.1s
48:	learn: 0.3513182	total: 5m 21s	remaining: 6.56s
49:	learn: 0.3466763	total: 5m 28s	remaining: 0us
0:	learn: 0.6816172	total: 6.86s	remaining: 5m 35s
1:	learn: 0.6714302	total: 13.4s	remaining: 5m 20s
2:	learn: 0.6605070	total: 20s	remaining: 5m 12s
3:	learn: 0.6492099	total: 26.5s	remaining: 5m 5s
4:	learn: 0.6378036	total: 33.1s	remaining: 4m 58s
5:	learn: 0.6276238	total: 39.7s	remaining: 4m 50s
6:	learn: 0.6177993	total: 46.3s	remaining: 4m 44s
7:	learn: 0.6069426	total: 52.9s	remaining: 4m 37s
8:	learn: 0.5964072	total: 59.3s	remaining: 4m 30s
9:	learn: 0.5872166	total: 1m 5s	remaining: 4m 23s
10:	learn: 0.5788251	total: 1m 12s	remaining: 4m 17s
11:	learn: 0.5697872	total: 1m 19s	remaining: 4m 11s
12:	learn: 0.5625097	total: 1m 25s	remaining: 4m 3s
13:	learn: 0.5533912	total: 1m 32s	remaining: 3m 56s
14:	learn: 0.5442543	total: 1m 38s	remaining: 3m 49s
15:	learn: 0.5365446	total: 1m 44s	remaining: 3m 42s
16:	learn: 0.5291370	total: 1m 51s	remaining: 3m 36s
17:	learn: 0.5205394	total: 1m 57s	remaining: 3m 29s
18:	learn: 0.5122904	total: 2m 4s	remaining: 3m 22s
19:	learn: 0.5047293	total: 2m 10s	remaining: 3m 16s
20:	learn: 0.4974306	total: 2m 17s	remaining: 3m 9s
21:	learn: 0.4898320	total: 2m 23s	remaining: 3m 2s
22:	learn: 0.4826910	total: 2m 29s	remaining: 2m 55s
23:	learn: 0.4756345	total: 2m 36s	remaining: 2m 49s
24:	learn: 0.4693659	total: 2m 42s	remaining: 2m 42s
25:	learn: 0.4620870	total: 2m 49s	remaining: 2m 36s
26:	learn: 0.4559901	total: 2m 55s	remaining: 2m 29s
27:	learn: 0.4504827	total: 3m 2s	remaining: 2m 23s
28:	learn: 0.4439787	total: 3m 8s	remaining: 2m 16s
29:	learn: 0.4372607	total: 3m 15s	remaining: 2m 10s
30:	learn: 0.4309354	total: 3m 21s	remaining: 2m 3s
31:	learn: 0.4249162	total: 3m 28s	remaining: 1m 57s
32:	learn: 0.4209523	total: 3m 34s	remaining: 1m 50s
33:	learn: 0.4156442	total: 3m 40s	remaining: 1m 43s
34:	learn: 0.4111826	total: 3m 47s	remaining: 1m 37s
35:	learn: 0.4055449	total: 3m 53s	remaining: 1m 30s
36:	learn: 0.4006449	total: 4m	remaining: 1m 24s
37:	learn: 0.3957891	total: 4m 6s	remaining: 1m 17s
38:	learn: 0.3914760	total: 4m 13s	remaining: 1m 11s
39:	learn: 0.3871325	total: 4m 19s	remaining: 1m 4s

```

40: learn: 0.3819607      total: 4m 25s    remaining: 58.3s
41: learn: 0.3775978      total: 4m 32s    remaining: 51.8s
42: learn: 0.3728153      total: 4m 38s    remaining: 45.3s
43: learn: 0.3692168      total: 4m 44s    remaining: 38.8s
44: learn: 0.3654944      total: 4m 51s    remaining: 32.3s
45: learn: 0.3622876      total: 4m 57s    remaining: 25.9s
46: learn: 0.3586039      total: 5m 3s     remaining: 19.4s
47: learn: 0.3539011      total: 5m 10s    remaining: 12.9s
48: learn: 0.3502644      total: 5m 16s    remaining: 6.46s
49: learn: 0.3467322      total: 5m 22s    remaining: 0us
<catboost.core.CatBoostClassifier object at 0x7d1968e0aa10>:
CROSS VAL Scores: [0.98935275 0.98999653 0.98979845 0.99227455 0.99172981
0.99034319
0.99034319 0.98910514 0.99187797 0.99167987]
Score'(mean,std)': (0.9906501439005829, 0.0010855232338973294)
0: learn: 0.6831659      total: 6.49s    remaining: 5m 18s
1: learn: 0.6715645      total: 13.1s    remaining: 5m 14s
2: learn: 0.6603169      total: 19.5s    remaining: 5m 4s
3: learn: 0.6490538      total: 25.8s    remaining: 4m 57s
4: learn: 0.6381048      total: 32.5s    remaining: 4m 52s
5: learn: 0.6289159      total: 38.8s    remaining: 4m 44s
6: learn: 0.6192138      total: 45.2s    remaining: 4m 37s
7: learn: 0.6099877      total: 51.7s    remaining: 4m 31s
8: learn: 0.6002195      total: 58.3s    remaining: 4m 25s
9: learn: 0.5900492      total: 1m 4s     remaining: 4m 19s
10: learn: 0.5796813     total: 1m 11s   remaining: 4m 13s
11: learn: 0.5692456     total: 1m 17s   remaining: 4m 6s
12: learn: 0.5610764     total: 1m 24s   remaining: 4m
13: learn: 0.5526541     total: 1m 30s   remaining: 3m 53s
14: learn: 0.5456215     total: 1m 37s   remaining: 3m 47s
15: learn: 0.5377285     total: 1m 43s   remaining: 3m 40s
16: learn: 0.5291146     total: 1m 50s   remaining: 3m 33s
17: learn: 0.5213319     total: 1m 56s   remaining: 3m 27s
18: learn: 0.5126184     total: 2m 3s    remaining: 3m 21s
19: learn: 0.5062879     total: 2m 9s    remaining: 3m 14s
20: learn: 0.4996302     total: 2m 16s   remaining: 3m 8s
21: learn: 0.4919375     total: 2m 22s   remaining: 3m 1s
22: learn: 0.4857917     total: 2m 29s   remaining: 2m 55s
23: learn: 0.4782165     total: 2m 35s   remaining: 2m 48s
24: learn: 0.4708381     total: 2m 42s   remaining: 2m 42s
25: learn: 0.4639990     total: 2m 48s   remaining: 2m 35s
26: learn: 0.4575539     total: 2m 55s   remaining: 2m 29s
27: learn: 0.4511849     total: 3m 1s    remaining: 2m 22s
28: learn: 0.4455764     total: 3m 7s    remaining: 2m 16s
29: learn: 0.4401382     total: 3m 14s   remaining: 2m 9s
30: learn: 0.4332185     total: 3m 20s   remaining: 2m 2s
31: learn: 0.4278849     total: 3m 26s   remaining: 1m 56s
32: learn: 0.4238827     total: 3m 33s   remaining: 1m 49s

```

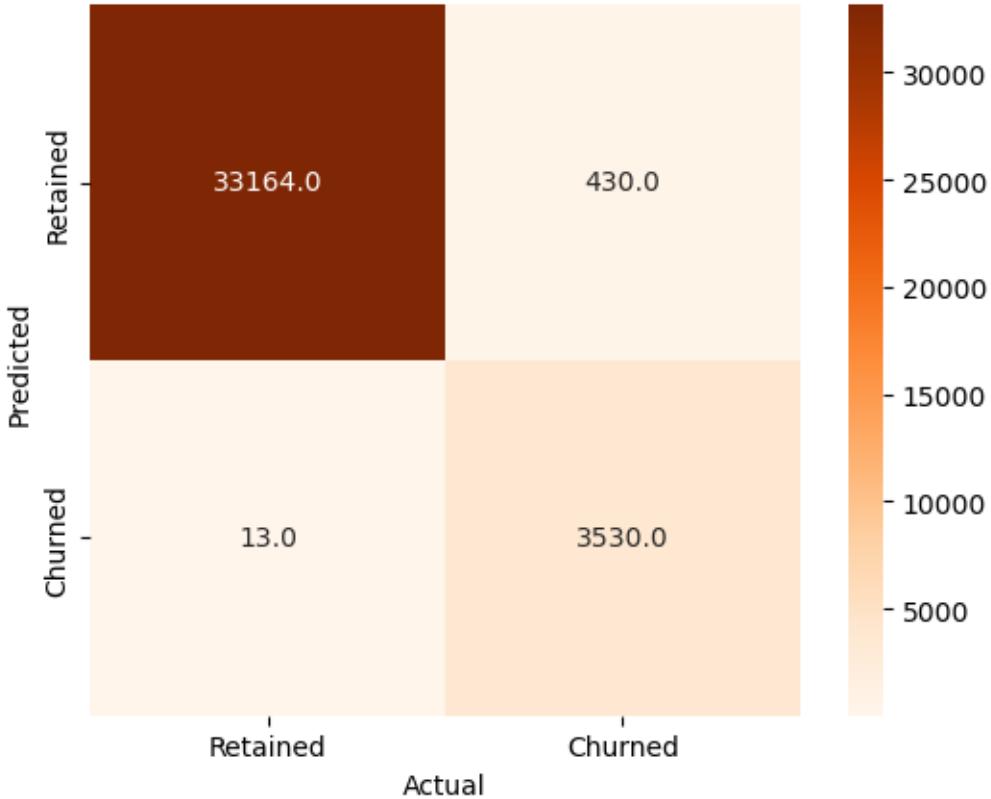
33:	learn: 0.4178975	total: 3m 40s	remaining: 1m 43s
34:	learn: 0.4133426	total: 3m 46s	remaining: 1m 37s
35:	learn: 0.4087287	total: 3m 53s	remaining: 1m 30s
36:	learn: 0.4045262	total: 3m 59s	remaining: 1m 24s
37:	learn: 0.3983589	total: 4m 6s	remaining: 1m 17s
38:	learn: 0.3923288	total: 4m 12s	remaining: 1m 11s
39:	learn: 0.3873423	total: 4m 19s	remaining: 1m 4s
40:	learn: 0.3827544	total: 4m 25s	remaining: 58.3s
41:	learn: 0.3781394	total: 4m 32s	remaining: 51.8s
42:	learn: 0.3740561	total: 4m 38s	remaining: 45.4s
43:	learn: 0.3694011	total: 4m 45s	remaining: 38.9s
44:	learn: 0.3641265	total: 4m 51s	remaining: 32.4s
45:	learn: 0.3598584	total: 4m 58s	remaining: 25.9s
46:	learn: 0.3553665	total: 5m 4s	remaining: 19.5s
47:	learn: 0.3517139	total: 5m 11s	remaining: 13s
48:	learn: 0.3470001	total: 5m 18s	remaining: 6.49s
49:	learn: 0.3431514	total: 5m 24s	remaining: 0us

CLASSIFICATION REPORT : <catboost.core.CatBoostClassifier object at 0x7d1968e0aa10>

	precision	recall	f1-score	support
0	1.00	0.99	0.99	33594
1	0.89	1.00	0.94	3543
accuracy			0.99	37137
macro avg	0.95	0.99	0.97	37137
weighted avg	0.99	0.99	0.99	37137

CONFUSION MARTIX : <catboost.core.CatBoostClassifier object at 0x7d1968e0aa10>

True Positive: 33164  
 False Positive: 430  
 False Negative: 13  
 True Negative: 3530



```
[99]: pickle.dump(cat,open('Categoricalboosting clf.pkl','wb'))
```

## ADA BOOST CLASSIFIER

```
[91]: dt=DecisionTreeClassifier(max_depth=15,min_samples_split=20,)

ada=AdaBoostClassifier(estimator=dt,n_estimators=62, learning_rate=0.01,
                      algorithm='SAMME.R', base_estimator='deprecated')

kfolds(ada)
```

```
AdaBoostClassifier(estimator=DecisionTreeClassifier(max_depth=15,
                                                    min_samples_split=20),
                   learning_rate=0.01, n_estimators=62):
CROSS VAL Scores: [1.          0.99995048 1.          0.99995048 0.99980191 1.
0.99980191 0.99995048 1.          0.99995048]
Score'(mean,std)': (0.9999405732207964, 7.278230930526106e-05)
```

### CLASSIFICATION REPORT :

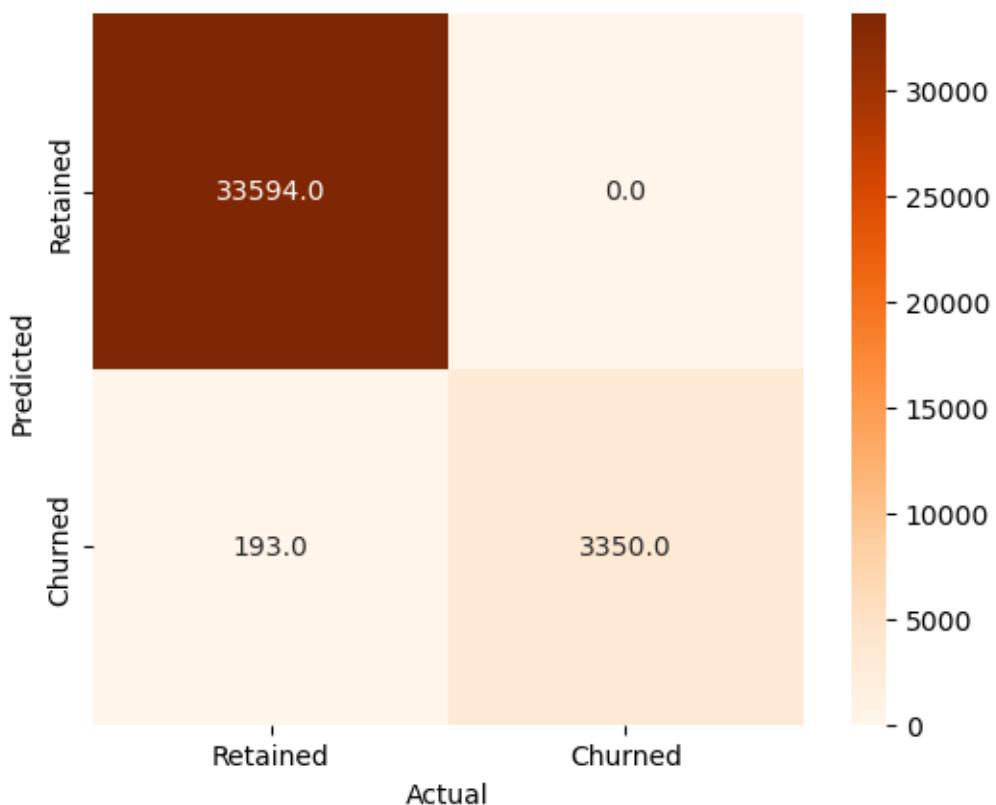
```
AdaBoostClassifier(estimator=DecisionTreeClassifier(max_depth=15,
                                                    min_samples_split=20),
                   learning_rate=0.01, n_estimators=62)
precision    recall   f1-score    support
```

0	0.99	1.00	1.00	33594
1	1.00	0.95	0.97	3543
accuracy			0.99	37137
macro avg	1.00	0.97	0.98	37137
weighted avg	0.99	0.99	0.99	37137

CONFUSION MATRIX :

```
AdaBoostClassifier(estimator=DecisionTreeClassifier(max_depth=15,
                                                    min_samples_split=20),
                   learning_rate=0.01, n_estimators=62)

True Positive: 33594
False Positive: 0
False Negative: 193
True Negative: 3350
```



```
[93]: import pickle
pickle.dump(ada,open('Adaboost clf.pkl','wb'))
```

## LOGISTIC REGRESSION

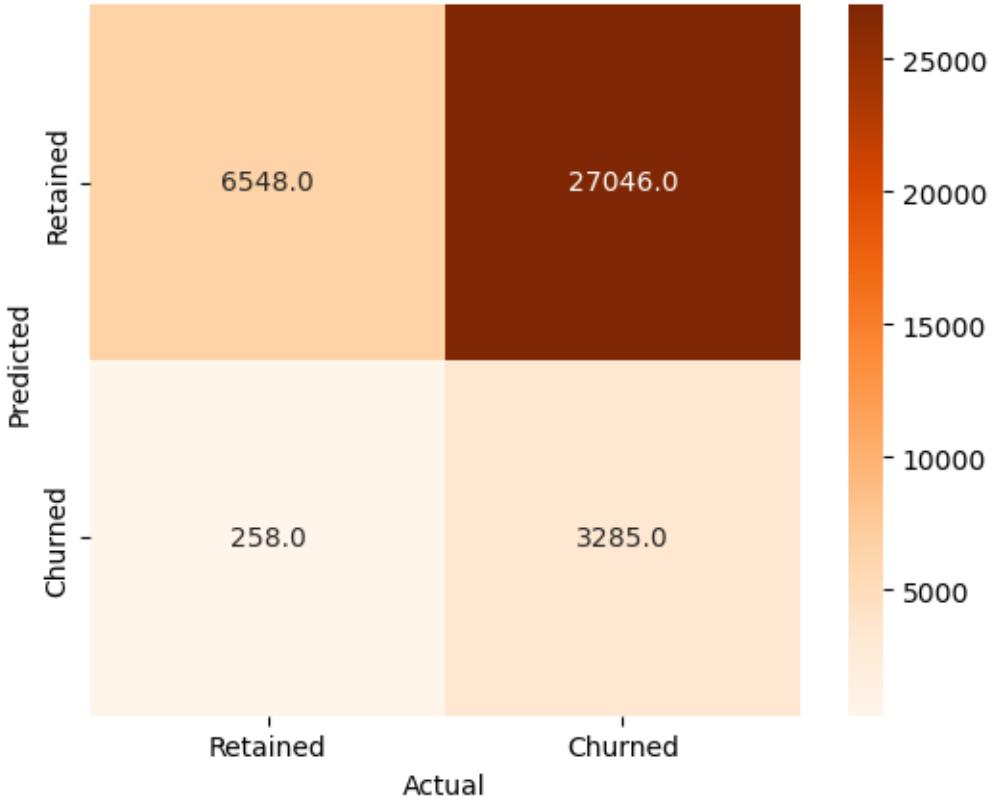
```
[142]: from sklearn.linear_model import LogisticRegression

lr1=LogisticRegression(penalty='l2',dual=False,tol=0.001,C=.
                     ↪01,fit_intercept=True,
                     intercept_scaling=1,solver='sag',□
                     ↪max_iter=1000,class_weight={1:4,0:2})
kfolds(lr1)

LogisticRegression(C=0.01, class_weight={0: 2, 1: 4}, max_iter=1000,
                   solver='sag', tol=0.001):
CROSS VAL Scores: [0.55573714 0.55613331 0.55712376 0.55950082 0.55964938
0.55583618
0.55915416 0.56004556 0.55462559 0.55720087]
Score'(mean,std)': (0.5575006775935518, 0.001845976793701835)

CLASSIFICATION REPORT : LogisticRegression(C=0.01, class_weight={0: 2, 1: 4},
max_iter=1000,
                     solver='sag', tol=0.001)
      precision    recall   f1-score   support
          0         0.96     0.19     0.32     33594
          1         0.11     0.93     0.19     3543
accuracy                          0.26     37137
macro avg                      0.54     0.56     0.26     37137
weighted avg                     0.88     0.26     0.31     37137

CONFUSION MARTIX : LogisticRegression(C=0.01, class_weight={0: 2, 1: 4},
max_iter=1000,
                     solver='sag', tol=0.001)
True Positive: 6548
False Positive: 27046
False Negative: 258
True Negative: 3285
```



## NEURAL NETWORK-LOGISTIC REGRESSION MODEL

```
[73]: import tensorflow as tf
from tensorflow import keras
from keras.layers import Dense
from tensorflow.keras.losses import BinaryCrossentropy
from tensorflow.keras.optimizers import SGD
#from tensorflow.optimizers import Lion
import tensorflow_addons as tfa
```

```
[74]: NeuralNetwork=tf.keras.Sequential([Dense(units=77,activation='relu',name='l1',input_shape=(xtrain.shape[1],)),
                                         Dense(units=154,activation='relu',name='l2'),
                                         Dense(units=308,activation='relu',name='l3'),#308
                                         Dense(units=1,activation='sigmoid',name='final')])
```

```
[75]: binary=tf.keras.losses.BinaryCrossentropy(from_logits=False,)
sgd=tf.keras.optimizers.SGD(learning_rate=0.01,)
```

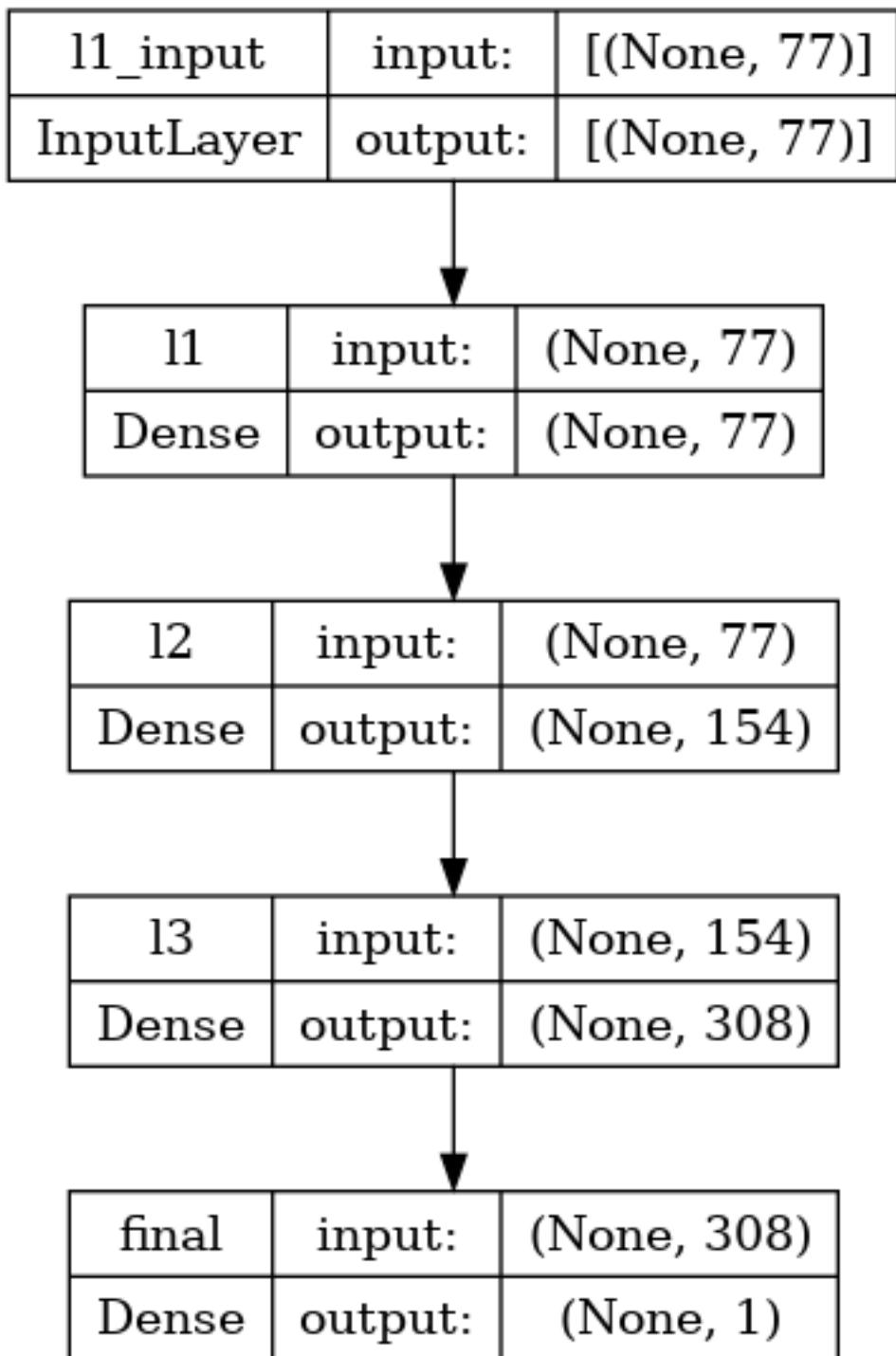
```
NeuralNetwork.compile(optimizer='sgd', loss='binary',  
                      metrics=['Precision', 'Recall', 'Accuracy'], steps_per_execution=100,)
```

[76]: NeuralNetwork.summary()

```
Model: "sequential"  
-----  
Layer (type)          Output Shape       Param #  
=====-----  
11 (Dense)           (None, 77)         6006  
12 (Dense)           (None, 154)        12012  
13 (Dense)           (None, 308)        47740  
final (Dense)         (None, 1)          309  
=====-----  
Total params: 66,067  
Trainable params: 66,067  
Non-trainable params: 0
```

[77]: from tensorflow.keras.utils import plot\_model  
plot\_model(NeuralNetwork, show\_shapes=True, to\_file='NeuralNetwork-logistic model.  
 png')

[77]:



```
best history=NeuralNetwork.fit(xtrain,ytrain,shuffle=True,epochs=60,class_weight={1:15,0:2},validation_split=0.2)  
his=history.history NeuralNetwork.save('/kaggle/working/NeuralNetwok-logistic reg.h5')
```

```
[78]: history=NeuralNetwork.  
      ↪fit(xtrain,ytrain,shuffle=True,epochs=50,validation_split=.2,class_weight={1:  
      ↪4,0:2})#,class_weight={1:4,0:2},e=60  
his=history.history  
NeuralNetwork.save('/kaggle/working/NeuralNetwok-logistic reg.h5')
```

Epoch 1/50  
5049/5049 [=====] - 10s 2ms/step - loss: 1.7928 -  
precision: 0.4535 - recall: 0.7384 - Accuracy: 0.5682 - val\_loss: 0.7749 -  
val\_precision: 1.0000 - val\_recall: 0.4562 - val\_Accuracy: 0.4562  
Epoch 2/50  
5049/5049 [=====] - 8s 2ms/step - loss: 1.6987 -  
precision: 0.4955 - recall: 0.7241 - Accuracy: 0.6201 - val\_loss: 0.7361 -  
val\_precision: 1.0000 - val\_recall: 0.5369 - val\_Accuracy: 0.5369  
Epoch 3/50  
5049/5049 [=====] - 7s 1ms/step - loss: 1.5927 -  
precision: 0.5304 - recall: 0.7480 - Accuracy: 0.6572 - val\_loss: 0.4153 -  
val\_precision: 1.0000 - val\_recall: 0.8542 - val\_Accuracy: 0.8542  
Epoch 4/50  
5049/5049 [=====] - 7s 1ms/step - loss: 1.4693 -  
precision: 0.5723 - recall: 0.7736 - Accuracy: 0.6983 - val\_loss: 0.3103 -  
val\_precision: 1.0000 - val\_recall: 0.8824 - val\_Accuracy: 0.8824  
Epoch 5/50  
5049/5049 [=====] - 8s 2ms/step - loss: 1.3367 -  
precision: 0.6134 - recall: 0.8048 - Accuracy: 0.7366 - val\_loss: 0.1660 -  
val\_precision: 1.0000 - val\_recall: 0.9650 - val\_Accuracy: 0.9650  
Epoch 6/50  
5049/5049 [=====] - 8s 2ms/step - loss: 1.1992 -  
precision: 0.6505 - recall: 0.8370 - Accuracy: 0.7702 - val\_loss: 0.4795 -  
val\_precision: 1.0000 - val\_recall: 0.7510 - val\_Accuracy: 0.7510  
Epoch 7/50  
5049/5049 [=====] - 7s 1ms/step - loss: 1.0648 -  
precision: 0.6870 - recall: 0.8643 - Accuracy: 0.8015 - val\_loss: 0.8093 -  
val\_precision: 1.0000 - val\_recall: 0.5677 - val\_Accuracy: 0.5677  
Epoch 8/50  
5049/5049 [=====] - 7s 1ms/step - loss: 0.9399 -  
precision: 0.7209 - recall: 0.8867 - Accuracy: 0.8288 - val\_loss: 0.1191 -  
val\_precision: 1.0000 - val\_recall: 0.9713 - val\_Accuracy: 0.9713  
Epoch 9/50  
5049/5049 [=====] - 8s 1ms/step - loss: 0.8259 -  
precision: 0.7528 - recall: 0.9061 - Accuracy: 0.8532 - val\_loss: 0.2178 -  
val\_precision: 1.0000 - val\_recall: 0.9204 - val\_Accuracy: 0.9204  
Epoch 10/50  
5049/5049 [=====] - 7s 1ms/step - loss: 0.7273 -  
precision: 0.7813 - recall: 0.9211 - Accuracy: 0.8737 - val\_loss: 0.0771 -  
val\_precision: 1.0000 - val\_recall: 0.9911 - val\_Accuracy: 0.9911  
Epoch 11/50  
5049/5049 [=====] - 8s 2ms/step - loss: 0.6402 -

```
precision: 0.8073 - recall: 0.9328 - Accuracy: 0.8913 - val_loss: 0.2470 -
val_precision: 1.0000 - val_recall: 0.9067 - val_Accuracy: 0.9067
Epoch 12/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.5610 -
precision: 0.8299 - recall: 0.9432 - Accuracy: 0.9062 - val_loss: 0.5941 -
val_precision: 1.0000 - val_recall: 0.7335 - val_Accuracy: 0.7335
Epoch 13/50
5049/5049 [=====] - 7s 1ms/step - loss: 0.5048 -
precision: 0.8457 - recall: 0.9499 - Accuracy: 0.9162 - val_loss: 0.8869 -
val_precision: 1.0000 - val_recall: 0.6941 - val_Accuracy: 0.6941
Epoch 14/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.4424 -
precision: 0.8650 - recall: 0.9585 - Accuracy: 0.9283 - val_loss: 0.1116 -
val_precision: 1.0000 - val_recall: 0.9626 - val_Accuracy: 0.9626
Epoch 15/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.3881 -
precision: 0.8813 - recall: 0.9649 - Accuracy: 0.9381 - val_loss: 0.1047 -
val_precision: 1.0000 - val_recall: 0.9700 - val_Accuracy: 0.9700
Epoch 16/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.3579 -
precision: 0.8918 - recall: 0.9670 - Accuracy: 0.9436 - val_loss: 0.0331 -
val_precision: 1.0000 - val_recall: 0.9973 - val_Accuracy: 0.9973
Epoch 17/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.3176 -
precision: 0.9039 - recall: 0.9730 - Accuracy: 0.9511 - val_loss: 0.0471 -
val_precision: 1.0000 - val_recall: 0.9906 - val_Accuracy: 0.9906
Epoch 18/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.2857 -
precision: 0.9134 - recall: 0.9758 - Accuracy: 0.9562 - val_loss: 0.0681 -
val_precision: 1.0000 - val_recall: 0.9815 - val_Accuracy: 0.9815
Epoch 19/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.2489 -
precision: 0.9246 - recall: 0.9801 - Accuracy: 0.9625 - val_loss: 0.0676 -
val_precision: 1.0000 - val_recall: 0.9850 - val_Accuracy: 0.9850
Epoch 20/50
5049/5049 [=====] - 7s 1ms/step - loss: 0.2519 -
precision: 0.9255 - recall: 0.9784 - Accuracy: 0.9624 - val_loss: 0.2343 -
val_precision: 1.0000 - val_recall: 0.9204 - val_Accuracy: 0.9204
Epoch 21/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.2234 -
precision: 0.9332 - recall: 0.9819 - Accuracy: 0.9668 - val_loss: 0.0344 -
val_precision: 1.0000 - val_recall: 0.9927 - val_Accuracy: 0.9927
Epoch 22/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1988 -
precision: 0.9407 - recall: 0.9839 - Accuracy: 0.9707 - val_loss: 0.0656 -
val_precision: 1.0000 - val_recall: 0.9750 - val_Accuracy: 0.9750
Epoch 23/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1782 -
```

```
precision: 0.9467 - recall: 0.9856 - Accuracy: 0.9738 - val_loss: 0.0525 -
val_precision: 1.0000 - val_recall: 0.9831 - val_Accuracy: 0.9831
Epoch 24/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1865 -
precision: 0.9466 - recall: 0.9843 - Accuracy: 0.9733 - val_loss: 0.1370 -
val_precision: 1.0000 - val_recall: 0.9464 - val_Accuracy: 0.9464
Epoch 25/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1792 -
precision: 0.9490 - recall: 0.9853 - Accuracy: 0.9746 - val_loss: 0.0960 -
val_precision: 1.0000 - val_recall: 0.9650 - val_Accuracy: 0.9650
Epoch 26/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1530 -
precision: 0.9551 - recall: 0.9875 - Accuracy: 0.9779 - val_loss: 0.0491 -
val_precision: 1.0000 - val_recall: 0.9877 - val_Accuracy: 0.9877
Epoch 27/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1295 -
precision: 0.9619 - recall: 0.9902 - Accuracy: 0.9816 - val_loss: 0.0234 -
val_precision: 1.0000 - val_recall: 0.9951 - val_Accuracy: 0.9951
Epoch 28/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1159 -
precision: 0.9659 - recall: 0.9914 - Accuracy: 0.9837 - val_loss: 0.0298 -
val_precision: 1.0000 - val_recall: 0.9904 - val_Accuracy: 0.9904
Epoch 29/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1485 -
precision: 0.9611 - recall: 0.9888 - Accuracy: 0.9808 - val_loss: 0.0291 -
val_precision: 1.0000 - val_recall: 0.9922 - val_Accuracy: 0.9922
Epoch 30/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1394 -
precision: 0.9621 - recall: 0.9887 - Accuracy: 0.9812 - val_loss: 0.0260 -
val_precision: 1.0000 - val_recall: 0.9906 - val_Accuracy: 0.9906
Epoch 31/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1273 -
precision: 0.9656 - recall: 0.9905 - Accuracy: 0.9832 - val_loss: 0.0428 -
val_precision: 1.0000 - val_recall: 0.9904 - val_Accuracy: 0.9904
Epoch 32/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1004 -
precision: 0.9726 - recall: 0.9930 - Accuracy: 0.9869 - val_loss: 0.0269 -
val_precision: 1.0000 - val_recall: 0.9964 - val_Accuracy: 0.9964
Epoch 33/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.0798 -
precision: 0.9782 - recall: 0.9947 - Accuracy: 0.9897 - val_loss: 0.0210 -
val_precision: 1.0000 - val_recall: 0.9960 - val_Accuracy: 0.9960
Epoch 34/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1218 -
precision: 0.9698 - recall: 0.9907 - Accuracy: 0.9849 - val_loss: 11.1686 -
val_precision: 1.0000 - val_recall: 0.1784 - val_Accuracy: 0.1784
Epoch 35/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1347 -
```

```
precision: 0.9688 - recall: 0.9905 - Accuracy: 0.9845 - val_loss: 1.1836 -
val_precision: 1.0000 - val_recall: 0.7289 - val_Accuracy: 0.7289
Epoch 36/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.0724 -
precision: 0.9804 - recall: 0.9952 - Accuracy: 0.9907 - val_loss: 0.4012 -
val_precision: 1.0000 - val_recall: 0.8941 - val_Accuracy: 0.8941
Epoch 37/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.0863 -
precision: 0.9767 - recall: 0.9934 - Accuracy: 0.9887 - val_loss: 0.0260 -
val_precision: 1.0000 - val_recall: 0.9943 - val_Accuracy: 0.9943
Epoch 38/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1014 -
precision: 0.9738 - recall: 0.9924 - Accuracy: 0.9871 - val_loss: 0.1521 -
val_precision: 1.0000 - val_recall: 0.9486 - val_Accuracy: 0.9486
Epoch 39/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.0546 -
precision: 0.9847 - recall: 0.9965 - Accuracy: 0.9929 - val_loss: 0.0242 -
val_precision: 1.0000 - val_recall: 0.9911 - val_Accuracy: 0.9911
Epoch 40/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.0605 -
precision: 0.9840 - recall: 0.9959 - Accuracy: 0.9924 - val_loss: 0.1017 -
val_precision: 1.0000 - val_recall: 0.9666 - val_Accuracy: 0.9666
Epoch 41/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1182 -
precision: 0.9716 - recall: 0.9905 - Accuracy: 0.9856 - val_loss: 0.0126 -
val_precision: 1.0000 - val_recall: 0.9957 - val_Accuracy: 0.9957
Epoch 42/50
5049/5049 [=====] - 7s 1ms/step - loss: 0.0592 -
precision: 0.9839 - recall: 0.9955 - Accuracy: 0.9922 - val_loss: 0.0094 -
val_precision: 1.0000 - val_recall: 0.9985 - val_Accuracy: 0.9985
Epoch 43/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.0588 -
precision: 0.9840 - recall: 0.9959 - Accuracy: 0.9924 - val_loss: 0.0815 -
val_precision: 1.0000 - val_recall: 0.9722 - val_Accuracy: 0.9722
Epoch 44/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.0524 -
precision: 0.9861 - recall: 0.9965 - Accuracy: 0.9934 - val_loss: 0.0077 -
val_precision: 1.0000 - val_recall: 0.9984 - val_Accuracy: 0.9984
Epoch 45/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.1216 -
precision: 0.9733 - recall: 0.9908 - Accuracy: 0.9863 - val_loss: 0.0148 -
val_precision: 1.0000 - val_recall: 0.9973 - val_Accuracy: 0.9973
Epoch 46/50
5049/5049 [=====] - 7s 1ms/step - loss: 0.0395 -
precision: 0.9894 - recall: 0.9976 - Accuracy: 0.9951 - val_loss: 0.0279 -
val_precision: 1.0000 - val_recall: 0.9924 - val_Accuracy: 0.9924
Epoch 47/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.0407 -
```

```

precision: 0.9893 - recall: 0.9973 - Accuracy: 0.9950 - val_loss: 0.0800 -
val_precision: 1.0000 - val_recall: 0.9747 - val_Accuracy: 0.9747
Epoch 48/50
5049/5049 [=====] - 8s 2ms/step - loss: 0.0658 -
precision: 0.9832 - recall: 0.9949 - Accuracy: 0.9917 - val_loss: 0.0098 -
val_precision: 1.0000 - val_recall: 0.9995 - val_Accuracy: 0.9995
Epoch 49/50
5049/5049 [=====] - 8s 1ms/step - loss: 0.0623 -
precision: 0.9839 - recall: 0.9955 - Accuracy: 0.9922 - val_loss: 0.0201 -
val_precision: 1.0000 - val_recall: 0.9970 - val_Accuracy: 0.9970
Epoch 50/50
5049/5049 [=====] - 7s 1ms/step - loss: 0.0412 -
precision: 0.9889 - recall: 0.9970 - Accuracy: 0.9947 - val_loss: 0.0041 -
val_precision: 1.0000 - val_recall: 1.0000 - val_Accuracy: 1.0000

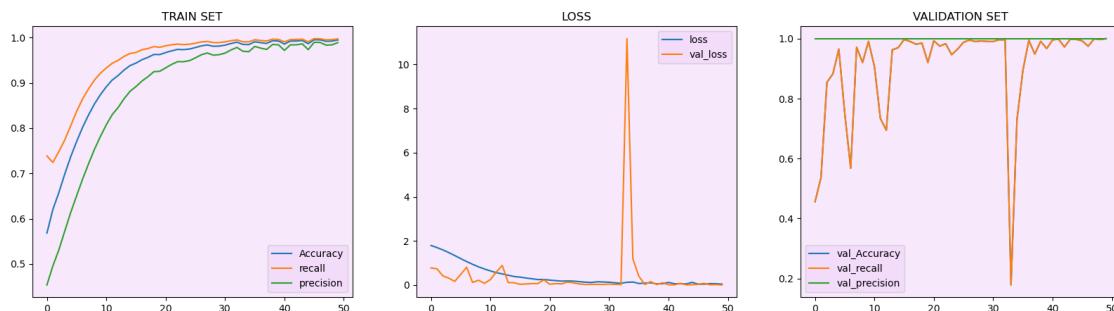
```

```
[84]: fig, axs=plt.subplots(1,3,figsize=(20,5))
axs[0].plot(his['Accuracy'])
axs[0].plot(his['recall'])
axs[0].plot(his['precision'])
axs[0].legend(['Accuracy','recall','precision'])
axs[0].set_title('TRAIN SET')

axs[1].plot(his['loss'])
axs[1].plot(his['val_loss'])
axs[1].legend(['loss','val_loss'])
axs[1].set_title('LOSS')

axs[2].plot(his['val_Accuracy'])
axs[2].plot(his['val_recall'])
axs[2].plot(his['val_precision'])
axs[2].legend(['val_Accuracy','val_recall','val_precision'])
axs[2].set_title('VALIDATION SET')
```

```
[84]: Text(0.5, 1.0, 'VALIDATION SET')
```



```
[85]: ypred=NeuralNetwork.predict(xtest)

# below method will classify the 0 or 1 target
#by predicted values >= 0.5 @ Class 1
# predicted values <0.5 @ Class 0

ypredb=np.where(ypred>=.5,1,0)
```

1161/1161 [=====] - 0s 406us/step

ytest,ypredb

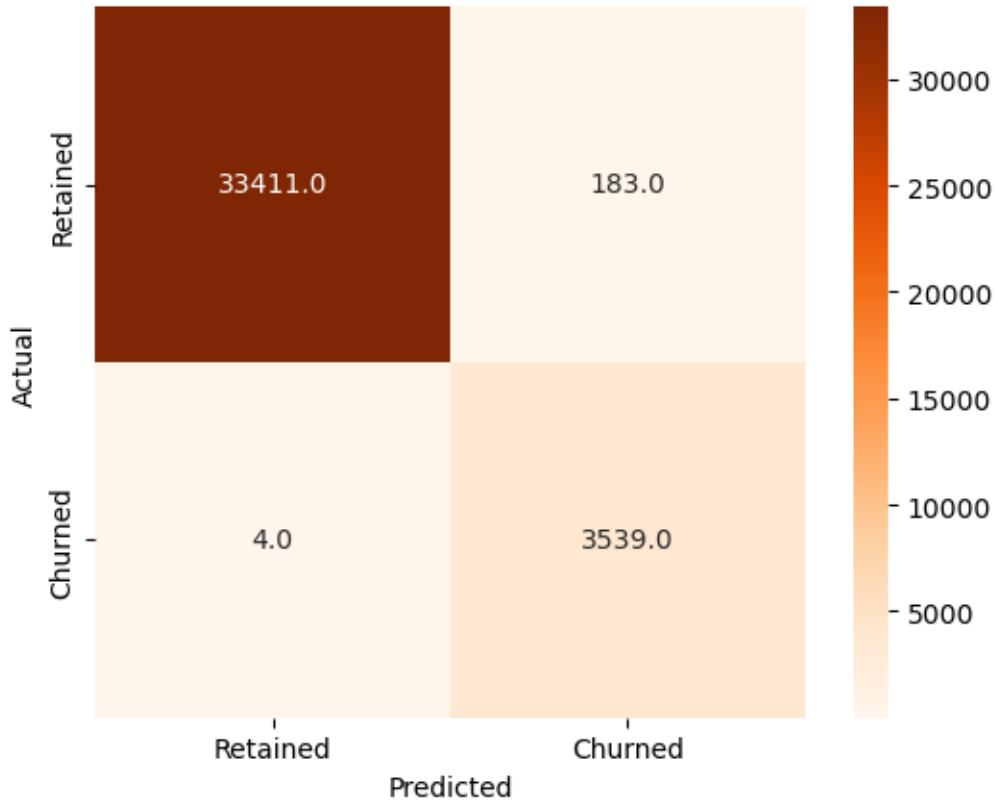
```
[86]: def nn(a,b):

    clreport=classification_report(y_true=a,y_pred=b)
    print("")
    print(f'CLASSIFICATION REPORT: NEURALNETWORK')
    print(clreport)
    print("")
    print(f'CONFUSION MARTIX: NEURALNETWORK')
    v=confusion_matrix(y_true=a,y_pred=b)
    sns.heatmap(v,square=True,annot=True,fmt='.
    ↪f',xticklabels=['Retained','Churned'],yticklabels=['Retained','Churned'],cmap='Oranges')
    plt.ylabel('Actual')
    plt.xlabel('Predicted')
    TP,FP,FN,TN=v.ravel()
    print(f" True Positive: {TP}\n False Positive: {FP}\n False Negative: ↪
    ↪{FN}\n True Negative: {TN}")
    plt.show()
```

```
[87]: nn(ytest,ypredb)
```

	precision	recall	f1-score	support
0	1.00	0.99	1.00	33594
1	0.95	1.00	0.97	3543
accuracy			0.99	37137
macro avg	0.98	1.00	0.99	37137
weighted avg	1.00	0.99	1.00	37137

CONFUSION MARTIX: NEURALNETWORK  
True Positive: 33411  
False Positive: 183  
False Negative: 4  
True Negative: 3539



```
[88]: acc=NeuralNetwork.evaluate(xtest,ytest)
acc
```

```
1161/1161 [=====] - 1s 559us/step - loss: 0.0155 -
precision: 0.9508 - recall: 0.9989 - Accuracy: 0.9950
```

```
[88]: [0.01553292665630579,
 0.9508329033851624,
 0.9988710284233093,
 0.994964599609375]
```

## 12 BEST MODELS IN TERMS OF F1 SCORES PERFORMANCES:

- Neural Network-logistic Regression Model
- Categorical Boost Classifier
- XGBoost Classifier
- GradientBoost Classifier
- KNeighbors Classifier

```
[141]: n=ytest[:10].to_list()
n
```

```
[141]: [0, 0, 0, 0, 0, 0, 0, 1, 0, 0]
```

```
[153]: prob=NeuralNetwork.predict(xtest)
prob.round(3)
```

```
1161/1161 [=====] - 0s 385us/step
```

```
[153]: array([[0.    ],
   [0.    ],
   [0.    ],
   ...,
   [0.    ],
   [0.    ],
   [0.998]], dtype=float32)
```

```
[156]: proba=np.where(prob>=0.5,1,0)
proba[:10]
```

```
[156]: array([[0],
   [0],
   [0],
   [0],
   [0],
   [0],
   [0],
   [1],
   [0],
   [0]])
```