In [1]:
```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_c
import seaborn as sns
import matplotlib.ticker as mtick
import matplotlib.pyplot as plt
%matplotlib inline
```

In [2]:
```
df = pd.read_csv('telecom_churn_data.csv')
df.head()
```

Out[2]:

	mobile_number	circle_id	loc_og_t2o_mou	std_og_t2o_mou	loc_ic_t2o_mou	last_date_c
0	7000842753	109	0.0	0.0	0.0	
1	7001865778	109	0.0	0.0	0.0	
2	7001625959	109	0.0	0.0	0.0	
3	7001204172	109	0.0	0.0	0.0	
4	7000142493	109	0.0	0.0	0.0	

5 rows × 226 columns

In [3]:
```
print(df.circle_id.value_counts())
df.drop('circle_id', axis =1, inplace = True)
```

```
109    99999
Name: circle_id, dtype: int64
```

In [4]:
```
numbers = df.mobile_number
df.drop('mobile_number', axis = 1, inplace=True)
```

In [5]:
```
df.drop(df.loc[df.isnull().sum(axis=1)>150].index, axis = 0, inplac
```

In [6]:
```
df.reset_index(inplace=True)
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 99413 entries, 0 to 99412
Columns: 225 entries, index to sep_vbc_3g
dtypes: float64(179), int64(34), object(12)
memory usage: 170.7+ MB
```

In [7]:
```
df.isnull().sum()
```

Out[7]:
```
index                     0
loc_og_t2o_mou          432
std_og_t2o_mou          432
loc_ic_t2o_mou          432
last_date_of_month_6      0
                       ...
aon                       0
aug_vbc_3g                0
jul_vbc_3g                0
jun_vbc_3g                0
sep_vbc_3g                0
Length: 225, dtype: int64
```

In [8]:
```
#plt.figure(figsize=(12,12))
#sns.heatmap(df.corr(), cmap="Paired")
```

In [9]:
```
df['Churn'] = np.where((df['total_ic_mou_9']==0) & (df['total_og_mou
```

In [10]:
```
df.Churn.nunique()
```

Out[10]: 2

In [11]:
```
df[(df['total_ic_mou_9']==0) & (df['total_og_mou_9']==0) & (df['vol
```

Out[11]:

	total_ic_mou_9	total_og_mou_9	vol_2g_mb_9	vol_3g_mb_9	Churn
0	0.0	0.0	0.0	0.0	1
7	0.0	0.0	0.0	0.0	1
29	0.0	0.0	0.0	0.0	1
32	0.0	0.0	0.0	0.0	1
35	0.0	0.0	0.0	0.0	1
...	...	...	...	...	...
99335	0.0	0.0	0.0	0.0	1
99377	0.0	0.0	0.0	0.0	1
99398	0.0	0.0	0.0	0.0	1
99410	0.0	0.0	0.0	0.0	1
99412	0.0	0.0	0.0	0.0	1

9605 rows × 5 columns

In [12]:
```
col_9 = [i for i in df.columns if i.endswith('_9')]
col_9
df9 = df[col_9]
df_churn = df['Churn']
df9_churn = pd.concat([df9, df_churn], axis=1)
df9_churn.head()
```

Out[12]:

	last_date_of_month_9	arpu_9	onnet_mou_9	offnet_mou_9	roam_ic_mou_9	roam_og_m
0	9/30/2014	21.100	NaN	NaN	NaN	
1	9/30/2014	86.285	18.34	53.76	0.00	
2	9/30/2014	290.714	74.81	118.91	38.49	
3	9/30/2014	389.500	241.71	113.54	0.00	
4	9/30/2014	163.426	58.78	45.81	0.00	

5 rows × 55 columns

In [13]:
```
df9_churn.fillna(0, inplace=True)
```

In [14]:
```
#for i, predictor in enumerate(df9_churn.drop(columns=['Churn'])):
    # plt.figure(i)
     #sns.countplot(data=df9_churn, x=predictor, hue='Churn')
```

In [15]:
```
df = df.drop(col_9, axis =1)
```

In [16]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 99413 entries, 0 to 99412
Columns: 172 entries, index to Churn
dtypes: float64(136), int64(27), object(9)
memory usage: 130.5+ MB
```

In [17]:
```
df[['av_rech_amt_data_7','total_rech_num_7','total_rech_amt_7','to
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 99413 entries, 0 to 99412
Data columns (total 4 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   av_rech_amt_data_7  25567 non-null  float64
 1   total_rech_num_7    99413 non-null  int64  
 2   total_rech_amt_7    99413 non-null  int64  
 3   total_rech_data_7   25567 non-null  float64
dtypes: float64(2), int64(2)
memory usage: 3.0 MB
```

In [18]:
```
col_7 = ['av_rech_amt_data_7','total_rech_num_7', 'total_rech_amt_7
df[col_7].head()
```

Out[18]:

	av_rech_amt_data_7	total_rech_num_7	total_rech_amt_7	total_rech_data_7
0	252.0	3	252	1.0
1	154.0	9	384	1.0
2	NaN	4	315	NaN
3	NaN	11	310	NaN
4	NaN	6	350	NaN

In [19]:
```
col_6 = ['av_rech_amt_data_6','total_rech_num_6', 'total_rech_amt_6
df[col_6].head()
```

Out[19]:

	av_rech_amt_data_6	total_rech_num_6	total_rech_amt_6	total_rech_data_6
0	252.0	4	362	1.0
1	NaN	4	74	NaN
2	NaN	5	168	NaN
3	NaN	10	230	NaN
4	56.0	5	196	1.0

In [20]:
```
print(df[col_7].info())
print(df[col_6].info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 99413 entries, 0 to 99412
Data columns (total 4 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   av_rech_amt_data_7  25567 non-null  float64
 1   total_rech_num_7    99413 non-null  int64  
 2   total_rech_amt_7    99413 non-null  int64  
 3   total_rech_data_7   25567 non-null  float64
dtypes: float64(2), int64(2)
memory usage: 3.0 MB
None
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 99413 entries, 0 to 99412
Data columns (total 4 columns):
 #   Column              Non-Null Count  Dtype  
```

In [21]:
```
df[col_7]=df[col_7].fillna(0)
df[col_6]=df[col_6].fillna(0)
```

In [22]:
```
print(df[col_7].info())
print(df[col_6].info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 99413 entries, 0 to 99412
Data columns (total 4 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   av_rech_amt_data_7  99413 non-null  float64
 1   total_rech_num_7    99413 non-null  int64  
 2   total_rech_amt_7    99413 non-null  int64  
 3   total_rech_data_7   99413 non-null  float64
dtypes: float64(2), int64(2)
memory usage: 3.0 MB
None
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 99413 entries, 0 to 99412
Data columns (total 4 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   av_rech_amt_data_6  99413 non-null  float64
 1   total_rech_num_6    99413 non-null  int64  
 2   total_rech_amt_6    99413 non-null  int64  
 3   total_rech_data_6   99413 non-null  float64
dtypes: float64(2), int64(2)
memory usage: 3.0 MB
None
```

In [23]:
```
df['total_amount_7']= df['total_rech_data_7']+(df['total_rech_data_
```

In [24]:
```
df['total_amount_6']= df['total_rech_data_6']+(df['total_rech_data_(
```

In [25]:
```
df['avg_amt_7_6'] = (df['total_amount_7']+df['total_amount_6'])/2.0
```

In [26]:
```
per_70 = (df['avg_amt_7_6'].quantile(0.7))
```

In [27]:
```
df = df[df['avg_amt_7_6']>=per_70]
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 30448 entries, 0 to 99412
Columns: 175 entries, index to avg_amt_7_6
dtypes: float64(139), int64(27), object(9)
```

In [28]:
```
high_null = df.columns[df.isnull().sum()>=9000]
high_null
```

Out[28]:
```
Index(['date_of_last_rech_data_8', 'total_rech_data_8', 'max_rech_
data_8',
       'count_rech_2g_8', 'count_rech_3g_8', 'av_rech_amt_data_8',
'arpu_3g_8',
       'arpu_2g_8', 'night_pck_user_8', 'fb_user_8'],
      dtype='object')
```

In [29]:
```
df[high_null]=df[high_null].fillna(0)
```

In [30]:
```
col = df.columns[df.isnull().sum()>=1500]
df[col] = df[col].fillna(0)
```

In [31]:
```
sum(df.isnull().sum(axis=1)>50)
```

Out[31]:
```
540
```

In [32]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 30448 entries, 0 to 99412
Columns: 175 entries, index to avg_amt_7_6
dtypes: float64(139), int64(27), object(9)
memory usage: 40.9+ MB
```

In [33]:
```
df.reset_index(inplace=True,drop=True)
df.head()
```

Out[33]:

	index	loc_og_t2o_mou	std_og_t2o_mou	loc_ic_t2o_mou	last_date_of_month_6	last_dat
0	0	0.0	0.0	0.0	6/30/2014	
1	1	0.0	0.0	0.0	6/30/2014	
2	4	0.0	0.0	0.0	6/30/2014	
3	8	0.0	0.0	0.0	6/30/2014	
4	9	0.0	0.0	0.0	6/30/2014	

5 rows × 175 columns

In [34]:
```
df.drop('index',inplace=True,axis=1)
```

In [35]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 30448 entries, 0 to 30447
Columns: 174 entries, loc_og_t2o_mou to avg_amt_7_6
dtypes: float64(139), int64(26), object(9)
memory usage: 40.4+ MB
```

In [36]:
```
df[df.isnull().sum(axis=1)>60].head()
```

Out[36]:

	loc_og_t2o_mou	std_og_t2o_mou	loc_ic_t2o_mou	last_date_of_month_6	last_date_of_
203	NaN	NaN	NaN	6/30/2014	
688	NaN	NaN	NaN	6/30/2014	
715	NaN	NaN	NaN	6/30/2014	
717	NaN	NaN	NaN	6/30/2014	
975	NaN	NaN	NaN	6/30/2014	

5 rows × 174 columns

In [37]:
```
df.fillna(0,inplace=True)
df.head()
```

Out[37]:

	loc_og_t2o_mou	std_og_t2o_mou	loc_ic_t2o_mou	last_date_of_month_6	last_date_of_m
0	0.0	0.0	0.0	6/30/2014	7/3
1	0.0	0.0	0.0	6/30/2014	7/3
2	0.0	0.0	0.0	6/30/2014	7/3
3	0.0	0.0	0.0	6/30/2014	7/3
4	0.0	0.0	0.0	6/30/2014	7/3

5 rows × 174 columns

In [38]:
```
df.to_csv('telChurn.csv',index=False)
```

In [ ]:
```
```