IMPORTING PYTHON LIBRARIES

In [31]:

```
import pandas as pd
import numpy as np
from pandas import datetime
```

```
<ipython-input-31-9b7fb8b3763a>:3: FutureWarning: The pandas.datetime class is deprecated and will be r
emoved from pandas in a future version. Import from datetime module instead.
  from pandas import datetime
```

IMPORTING THE DATASET OF HOUSEHOLD POWER CONSUMPTION

In [32]:

```
## Choosing index column as date_time because it is Time series data set
## Its is having dates so parse dates is true
## Its a large file so low memory is false
df = pd.read_csv('household_power_consumption.txt', sep = ';', parse_dates= ['Date'], infer_datetime_for
```

TO CHECK DATA TYPES

In [4]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2075259 entries, 0 to 2075258
Data columns (total 9 columns):
 #   Column                 Dtype
---  ------                 -----
 0   Date                   datetime64[ns]
 1   Time                   object
 2   Global_active_power    float64
 3   Global_reactive_power  float64
 4   Voltage                float64
 5   Global_intensity       float64
 6   Sub_metering_1         float64
 7   Sub_metering_2         float64
 8   Sub_metering_3         float64
dtypes: datetime64[ns](1), float64(7), object(1)
memory usage: 142.5+ MB
```

TO CHECK NULL VALUES

In [5]:

```
df.isna().sum()
```

Out[5]:

```
Date                         0
Time                         0
Global_active_power      25979
Global_reactive_power    25979
Voltage                  25979
Global_intensity         25979
Sub_metering_1           25979
Sub_metering_2           25979
Sub_metering_3           25979
dtype: int64
```

DROPPING THE NULL VALUES- DATA CLEANING

In [6]:
```
df = df.dropna()
df.isna().sum()
```

Out[6]:
```
Date                     0
Time                     0
Global_active_power      0
Global_reactive_power    0
Voltage                  0
Global_intensity         0
Sub_metering_1           0
Sub_metering_2           0
Sub_metering_3           0
dtype: int64
```

THE CLEANED DATASET AND ITS ATTRIBUTES

In [7]:
```
df
```

Out[7]:

	Date	Time	Global_active_power	Global_reactive_power	Voltage	Global_intensity	Sub_metering_1	Sub_metering_
0	2006-12-16	17:24:00	4.216	0.418	234.84	18.4	0.0	1.
1	2006-12-16	17:25:00	5.360	0.436	233.63	23.0	0.0	1.
2	2006-12-16	17:26:00	5.374	0.498	233.29	23.0	0.0	2.
3	2006-12-16	17:27:00	5.388	0.502	233.74	23.0	0.0	1.
4	2006-12-16	17:28:00	3.666	0.528	235.68	15.8	0.0	1.
...	...	...	...	...	...	...	...	.
2075254	2010-11-26	20:58:00	0.946	0.000	240.43	4.0	0.0	0.
2075255	2010-11-26	20:59:00	0.944	0.000	240.00	4.0	0.0	0.
2075256	2010-11-26	21:00:00	0.938	0.000	239.82	3.8	0.0	0.
2075257	2010-11-26	21:01:00	0.934	0.000	239.70	3.8	0.0	0.
2075258	2010-11-26	21:02:00	0.932	0.000	239.55	3.8	0.0	0.

2049280 rows × 9 columns

IMPORTING LIBRARIES FOR DATA VISUALIZATION

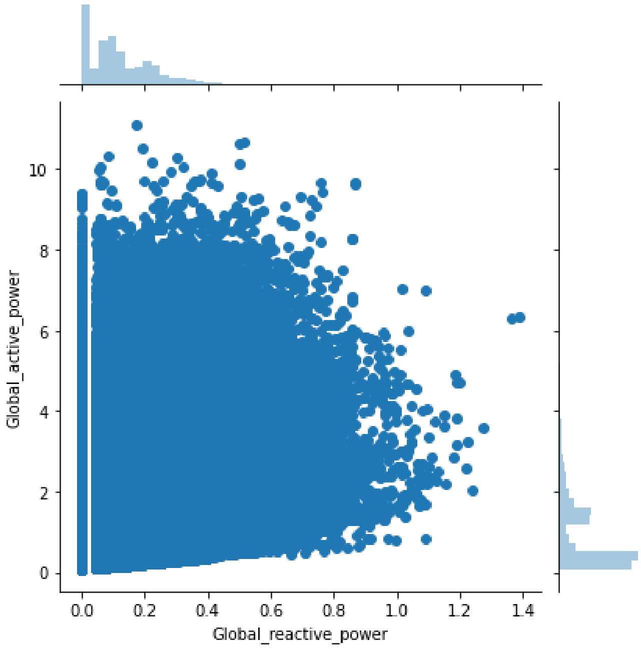
In [21]:
```
import pandas.testing as tm
import matplotlib.pyplot as plt
import seaborn as sns
import statsmodels.api as sm
```

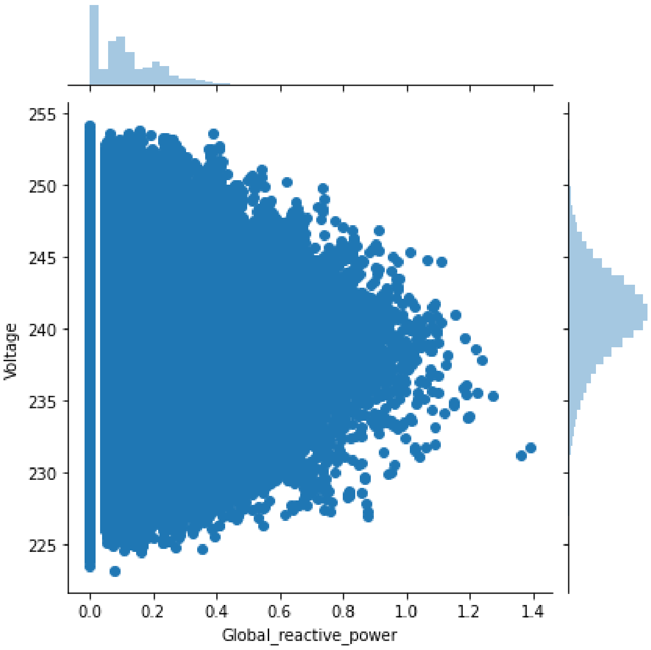
In [26]:

```
sns.jointplot(x='Global_reactive_power',y='Global_active_power',data=df,kind='scatter'),
sns.jointplot(x='Global_reactive_power',y='Voltage',data=df,kind='scatter'),
sns.jointplot(x='Global_intensity',y='Global_active_power',data=df,kind='scatter'),
sns.jointplot(x='Sub_metering_1',y='Global_active_power',data=df,kind='scatter'),
sns.jointplot(x='Sub_metering_2',y='Global_active_power',data=df,kind='scatter'),
sns.jointplot(x='Sub_metering_3',y='Global_active_power',data=df,kind='scatter'),
plt.figure(figsize=(20,10))
df.corrwith(df['Global_active_power']).plot.bar(grid=True,rot=45),
plt.figure(figsize=(10,10))
sns.heatmap(df.corr(),cmap='viridis',annot=True),
```

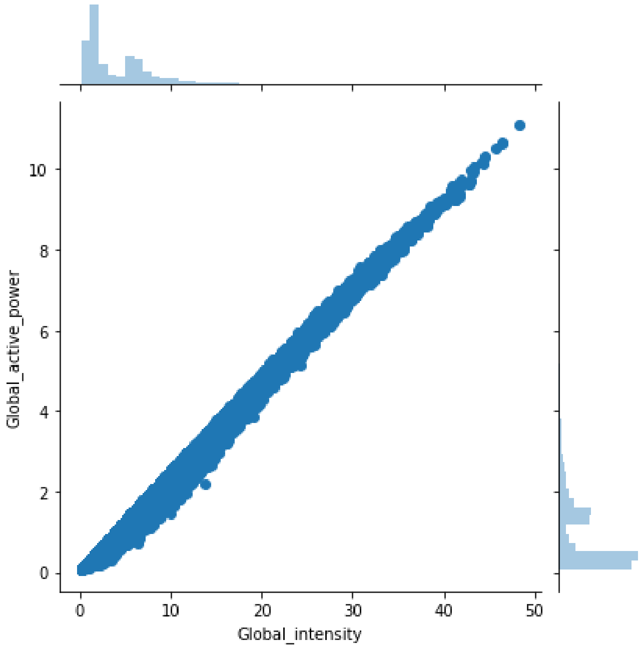
Out[26]:

```
(<matplotlib.axes._subplots.AxesSubplot at 0x2098024bf10>,)
```

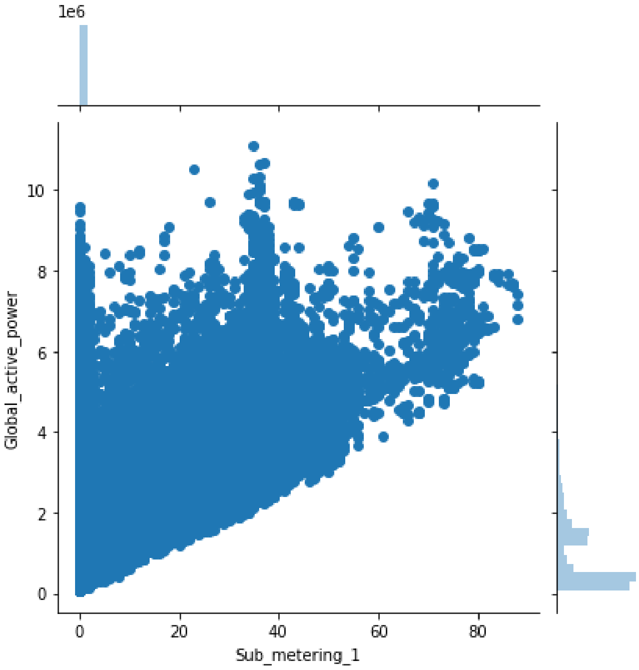


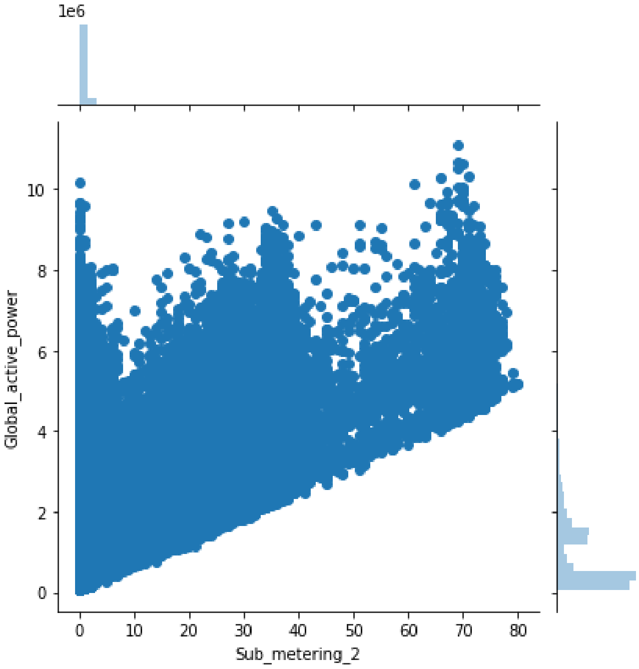


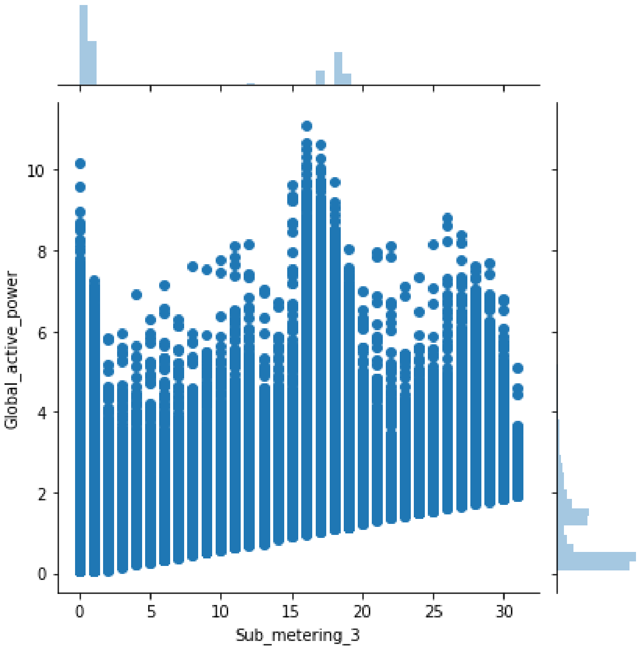
255
250
245
240
235
230
225
Voltage
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
Global_reactive_power

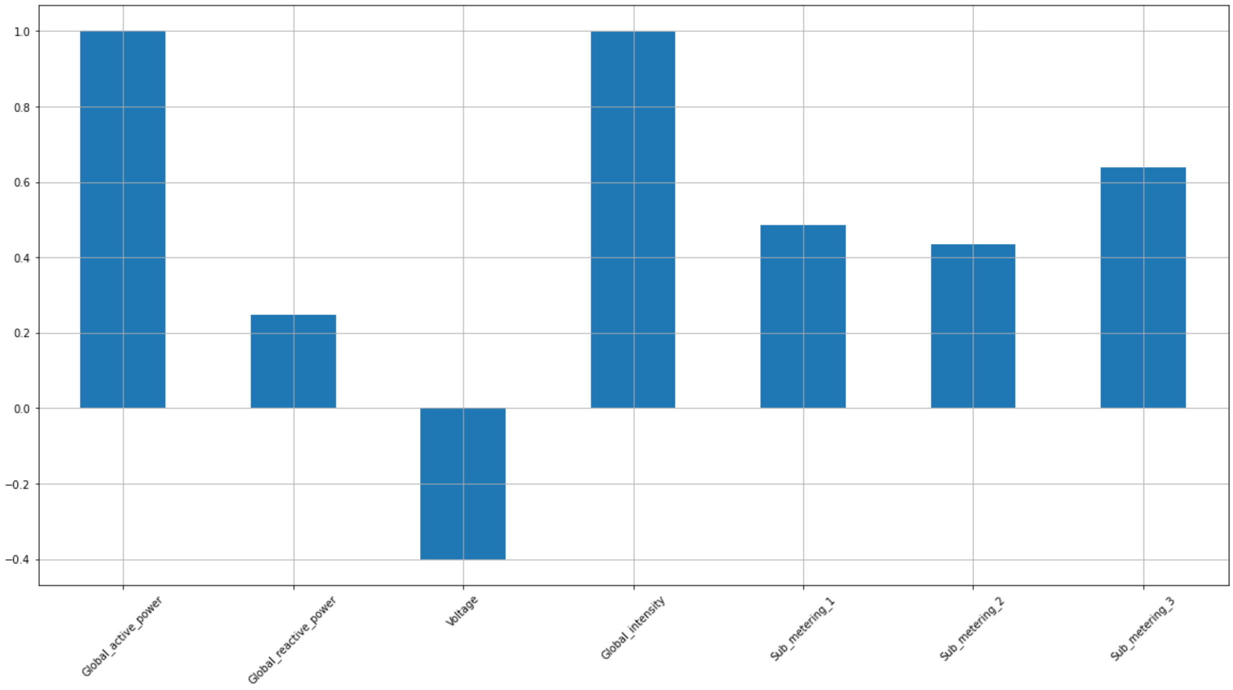


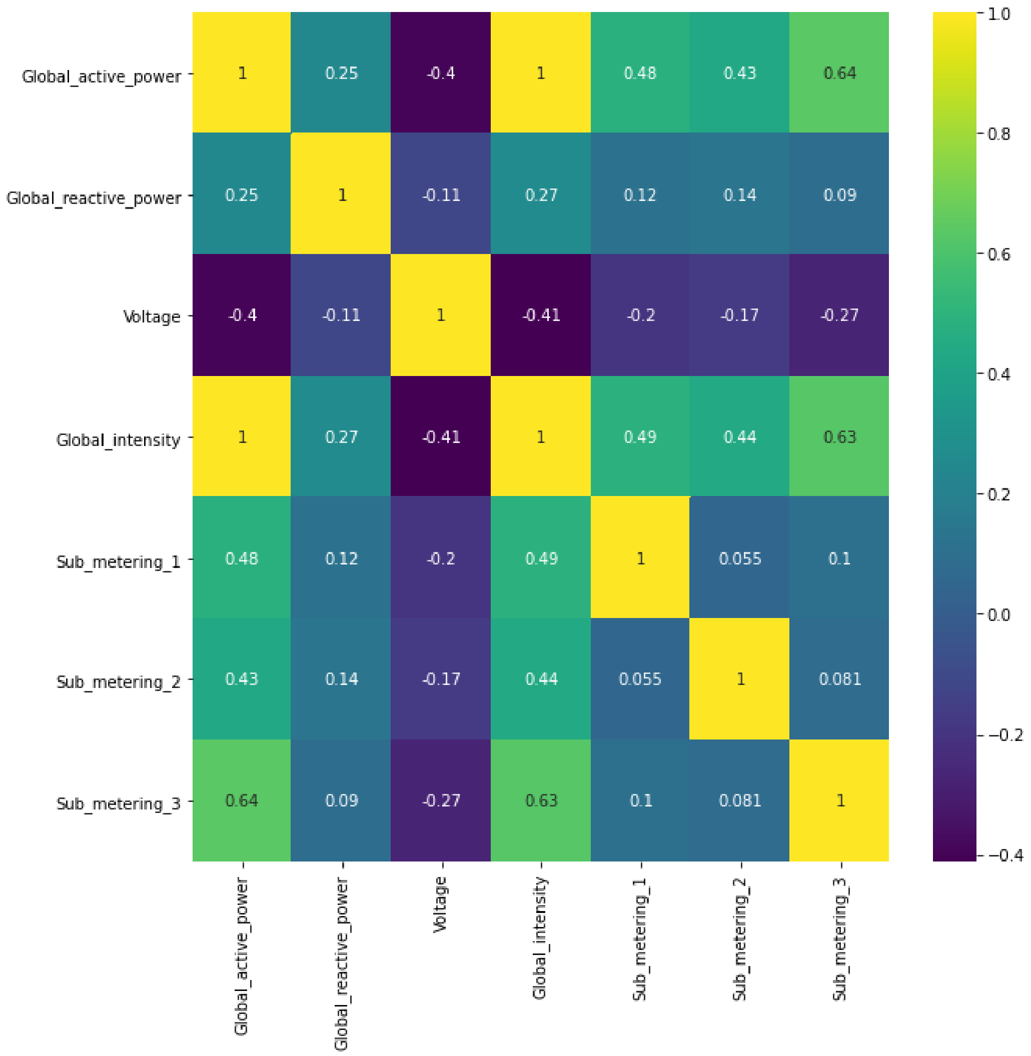
10
8
6
4
2
0
Global_active_power
0
10
20
30
40
50
Global_intensity


1e6
Global_active_power
0
2
4
6
8
10
0
20
40
60
80
Sub_metering_1


1e6
Global_active_power
0
2
4
6
8
10
0
10
20
30
40
50
60
70
80
Sub_metering_2


Global_active_power
0
2
4
6
8
10
0
5
10
15
20
25
30
Sub_metering_3


1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
Global_active_power
Global_reactive_power
Voltage
Global_intensity
Sub_metering_1
Sub_metering_2
Sub_metering_3


Global_active_power 1 0.25 -0.4 1 0.48 0.43 0.64
Global_reactive_power 0.25 1 -0.11 0.27 0.12 0.14 0.09
Voltage -0.4 -0.11 1 -0.41 -0.2 -0.17 -0.27
Global_intensity 1 0.27 -0.41 1 0.49 0.44 0.63
Sub_metering_1 0.48 0.12 -0.2 0.49 1 0.055 0.1
Sub_metering_2 0.43 0.14 -0.17 0.44 0.055 1 0.081
Sub_metering_3 0.64 0.09 -0.27 0.63 0.1 0.081 1
Global_active_power
Global_reactive_power
Voltage
Global_intensity
Sub_metering_1
Sub_metering_2
Sub_metering_3
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4