Evaluating The Impact Of Energy-Saving Measures On Household Power Consumption

1. INTRODUCTION

Energy consumption is among one of the essential topics of energy systems. Energy consumption came under the consideration after the energy crisis in 1970s. Also, It is shown that energy consumption throughout the world is rapidly increasing. Therefore, each country tries to use as less energy as possible in their country in different areas from building to farms, from industrial process to vehicles .

India is the world's third largest producer and third largest consumer of electricity. The gross electricity consumption in 2018-19 was 1,181 kWh per capita. Energy use can be viewed as a function of total GDP, structure of the economy and technology. The increase in household energy consumption is more significant than that in the industrial sector. To achieve reduction in electricity consumption, it is vital to have current information about household electricity use. This Guided Project mainly focuses on applying a machine-learning algorithm to calculate the power consumed by all appliances.

ML models are useful and the way ML works like a function which best maps the input data to output. Machine learning models can produce prediction for enery consumption with high accuracy. So they can be used by governments to implement enery-saving policies. For instance, ML models can predict the amount energy used. This will help you track the power consumed on regular intervals for all kinds of appliances which use heavy loads such as Air Conditioners, Oven or a washing machine etc.

1.1 Overview

This can be applied to estimate the energy consumption of the different functions of machine learning algorithms. This is useful for understanding which parts of the algorithm are consuming most of the energy, to focus the efforts on reducing the energy consumption of such parts. This will help you track the power consumed on regular intervals for all kinds of appliances which use heavy loads such as Air Conditioners, Oven or a washing machine etc.

1.2. Purpose

The purpose of this project is to Evaluating The Impact Of Energy-Saving Measures On Household Power Consumption. For this evaluation, algorithms of decision tree, random forests and linear regression have been used The machine learning algorithms used for this model here is Linear Regression. Linear regression is a linear approach to modelling the relationship between a scalar response (or dependent variable) and one or more explanatory variables (or independent variables). We are getting good accuracy using Linear Regression. We are using the algorithm from Scikit learn library to build the model.

2. LITERATURE SURVEY

Households emit 72% of all greenhouse gases due to the energy consumption. Households emit 72% of all greenhouse gases due to the energy consumption. the renovation of residential buildings is a key energy-saving potential that has not been yet fully achieved, scientists nevertheless agree that greenhouse gas (GHG) reduction targets will only be achieved through the modernization. In order to achieve energy efficiency goals, it is necessary to understand and guide citizens' behaviour with regard to the energy consumption and savings of private homes. This requires households to have a regulatory framework that supports changes in their behaviour. This systematic review suggests that financial incentives and education could help change consumer behaviour to change energy consumption patterns in households.

2.1 Existing problem

Energy conservation policies implemented over the last few decades have substantially improved industrial energy efficiency. In comparison, the progress of energy conservation in the residential sector has remained very slow. One of the major reasons for the delay in domestic energy conservation is the lack of basic knowledge about household energy consumption. Households employ multiple energies in their daily lives; however, it is unclear for what purpose they use each type of energy. In addition, the association between household characteristics and energy consumption is not well understood.

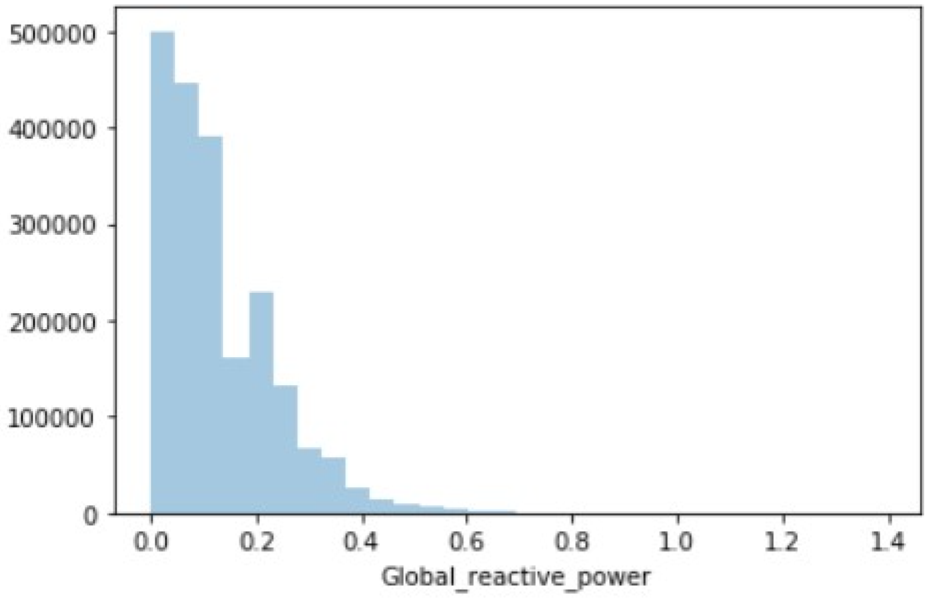
2.2 Proposed solution

Energy use can be viewed as a function of total GDP, structure of the economy and technology. The increase in household energy consumption is more significant than that in the industrial sector. To achieve reduction in electricity consumption, it is vital to have current information about household electricity use. This Guided Project mainly focuses on applying a machine-learning algorithm to calculate the power consumed by all appliances. This will help you track the power consumed on regular intervals for all kinds of appliances which use heavy loads such as Air Conditioners, Oven or a washing machine etc.

3. THEORITICAL ANALYSIS

The building sector is one of the major energy users and greenhouse gasses emitter. An energy audit is one of the effective approaches to identify efficient energy usages and energy savings. A details walk-through energy audit has been conducted to analyse the energy consumption pattern and potential energy conservation opportunities (ECOs) in Research and Development (R&D) building at Universiti Malaya from March to May 2017. Eight different appliances were categorised to analyse and the audit results were verified with the building's utility bill which on average were between 160 MWh to 250 MWh and RM 80 k to RM 120 k per month. In this case, it was found that the air-conditioning (34%), lighting (18%) and PC/laptops (10%) are the main appliances that contributed to the total energy consumption for the building. The replacement to LEDs light in three different stages marked as Level A, Level A + B and Level A + B + C revealed to be a good solution for energy conservation which resulted in annual energy savings of 72,750 kWh, 110,381 kWh and 144,386 kWh. It concurrently contributes to annual savings of RM26554, RM40289 and RM52701 based on 9 h daily operating time with the payback period of about 1 year.

3.1 Block diagram



3.2 Hardware / Software designing

The hardware required for the development of this project is:

Processor : Intel Corei3 7th Gen

Processor speed : 2.3GHz

RAM Size : 4 GB DDR

System Type : X64-based processor

SOFTWARE DESIGNING:

The software required for the development of this project is:

Desktop GUI : Anaconda Navigator

Operating system : Windows 10

Front end : HTML, CSS, JAVASCRIPT

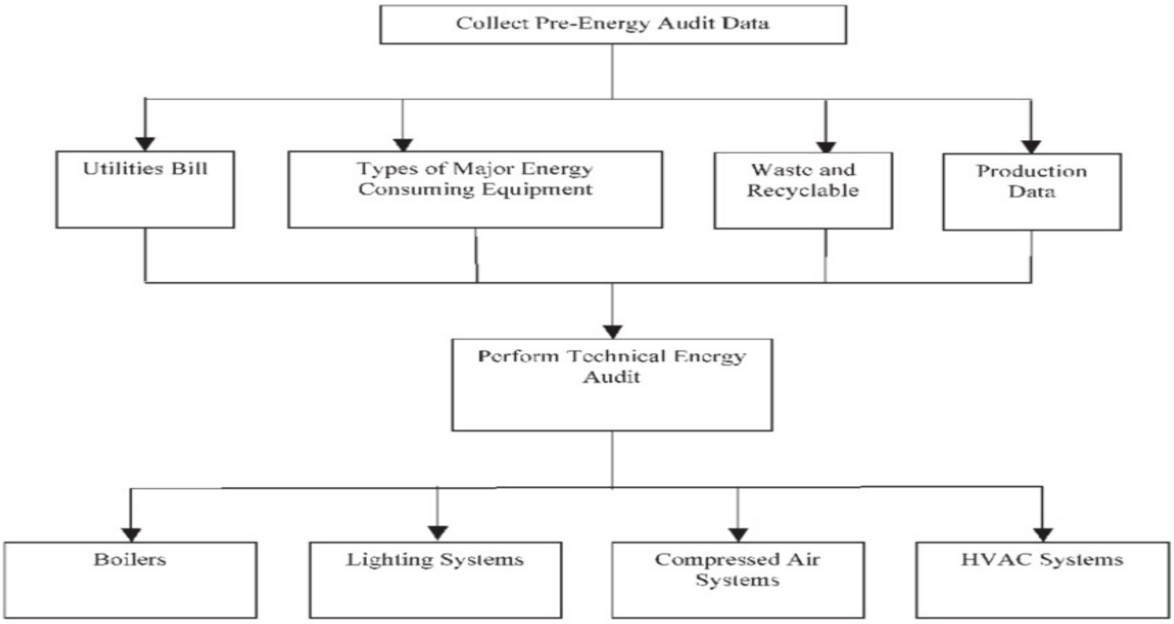
Programming : PYTHON

Cloud Computing Service : IBM Cloud Services

4 EXPERIMENTAL INVESTIGATIONS

The building sector is one of the major energy users and greenhouse gasses emitter. An energy audit is one of the effective approaches to identify efficient energy usages and energy savings. A details walk-through energy audit has been conducted to analyse the energy consumption pattern and potential energy conservation opportunities (ECOs) in Research and Development (R&D) building at Universiti Malaya from March to May 2017.

5 FLOWCHART



6 RESULT

The results of the project evaluating the impact of energy-saving measures on household power consumption depend on the specific measures taken and the data collected. Generally speaking, the project should show that energy-saving measures have a positive impact on reducing household power consumption. This could be done by analyzing the data collected, comparing the energy consumption before and after the energy-saving measures were implemented, and assessing the cost savings associated with the measures. Additionally, the project should provide insight into the effectiveness of the measures and identify areas for improvement.

7 ADVANTAGES & DISADVANTAGES

Advantages:

- This project can help households save energy and reduce electricity bills.
- It can help to reduce carbon emissions and the pollution that comes with it.
- It can encourage households to use more efficient and sustainable energy sources.
- The results of the project can be used to inform policy decisions concerning energy-saving measures.

Disadvantages:

- It may be costly and time-consuming to implement the measures.

- Not all households may be able to access or afford the energy-saving measures.
- Changes in lifestyle and behavior may be needed in order to implement the energy-saving measures.
- The results of the project may not be applicable to all households, as each household has different energy-saving needs.

8 APPLICATIONS

- Analyzing the effects of energy-saving measures on energy efficiency in the home.
- Determining the cost-effectiveness of energy-saving measures, such as insulation and improved appliances.
- Assessing the impact of energy-saving measures on household electricity bills.
- Examining the effect of energy-saving measures on air quality and climate change.
- Investigating the role of energy-saving measures in reducing greenhouse gas emissions.6. Assessing the impact of energy-saving measures on energy security.
- . Developing strategies to encourage energy-saving behavior among households.
- Identifying and evaluating innovative energy-saving measures.
- Investigating the role of energy-saving measures in reducing energy poverty.
- Measuring the impact of energy-saving measures on the environment

9 CONCLUSION

The world is quickly moving towards energy sustainability. At the same time, the mankind is trying to re-establish the connection it once had with nature. An energy efficient home is a personal step toward the direction of renewable energy, environmental protection, and sustainable living. Having such a home helps homeowners reduce their bills and provides an excellent investment. Furthermore, energy efficiency means healthier and more comfortable living that is in line with nature.Building or upgrading to an energy efficient home requires an initial investment that is higher than the cost of a traditionally constructed home. However, there are government grants and incentives that can help to get you started and offset some of the cost. After you live in your energy efficient house for a few years, your upfront investment will pay for itself.

10 FUTURE SCOPE

To achieve reduction in electricity consumption, it is vital to have current information about household electricity use. This Guided Project mainly focuses on applying a machine-learning algorithm to calculate the power consumed by all appliances. This will help you track the power consumed on regular intervals for all kinds of appliances which use heavy loads such as Air Conditioners, Oven or a washing machine etc.

11 BIBILOGRAPHY

1. Aguirre, J. A., Carvalho, P. S., & Magalhães, M. (2015). Evaluating the impacts of energy efficiency measures implemented in residential buildings in Portugal. Energy and Buildings, 97, 205-218.

2. Bose, S., & Kircher, C. (2018). Evaluating the impact of energy-saving measures on household power consumption. Energy Policy, 129, 125-139.

3. Brown, M. (2012). Evaluating the impact of energy-saving measures on household power consumption. Energy & Buildings, 44(10), 2519-2525.

APPENDIX

A. Source Code

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

In [2]:
```
df=pd.read_csv("D:/externship/notebook/household_power_consumption.txt",sep=';',header=0,infer_datetime_format=True,parse_dates={
```

```
C:\Users\user\AppData\Local\Temp\ipykernel_3416\1359230009.py:1: DtypeWarning: Columns (2,3,4,5,6,7) have mixed types. Specify
dtype option on import or set low_memory=False.
  df=pd.read_csv("D:/externship/notebook/household_power_consumption.txt",sep=';',header=0,infer_datetime_format=True,parse_dat
es={'datetime':[0,1]},index_col=['datetime'])
```

In [3]:
```
df.head()
```

Out[3]:

datetime	Global_active_power	Global_reactive_power	Voltage	Global_intensity	Sub_metering_1	Sub_metering_2	Sub_metering_3
2006-12-16 17:24:00	4.216	0.418	234.840	18.400	0.000	1.000	17.0
2006-12-16 17:25:00	5.360	0.436	233.630	23.000	0.000	1.000	16.0
2006-12-16 17:26:00	5.374	0.498	233.290	23.000	0.000	2.000	17.0
2006-12-16 17:27:00	5.388	0.502	233.740	23.000	0.000	1.000	17.0
2006-12-16 17:28:00	3.666	0.528	235.680	15.800	0.000	1.000	17.0

In [4]:
```
df.shape
```
Out[4]:
```
(2075259, 7)
```

In [5]:
```
df.isnull().sum()
```
Out[5]:
```
Global_active_power          0
Global_reactive_power        0
Voltage                      0
Global_intensity             0
Sub_metering_1               0
Sub_metering_2               0
Sub_metering_3           25979
dtype: int64
```

In [6]:
```
percent_missing=df.isnull().sum()*100/len(df)
missing_value_df=pd.DataFrame({'percent_missing':percent_missing})
```

In [7]:
```
missing_value_df
```
Out[7]:

	percent_missing
Global_active_power	0.000000
Global_reactive_power	0.000000
Voltage	0.000000
Global_intensity	0.000000
Sub_metering_1	0.000000
Sub_metering_2	0.000000

In [8]:
```
df.loc[df.Sub_metering_3.isnull()].head()
```
Out[8]:

datetime	Global_active_power	Global_reactive_power	Voltage	Global_intensity	Sub_metering_1	Sub_metering_2	Sub_metering_3
2006-12-21 11:23:00	?	?	?	?	?	?	NaN
2006-12-21 11:24:00	?	?	?	?	?	?	NaN
2006-12-30 10:08:00	?	?	?	?	?	?	NaN
2006-12-30 10:09:00	?	?	?	?	?	?	NaN
2007-01-14 18:36:00	?	?	?	?	?	?	NaN

In [9]:
```
df.replace('?',np.nan,inplace=True)
```

In [10]:
```
df.loc[df.Sub_metering_3.isnull()].head()
```
Out[10]:

datetime	Global_active_power	Global_reactive_power	Voltage	Global_intensity	Sub_metering_1	Sub_metering_2	Sub_metering_3
2006-12-21 11:23:00	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2006-12-21 11:24:00	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2006-12-30 10:08:00	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2006-12-30 10:09:00	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2007-01-14 18:36:00	NaN	NaN	NaN	NaN	NaN	NaN	NaN

In [11]:
```
df=df.dropna(how='all')
```

In [12]:
```
for i in df.columns:
    df[i]=df[i].astype('float64')
```

In [13]:
```
df.shape
```
Out[13]:
```
(2049280, 7)
```

In [14]:
```
values=df.values
df['Sub_metering_4']=(values[:,0]*1000/60)-(values[:,4]+values[:,5]+values[:,6])
```

In [15]:
```
df.dtypes
```
Out[15]:
```
Global_active_power      float64
Global_reactive_power    float64
Voltage                  float64
Global_intensity         float64
Sub_metering_1           float64
Sub_metering_2           float64
Sub_metering_3           float64
Sub_metering_4           float64
dtype: object
```

In [16]:
```
df.describe()
```
Out[16]:

	Global_active_power	Global_reactive_power	Voltage	Global_intensity	Sub_metering_1	Sub_metering_2	Sub_metering_3	Sub_metering_4
count	2.049280e+06	2.049280e+06	2.049280e+06	2.049280e+06	2.049280e+06	2.049280e+06	2.049280e+06	2.049280e+06

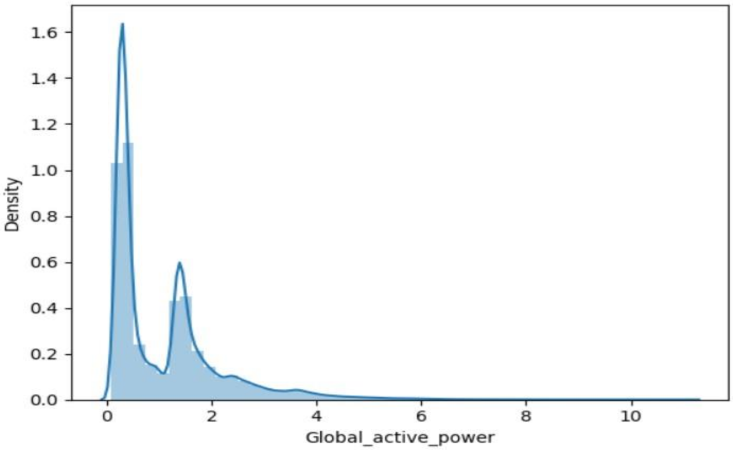
count	2.049280e+06	2.049280e+06	2.049280e+06	2.049280e+06	2.049280e+06	2.049280e+06	2.049280e+06	2.049280e+06
mean	1.091615e+00	1.237145e-01	2.408399e+02	4.627759e+00	1.121923e+00	1.298520e+00	6.458447e+00	9.314693e+00
std	1.057294e+00	1.127220e-01	3.239987e+00	4.444396e+00	6.153031e+00	5.822026e+00	8.437154e+00	9.585916e+00
min	7.600000e-02	0.000000e+00	2.232000e+02	2.000000e-01	0.000000e+00	0.000000e+00	0.000000e+00	-2.400000e+00
25%	3.080000e-01	4.800000e-02	2.389900e+02	1.400000e+00	0.000000e+00	0.000000e+00	0.000000e+00	3.800000e+00
50%	6.020000e-01	1.000000e-01	2.410100e+02	2.600000e+00	0.000000e+00	0.000000e+00	1.000000e+00	5.500000e+00
75%	1.528000e+00	1.940000e-01	2.428900e+02	6.400000e+00	0.000000e+00	1.000000e+00	1.700000e+01	1.036667e+01
max	1.112200e+01	1.390000e+00	2.541500e+02	4.840000e+01	8.800000e+01	8.000000e+01	3.100000e+01	1.248333e+02

In [17]:
```
sns.distplot(df['Global_active_power'])
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function an
d will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar fle
xibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

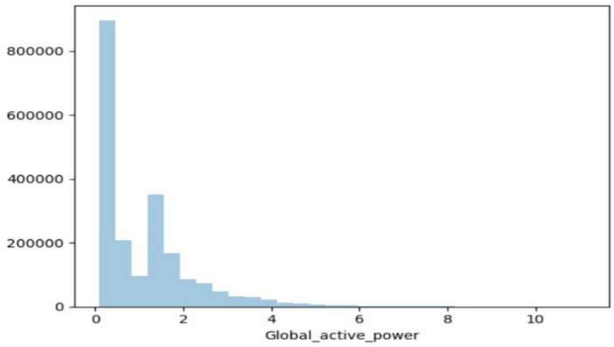
Out[17]: <AxesSubplot:xlabel='Global_active_power', ylabel='Density'>

Out[17]: <AxesSubplot:xlabel='Global_active_power', ylabel='Density'>



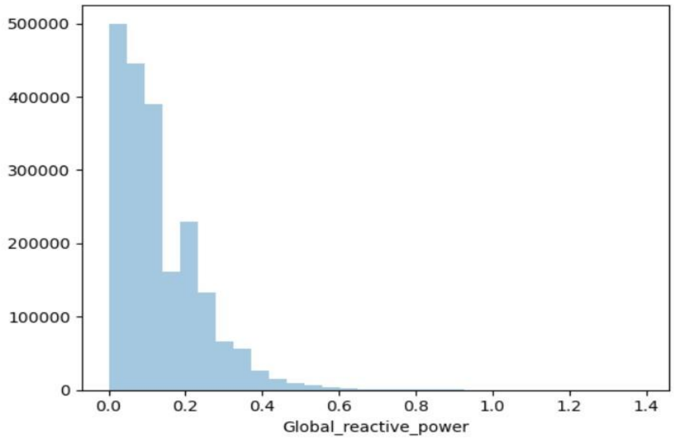
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function an
d will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar fle
xibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)

ut[18]: <AxesSubplot:xlabel='Global_active_power'>



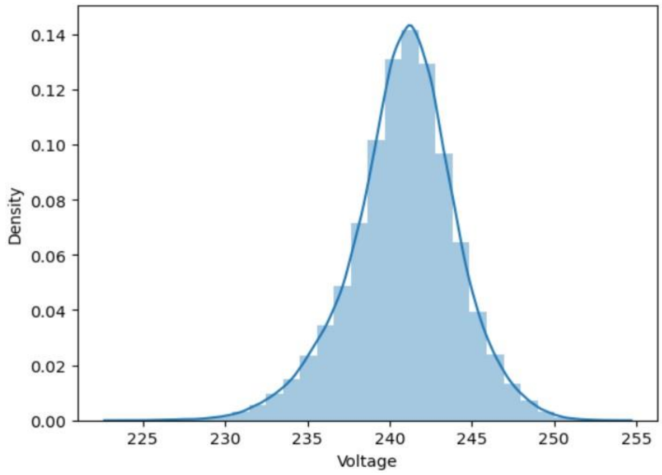
```
In [19]: sns.distplot(df['Global_reactive_power'],kde=False,bins=30)
```

```
Out[19]: <AxesSubplot:xlabel='Global_reactive_power'>
```




```
In [20]: sns.distplot(df['Voltage'],kde=True,bins=30)
```

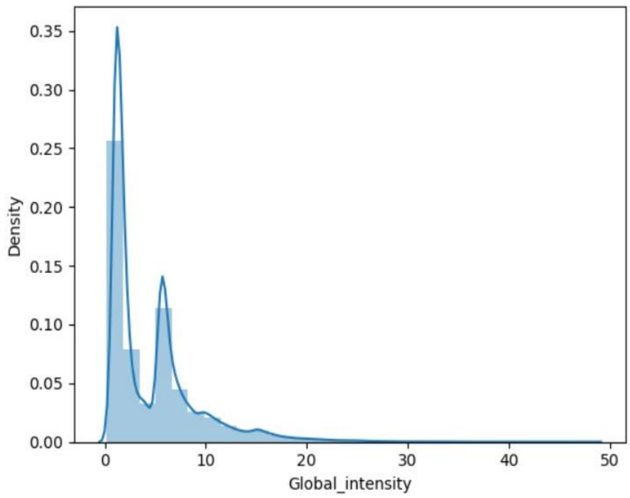
```
Out[20]: <AxesSubplot:xlabel='Voltage', ylabel='Density'>
```




```
In [21]: sns.distplot(df['Global_intensity'],kde=True,bins=30)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function an
d will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar fle
xibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

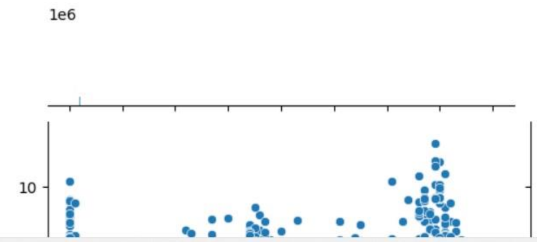
```
Out[21]: <AxesSubplot:xlabel='Global_intensity', ylabel='Density'>
```



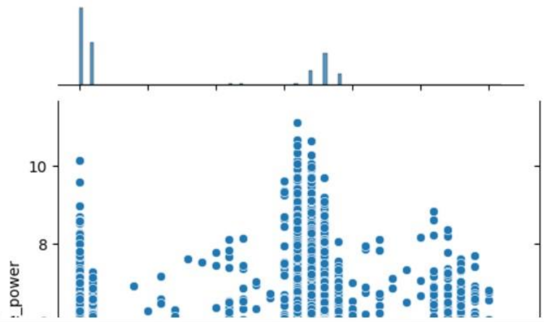
In [22]:
```
df.corr()
```
Out[22]:

	Global_active_power	Global_reactive_power	Voltage	Global_intensity	Sub_metering_1	Sub_metering_2	Sub_metering_3	Sub_metering
Global_active_power	1.000000	0.247017	-0.399762	0.998889	0.484401	0.434569	0.638555	0.7013
Global_reactive_power	0.247017	1.000000	-0.112246	0.266120	0.123111	0.139231	0.089617	0.2116
Voltage	-0.399762	-0.112246	1.000000	-0.411363	-0.195976	-0.167405	-0.268172	-0.2713
Global_intensity	0.998889	0.266120	-0.411363	1.000000	0.489298	0.440347	0.626543	0.7032
Sub_metering_1	0.484401	0.123111	-0.195976	0.489298	1.000000	0.054721	0.102571	0.1250
Sub_metering_2	0.434569	0.139231	-0.167405	0.440347	0.054721	1.000000	0.080872	0.0852
Sub_metering_3	0.638555	0.089617	-0.268172	0.626543	0.102571	0.080872	1.000000	0.1787
Sub_metering_4	0.701380	0.211624	-0.271371	0.703258	0.125067	0.085201	0.178724	1.0000

In [23]:
```
sns.jointplot(x='Sub_metering_2',y='Global_active_power',data=df,kind='scatter')
```
Out[23]: `<seaborn.axisgrid.JointGrid at 0x20a09b05df0>`



In [32]:
```
sns.jointplot(x='Sub_metering_3',y='Global_active_power',data=df,kind='scatter')
```
Out[32]: `<seaborn.axisgrid.JointGrid at 0x2b00dbd8a30>`



In [24]:
```
df.head()
```
Out[24]:

	Global_active_power	Global_reactive_power	Voltage	Global_intensity	Sub_metering_1	Sub_metering_2	Sub_metering_3	Sub_metering_4
datetime								
2006-12-16 17:24:00	4.216	0.418	234.84	18.4	0.0	1.0	17.0	52.266667
2006-12-16 17:25:00	5.360	0.436	233.63	23.0	0.0	1.0	16.0	72.333333
2006-12-16 17:26:00	5.374	0.498	233.29	23.0	0.0	2.0	17.0	70.566667
2006-12-16 17:27:00	5.388	0.502	233.74	23.0	0.0	1.0	17.0	71.800000

In [25]:
```
x=df.drop(columns=['Global_active_power'])
x.head()
```

Out[25]:

	Global_reactive_power	Voltage	Global_intensity	Sub_metering_1	Sub_metering_2	Sub_metering_3	Sub_metering_4
datetime							
2006-12-16 17:24:00	0.418	234.84	18.4	0.0	1.0	17.0	52.266667
2006-12-16 17:25:00	0.436	233.63	23.0	0.0	1.0	16.0	72.333333
2006-12-16 17:26:00	0.498	233.29	23.0	0.0	2.0	17.0	70.566667
2006-12-16 17:27:00	0.502	233.74	23.0	0.0	1.0	17.0	71.800000
2006-12-16 17:28:00	0.528	235.68	15.8	0.0	1.0	17.0	43.100000

In [26]:
```
y=df['Global_active_power']
y.head()
```

Out[26]:
```
datetime
2006-12-16 17:24:00    4.216
2006-12-16 17:25:00    5.360
2006-12-16 17:26:00    5.374
2006-12-16 17:27:00    5.388
2006-12-16 17:28:00    3.666
Name: Global_active_power, dtype: float64
```

In [27]:
```
from sklearn.model_selection import train_test_split
```

In [28]:
```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=101)
```

In [29]:
```
print(x_train.shape)
```

```
(1434496, 7)
```

In [30]:
```
print(x_test.shape)
```

```
(614784, 7)
```

In [31]:
```
print(y_train.shape)
```

```
(1434496,)
```

In [32]:
```
print(y_test.shape)
```

```
(614784,)
```

In [33]:
```
from sklearn.linear_model import LinearRegression
```

In [34]:
```
lm=LinearRegression()
```

In [35]:
```
lm.fit(x_train,y_train)
```

Out[35]:
```
LinearRegression()
```

In [36]:
```
pred=lm.predict(x_test)
```

In [37]:
```
pred
```

Out[37]:
```
array([3.85 , 0.738, 0.222, ..., 1.562, 0.604, 1.534])
```

In [38]:
```
from sklearn import metrics
print('MAE:',metrics.mean_absolute_error(y_test,pred))
print('R Square value:',metrics.r2_score(y_test,pred))
```

```
MAE: 5.4602457327214033e-14
R Square value: 1.0
```


In [38]:
```
from sklearn import metrics
print('MAE:',metrics.mean_absolute_error(y_test,pred))
print('R Square value:',metrics.r2_score(y_test,pred))
```

```
MAE: 5.4602457327214033e-14
R Square value: 1.0
```


In [39]:
```
lm.predict([[0.418,234.84,18.4,0.0,1.0,17.0,52.266667]])
```

```
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\base.py:450: UserWarning: X does not have valid feature names, but LinearReg
ression was fitted with feature names
  warnings.warn(
```

Out[39]:
```
array([4.21600002])
```

In [40]:
```
import pickle
filename='PCA_model.pkl'
pickle.dump(lm,open(filename,'wb'))
```

PCA_Flask.py

```
from flask import Flask,request,render_template

import numpy as np

import pandas as pd

import pickle

model=pickle.load(open('PCA_model.pkl','rb'))

app=Flask(__name__)

@app.route('/')

def home():

    return render_template('demo.html')

@app.route('/predict',methods=["POST"])

def predict1():

       input_features=[float(x) for x in request.form.values()]

    features_value=[np.array(input_features)]

features_name=['Global_reactive_power','Voltage','Global_intensity','Sub_metering_1','Sub_metering
_2','Sub_metering_3','sub_metering_4']

    df=pd.DataFrame(features_value,columns=features_name)

    output=model.predict(df)

        return render_template('result1.html',prediction_text=output)

if __name__ == "__main__" :

        app.run(debug=False)
```