5 Global_intensity 6 Sub_metering_1 7 Sub_metering_2 8 Sub_metering_3	Non-Null Count Dtype 1048575 non-null object 1048575 non-null object 1048575 non-null object
<pre>dtypes: float64(1), object memory usage: 72.0+ MB  data.head()</pre>	1048575 non-null object 1044506 non-null float64 (8)
Date Time Globa  0 16/12/2006 17:24:00  1 16/12/2006 17:25:00  2 16/12/2006 17:26:00  3 16/12/2006 17:27:00  4 16/12/2006 17:28:00	al_active_power         Clobal_reactive_power         Voltage         Global_intensity         Sub_metering_2         Sub_metering_3           4.216         0.418         234.84         18.4         0         1         17.0           5.36         0.436         233.63         23         0         1         16.0           5.374         0.498         233.29         23         0         2         17.0           5.388         0.502         233.74         23         0         1         17.0           3.666         0.528         235.68         15.8         0         1         17.0
: Date Time  0 16/12/2006 17:24:00  1 16/12/2006 17:25:00  2 16/12/2006 17:26:00  3 16/12/2006 17:27:00	5.36       0.436       233.63       23       0       1       16.0         5.374       0.498       233.29       23       0       2       17.0
4 16/12/2006 17:28:00  1048570 13/12/2008 21:34:00 1048571 13/12/2008 21:35:00 1048572 13/12/2008 21:36:00	3.666       0.528       235.68       15.8       0       1       17.0                 0.426       0.076       242.27       1.8       0       0       0.0         0.424       0.076       242.1       1.8       0       0       0.0
1048573 13/12/2008 21:37:00  1048574 13/12/2008 21:38:00  1048575 rows × 9 columns  Data Cleaning  : # Handling Missing Values	0.422 0.078 242.61 1.8 0 0 0.0
	0 0 0 0 0 0 0 0 0
<pre>dtype: int64  : data = data.dropna(subset  : #Again checking for null data.isnull().sum().sum()  : 0  : # Convert Data Types</pre>	values
<pre>data = data.set_index('Data C:\Users\DELL\AppData\Local    data['Date_Time'] = pd.to C:\Users\DELL\AppData\Local A value is trying to be set Try using .loc[row_indexer, See the caveats in the document</pre>	_datetime(data['Date'] + ' ' + data['Time'])
<pre>from sklearn.preprocessing scaler = MinMaxScaler()</pre>	Standardization for numerical columns
Feature Engineering  : #Extract feature like house data['Hour'] = data.index data['Day_of_week'] = data  Validate the Prepared Data  : #Plot time-series data to	.hour a.index.dayofweek
data['Global_active_power plt.title("Energy Consumpt plt.show()	
0.6 -	
0.0 - 2001-0 <sup>A</sup>	2007.07 2001.30 2008.04 2008.01 2008.10 2009.01  Date_Time
<pre>data['Date'] = pd.to_date' data['Date'] = data['Date'  data['Time'] = pd.to_date' data['Time'] = data['Time'] #select only numeric column.</pre>	time(data['Date'], errors='coerce') '].map(pd.Timestamp.toordinal)  time(data['Time'], errors='coerce') '].map(pd.Timestamp.toordinal)
<pre>plt.show()  C:\Users\DELL\AppData\Local   data['Date'] = pd.to_date  C:\Users\DELL\AppData\Local   cify a format.</pre>	annot = True, cmap = 'coolwarm')  1\Temp\ipykernel_12848\2748350882.py:5: UserWarning: Parsing dates in %d/%m/%Y format when dayfirst=False (the default) was specified. Pass `dayfirst=True` or specify a format to silence this warning etime(data['Date'], errors='coerce')  1\Temp\ipykernel_12848\2748350882.py:8: UserWarning: Could not infer format, so each element will be parsed individually, falling back to `dateutil`. To ensure parsing is consistent and as-expected, etime(data['Time'], errors='coerce')  -0.065.00650.14-0.0660.0110.03400082001X0041
Time - Global_active_power -0.065 Global_reactive_power -0.006 Voltage -0.14 Global_intensity -0.066	-0.4
Sub_metering_1 -0.011 Sub_metering_2 -0.034 Sub_metering_3-0.0003 Hour-0.001 Day_of_week 0.0043	-0.0  -0.16 0.15 -0.18 0.46 0.061
Exploratory Data Analysis	Global_active_po Global_reactive_po Volt Global_inten Sub_meterin Sub_meterin Day_of_w
#Visualize Energy Consump data['Global_active_power plt.xlabel('Time') plt.ylabel('Global Active plt.show()	'].plot(figsize = (10,5) , title = 'Energy Consumption over Time')
Global Active Power (KW)  9 9 8'0	
0.0 - 2001-0 <sup>1</sup> 2001-0 <sup>4</sup>	2007.01 2007.10 2008.01 2008.01 2008.01 2008.10 2009.01 Time
<pre>daily_usage.plot(title = plt.show()</pre>	l_active_power'].resample('D').sum() 'Daily Enrgy Consumption', figsize = (10,6))  Daily Enrgy Consumption
300 -	
100 -	
	Jul Oct Jan Apr Jul Oct 2008 Date_Time  al_active_power'].resample('W').sum() Weekly Energy Consumption", figsize=(10, 6))  Weekly Energy Consumption
1500 -	
1000 - 750 - 500 -	
	Jul Oct Jan Apr Jul Oct 2008 Date_Time  by (data.index.hour) ['Global_active_power'].mean() ar', title="Average Hourly Energy Usage", figsize=(10, 5))
plt.xlabel("Hour of the Dapht.ylabel("Global Active plt.show()	ay")
0.175 - 0.150 -	Average Hourly Energy Usage
	Average Hourly Energy Usage
0.150 -  (MX) 0.125 -  0.100 -  0.075 -  0.050 -  0.025 -	Hour of the Day 51 91 51 51 52 52 52 53 54 54 54 54 54 54 54 54 54 54 54 54 54
O.150 -  (MX) O.125 -  O.075 -  O.000	princ spides or stops al.active_power() colling_totalew = 240 .mean() loctive_power() colling (window = 24) .mean() loctive_power() colling (window = 24) .mean() loctive_power(), label = 'Original') bel = 'Wealing Mean(, color = 'cec') ean = relling_end , rolling_maan + rolling_and, color = 'aranga', alpha = 0.0, label = 'std Daw,'
O.150 -  (MX) 0.125 -  0.000	Declar spiles or draps  **All and the Day  **Hour of the Day  **Proposed 1. reliting (window = 24) .csex()  **Lamine_power(). reliting (window = 24) .csex()  **Lamine_power(). reliting (window = 24) .csex()  **Siling (mean + rolling_std, color = freed)  **Ext = "Veiling_then", reliting_mean + rolling_std, color = forange(, algas = 0.3, label = 10td Day,')  **The Rolling Mean and Std Day()
O.150 -  O.100 -  O.000 -  O.0050 -	prior spikes or drope  Hour of the Day
Detect Anomalies  # Identify Unusual consum rolling_mean = data['Globa. plt.figure(figsize = (12, plt.plot(data ['Global_act plt.plot(rolling_mean, lal plt.fill_between( data.index, rolling_me))  plt.legend() plt.title('Energy Usage w. plt.show()  C:\Users\DELL\AppData\Local fig.canvas.print_figure(figure)  1.0  0.8  0.6  0.4  0.7  0.8  0.8	Hard the Dy  Hard
O.150  O.100  O.075  O.000  O.025  O.000  O.025  O.001  Detect Anomalies  # Identify Unusual consum, rolling_mean = data['Globa.rolling_std = data['Globa.glt.plot(rolling_mean, lalplt.plot(rolling_mean, lalplt.fill_between( data.index, rolling_me))  plt.legend() plt.title('Energy Usage w.plt.show())  C:\Users\DELL\AppData\Localfig.canvas.print_figure()  1.0  O.8  O.6  O.6  O.7  O.8  O.6  O.7  O.8  O.7  O.8  O.9  O.9  O.9  O.9  O.9  O.9  O.9	State States of Control Contro
O.150  O.125  O.100  O.075  O.000  O.025  O.0025  O.001  O.025  O.001  Detect Anomalies   # Identify Unusual consum, rolling_mean = data['Globarolling_std = data['Globarolling_std = 'Global_ac'plt.plot(data ['Global_ac'plt.plot(rolling_mean, laiplt.fill_between( data.index, rolling_mean))  plt.legend() plt.title('Energy Usage w.plt.show()  C:\Users\Della\AppData\Localfig.canvas.print_figure()  1.0  O.8  O.6  O.6  O.7  Correlation Analysis  # Analyze relationship be cort.show()  C:\Users\Della\AppData\Localfig.canvas.print_figure()  1.0  O.8  O.6  O.7  O.8  O.7  Correlation ('Forture Correlation of the correlation	2
O.150 -  (W) 0.125 -  (W) 0.125 -  (W) 0.0125 -  (W) 0.075 -  (W) 0.075 -  (W) 0.075 -  (W) 0.075 -  (W) 0.050 -  (W) 0.025 -  (W) 0.026 -  (W) 0.027 -  (W) 0.027 -  (W) 0.027 -  (W) 0.027 -  (W) 0.028 -  (W) 0.029 -  (W) 0.029 -  (W) 0.029 -  (W) 0.030 -  (W) 0.04 -  (W) 0.050	## SECRET OF STATE OF
Detect Anomalies  # Identify Unusual consum, rolling_mean = data['Globarolling_std = data['Globarolling_mean, lall plt.fill_between( data.index, rolling_mean, lall plt.fill_between( data.index, rolling_mean, lall plt.show()  C:\Users\Delta\AppData\Location_data.index, rolling_mean, lall plt.show()  Data-index index i	Technique (1994) 2001 12 2018 2 20 20 20 20 20 20 20 20 20 20 20 20 2
O.150 -  (W) 0.125 -  (W) 0.125 -  (W) 0.125 -  (W) 0.075	Treature Correlation    Control   Co
Detect Anomalies  # Identify Unusual consum rolling_mean = data['Globa plt.figure(figsize = (12, plt.plot(data ['Globa]acriplt.plot(rolling_mean, lal plt.fil_between( data.index, rolling_me)  plt.legend() plt.title('Energy Usage w. plt.show()  C:\Users\DELL\AppData\Locafig.canvas.print_figure(i)  1.0  0.8  0.6  0.4  0.2  0.0  Date - 1  Time - Global_active_power -0.066  Global_reactive_power -0.066  Global_reactive_power -0.066  Sub_metering_2 -0.03  Sub_metering_2 -0.03  Sub_metering_2 -0.03  Sub_metering_3 -0.00  Hour -0.001  Day_of_week -0.004  ### Find and Visualize perions of the plt.show()  Date -1  Time -1  Global_reactive_power -0.066  Global_reactive_power -0.066  Sub_metering_1 -0.01  Sub_metering_2 -0.03  Sub_metering_2 -0.03  Sub_metering_3 -0.00  #### Pind and Visualize perions of the plt.show()  Date -1  Time -1  Global_reactive_power -0.066  Global_reactive_power -0.066  Global_reactive_power -0.066  Global_reactive_power -0.066  Global_reactive_power -0.066  Sub_metering_1 -0.01  Sub_metering_2 -0.03  Sub_metering_2 -0.03  Sub_metering_3 -0.00  #################################	The control of the co
Detect Anomalies  # Identify Unusual consum, rolling_mean = data['Globa plt.figure(figsize = (12, plt.plot(data ['Globan]tfill_between( data.index, rolling_ms))  plt.legend() plt.fill_between( data.index, rolling_ms))  plt.legend() plt.title('Energy Usage wplt.show())  C:\Users\DELL\AppData\Local fig.canvas.print_figure()  1.0	
Detect Anomalies  # Identify Unusual consum, rolling, mean = data('Globarolling, mean = data('Globarolling, mean') and plt. figure (figsize = (12, plt.plot(data('Global, ac)) title('Energy Usage with title('Distributh on c) goods-10-120 12123:00 0.99 2008-10-120 12123:00 0.99 2008-10-120 12123:00 0.99 2008-11-30 2013-127:00 0.99	
Detect Anomalies  # Identify Unusual consum, rolling mean = data ("Global plt. figure (figsize = (12, plt. plot (data ("Global colling, std = data ("Global colli	The state of Control of the State of the S
Detect Anomalies  Detect Anomalies  # Identify Unusual consum rolling, man a data ("Global plt. figure (figisize = (12, plt. plot (data   Global plt. figure (figisize = (12, plt. plot (data   Global plt. figure (figisize = (12, plt. plot (data   Global plt. figure (figisize = (12, plt. plot (colling, pean   glt. plot (glt. plot   glt. plot   glt. plot (glt. plot   glt. plot   glt. plot (glt. plot   glt. plot   glt	
O.150  W 0.125  O.100  Detect Anomalies  * Identify Unusual consum; rolling_mean = data('Globarolling_mean = data('Globarolling_mean = data('Globarolling_mean, laipt.fill_between data.index, rolling_mean, laipt.fill_between data.index	
Detect Anomalies  **S Identify Unusual consum, rolling mean adata ("clobal poling and adata" ("clobal poling, and adata	
Detect Anomalies  Pisonate of Detect	
0.150  0.0125  0.0125  0.0125  0.025  0.025  0.025  0.025  0.025  0.025  0.025  0.025  0.026  0.026  0.027  0.027  0.027  0.028  0.028  0.029  0.029  0.029  0.029  0.029  0.029  0.029  0.029  0.029  0.029  0.029  0.029  0.029  0.029  0.029  0.029  0.029  0.030  0.030  0.04  0.04  0.14  0.15  0.05  0	
0.150  0.100  0.0000  0.0000  0.0000  0.0000  0.0000  0.00000  0.000000	
Detect Anomales  0.025	
Detect Anomalies  0.125- 0.125	
Delect Anomalies  0.100  0.0000  0.0000  0.0000  0.0000  0.0000  0.00000  0.000000	
Detect Anomalies    Florance   Charles   Consumer   Con	
Delect Anomalies  7	