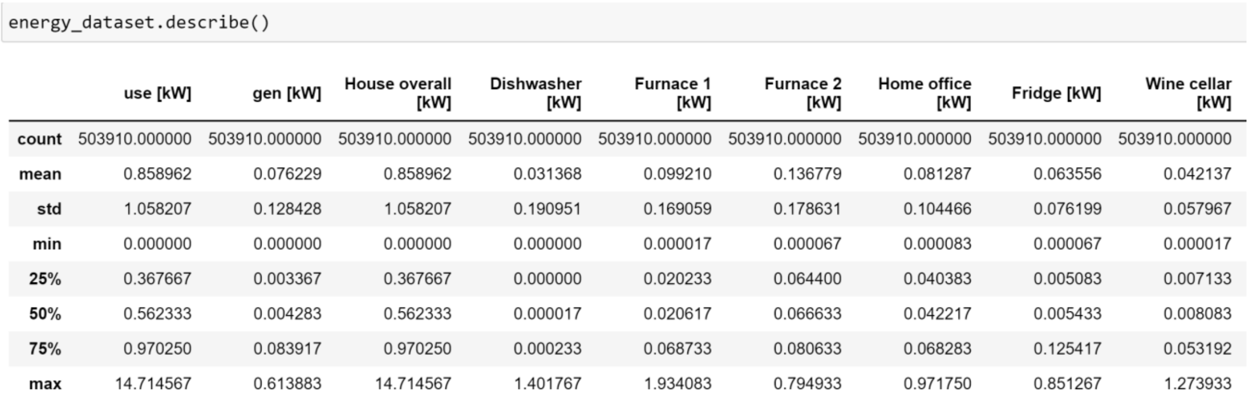
Project: Supervised Machine Learning Regression

Data Source: https://www.kaggle.com/datasets/taranvee/smart-home-dataset-with-weather-information?resource=download

Objective: The main objective of the analysis is to predict the total usage of energy of that smart home depending on the various use of the appliances. The resultant model will be focused on prediction how the energy is used in different places of that smart home.

Description of the data set: This Dataset contains the readings with a time span of 1minute of 350 days of house appliances in kW from a smart meter and weather conditions of that particular region. This dataset is involved with the energy usage of different appliances such as dishwasher, furnace, fridge, microwave and the different room such as living room, home office, kitchen, etc. Initially, this dataset has shape of (503911, 32).

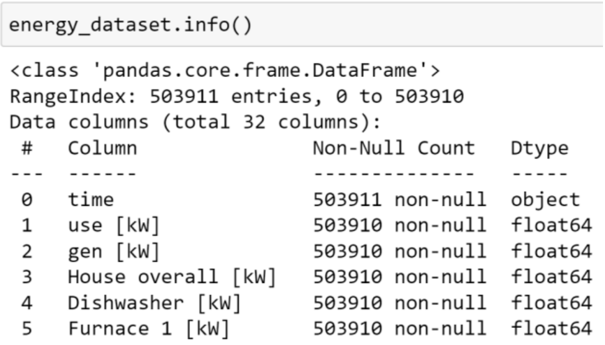
From the description of data we can observe the following information where we can find mean, standard deviation, maximum, minimum and the percentile of each features of the data:



```
energy_dataset.describe()
```

	use [kW]	gen [kW]	House overall [kW]	Dishwasher [kW]	Furnace 1 [kW]	Furnace 2 [kW]	Home office [kW]	Fridge [kW]	Wine cellar [kW]
count	503910.000000	503910.000000	503910.000000	503910.000000	503910.000000	503910.000000	503910.000000	503910.000000	503910.000000
mean	0.858962	0.076229	0.858962	0.031368	0.099210	0.136779	0.081287	0.063556	0.042137
std	1.058207	0.128428	1.058207	0.190951	0.169059	0.178631	0.104466	0.076199	0.057967
min	0.000000	0.000000	0.000000	0.000000	0.000017	0.000067	0.000083	0.000067	0.000017
25%	0.367667	0.003367	0.367667	0.000000	0.020233	0.064400	0.040383	0.005083	0.007133
50%	0.562333	0.004283	0.562333	0.000017	0.020617	0.066633	0.042217	0.005433	0.008083
75%	0.970250	0.083917	0.970250	0.000233	0.068733	0.080633	0.068283	0.125417	0.053192
max	14.714567	0.613883	14.714567	1.401767	1.934083	0.794933	0.971750	0.851267	1.273933

Data Cleaning: At the beginning of the analysis, it is very important to do exploratory data analysis to visualize the raw data to handle the missing data, outliers. From the following inquiry, we can find how many missing values are there in each feature:

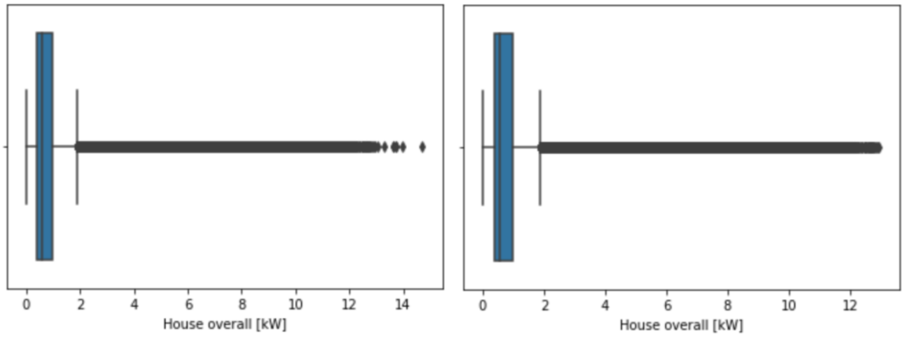


```
energy_dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 503911 entries, 0 to 503910
Data columns (total 32 columns):
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   time                 503911 non-null  object
 1   use [kW]             503910 non-null  float64
 2   gen [kW]             503910 non-null  float64
 3   House overall [kW]   503910 non-null  float64
 4   Dishwasher [kW]      503910 non-null  float64
 5   Furnace 1 [kW]       503910 non-null  float64
```

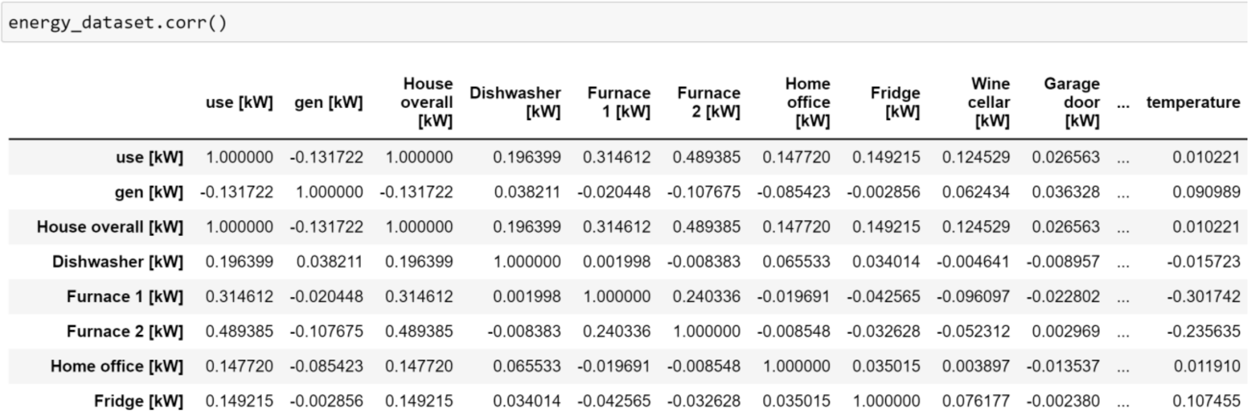
We have observed that the number of missing values is very few, therefore, we can drop those which will not do any impact in our observations.

The target feature is overall household usage of energy, we can check whether there is any outlier. From the following inquiry, it is observed that there are some anomalies. We can take care of those.



(a) Before removing outliers (b) After removing outliers

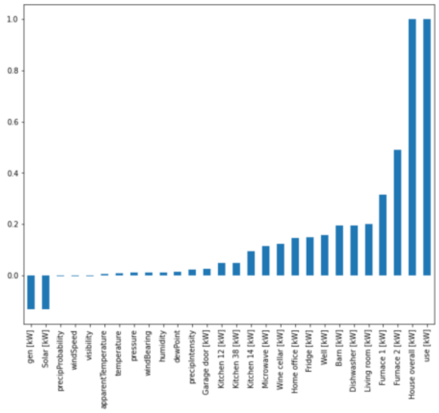
Feature Engineering: We need to find the correlation among the features to select the most appropriate features.



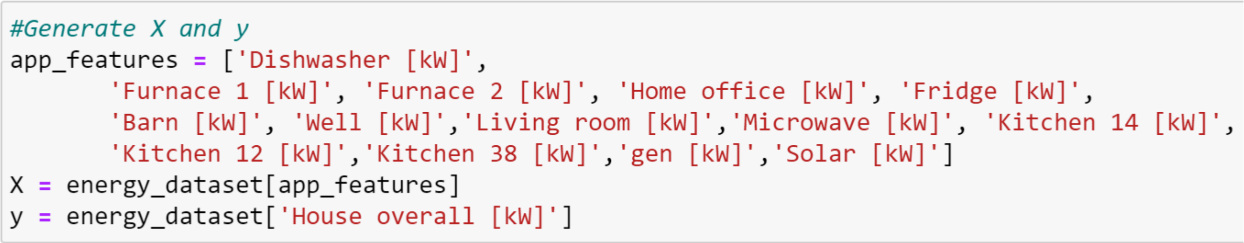
```
energy_dataset.corr()
```

	use [kW]	gen [kW]	House overall [kW]	Dishwasher [kW]	Furnace 1 [kW]	Furnace 2 [kW]	Home office [kW]	Fridge [kW]	Wine cellar [kW]	Garage door [kW]	...	temperature
use [kW]	1.000000	-0.131722	1.000000	0.196399	0.314612	0.489385	0.147720	0.149215	0.124529	0.026563	...	0.010221
gen [kW]	-0.131722	1.000000	-0.131722	0.038211	-0.020448	-0.107675	-0.085423	-0.002856	0.062434	0.036328	...	0.090989
House overall [kW]	1.000000	-0.131722	1.000000	0.196399	0.314612	0.489385	0.147720	0.149215	0.124529	0.026563	...	0.010221
Dishwasher [kW]	0.196399	0.038211	0.196399	1.000000	0.001998	-0.008383	0.065533	0.034014	-0.004641	-0.008957	...	-0.015723
Furnace 1 [kW]	0.314612	-0.020448	0.314612	0.001998	1.000000	0.240336	-0.019691	-0.042565	-0.096097	-0.022802	...	-0.301742
Furnace 2 [kW]	0.489385	-0.107675	0.489385	-0.008383	0.240336	1.000000	-0.008548	-0.032628	-0.052312	0.002969	...	-0.235635
Home office [kW]	0.147720	-0.085423	0.147720	0.065533	-0.019691	-0.008548	1.000000	0.035015	0.003897	-0.013537	...	0.011910
Fridge [kW]	0.149215	-0.002856	0.149215	0.034014	-0.042565	-0.032628	0.035015	1.000000	0.076177	-0.002380	...	0.107455

It is much better representation if we can visualize this correlation in graph as follow,



From the graph, it is observed that “House overall [kW]” and “use [kW]” contain the same information. We can select the appropriate features which are more related to the target feature of “House overall [kW]” from the graph as follows:



```
#Generate X and y
app_features = ['Dishwasher [kW]',
       'Furnace 1 [kW]', 'Furnace 2 [kW]', 'Home office [kW]', 'Fridge [kW]',
       'Barn [kW]', 'Well [kW]','Living room [kW]','Microwave [kW]', 'Kitchen 14 [kW]',
       'Kitchen 12 [kW]','Kitchen 38 [kW]','gen [kW]','Solar [kW]']
X = energy_dataset[app_features]
y = energy_dataset['House overall [kW]']
```

Polynomial features: To extend the features we used polynomial features from scikit learn preprocessing library.

Then, we scale the features and split training and test set where 30 percent data is in test set.

Three Models: We built three models (Linear Regression, Lasso Regression and Ridge Regression) to compare each other to get the most accurate model.

We got the following results for three regressions. The number of coefficients which are not equal to zero is higher for linear regression.

```
r2 score for Linear Regression: 0.3032518396506876
Magnitude of Linear Regression coefficients: 107467260239.76361
Number of coefficients not equal to 0 for Linear Regression: 120
```

Regularization: We can use regularization to do the trade off between bias and variance to prevent overfitting problem. Here we used Lasso and Ridge regression.

```
Lasso Regression: alpha =0.01
r2 score for Lasso Regression: 0.1356309503665677
Magnitude of Lasso Regression coefficients: 2.7702535912862603
Number of coefficients not equal to 0 for Lasso Regression: 36

Lasso Regression: alpha =0.001
r2 score for Lasso Regression: 0.28684240610827316
Magnitude of Lasso Regression coefficients: 6.426406228452473
Number of coefficients not equal to 0 for Lasso Regression: 96

Ridge Regression: alpha = 0.01
r2 score for Ridge Regression: 0.30325600207175707
Magnitude of Ridge Regression coefficients: 7.601464995674554
Number of coefficients not equal to 0 for Ridge Regression: 119
```

From the above analysis, for this particular dataset, it is found that in case of Linear regression, the number of nonzero coefficients is higher. After utilizing regularization, we can reduce these coefficients lower, however, in case of Ridge regression for this dataset, the number of non-zero coefficients remain same.